Window-aware guided image filtering via local entropy

Chong Liu1,2 | Cui Yang2 | Jun Wang1

1 College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China
2 College of Mathematics and Computing Science, Anqing Normal University, Anqing, Anhui, China

Correspondence
Jun Wang, College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China.
Email: liuchong@aau.edu.cn

Abstract
Guided image filtering is one of the widely used techniques in computer vision. However, it commonly leads to over-smoothed edges and a distorted appearance when tackling intricate texture patterns and complex noise. In this paper, a window-aware image filtering framework based on the bilateral filter guided by the local entropy is presented. The key idea of the authors’ proposed approach is to design a novel guidance input and a non-box filtering window. Specifically, using the Gaussian spatial kernel and the local entropy, a GEF that can maintain image feature details and yield a robust guidance input for BF is constructed. Meanwhile, based on an intensity-similar strategy, the local non-box filtering window is designed for the further preservation of edge structures. The authors’ approach not only inherits the advantages of bilateral filter i.e. simplicity, parallelisation and easiness of programming, but also is more powerful than bilateral filter and its variants. In addition, the guided entropy filter and the non-box window can also be transplanted to other local filters and can effectively improve the filtering effects. The qualitative and quantitative experimental results demonstrate that the authors’ approach has good performance in image denoising, texture (or background) smoothing, edge extraction and other applications in image processing.

1 | INTRODUCTION

Image smoothing aims to smooth interference information, such as noise and unexpected texture background, which is an important operator for maintaining the image quality in image transportation and preservation. So far, although many existing image filtering methods can remove unexpected details well, they commonly fail to preserve the small- and even medium-scale structures.

Among these excellent methods, due to the simplicity, parallelisation and easiness of programming, bilateral filter (BF) [1] is widely used in various image processing tasks. The success of BF originates from the collaboration between the Gaussian spatial kernel [2] responsible for smoothing out interference and the Gaussian range kernel for maintaining image features. However, for images contaminated by high-frequency noise or coarse textures, bilateral filtering, like many other filtering methods, still has trouble in reaching a proper balance between detail preservation and image smoothing. To alleviate this problem, the simple and effective improvements of BF have been proposed by designing a proper guidance input during the past two decades. For example, local schemes [3–5] are put forward to improve the distance norm of the Gaussian range kernel of BF, by utilising the information in the filtering neighbourhood. Non-local methods, such as Means-BF [6, 7] and NLBF [8, 9], attempt to obtain the input of the range kernel through mean filtering or other non-linear filtering approaches. These approaches are able to yield high-fidelity results for processing low-frequency noise or low-scale texture background, while they could not work well when removing complex texture backgrounds.

The recently proposed multi-layer frameworks based on the guided BF can be exploited to better understand the low-frequency part of an image than the aforementioned single improvement of the range kernel. The multiple resolution bilateral filter (MBF) [10] applies a wavelet filter to approximate the low-frequency part of the image and achieves blending filtering. Similarly, Khadidja et al. [11] utilise the wavelet transform to deal with the low-frequency part of the image and combine WBF to perform image denoising. It is worth mentioning that the joint bilateral filtering [12] introduces the local image entropy to construct a threshold function, which could provide self-adaptive parameters for different types of images. Although these multi-layer frameworks of BF provide a more accurate output, they...
have more troubles of the time consumption and parameter settings.

In order to reduce the parameter settings of the guidance image and provide a robust guidance input for BF, we have designed an effective Gaussian entropy filter (GEF). The GEF can avoid setting the spatial and range kernel bandwidth similar to that of existing bilateral filtering. Meanwhile, we address a local non-box filtering window for our guided bilateral filtering to retain some low-scale significant features.

The main contributions of our proposed method are as follows.

- We construct an Gaussian spatial entropy filter to generate a robust guided image. The spatial entropy filter can utilise the grey information of pixels while avoiding the setting of the range bandwidth, which could improve the efficiency of the guided filter.
- We provide a simple window-aware strategy to find non-box filtering windows. All significant structure details in a non-box window could be effectively preserved by weighting the similar pixels in non-box window.
- Our proposed window-aware strategy could reduce the sensitivity of the filtering parameters due to the similarity of pixels in the filtering window, which is conducive to adjusting the experimental parameters for the final best results.

The dual improvements from the robust guidance and the sensitivity-aware filtering window could process the different image interference while preserving visually significant details (see Figure 1). The more results of both quantitative and qualitative comparisons are included in Section 5.

The remainder of this paper is organised as follows. In Section 2, we briefly review the related works, including the basic Gaussian filter, bilateral filter and their extensions. In Section 3, the definitions of BF, Gaussian filter, local entropy, the intensity-aware window and our window-aware guided bilateral filtering are detailed. The implementation of our approach is analysed in Section 4. In Section 6, we demonstrate some applications of our method compared with several similar methods, followed by the conclusion in Section 7.

## 2 RELATED WORKS

Image filtering is a fundamental skill in image processing. There are many literatures of this topic and representative approaches include wavelet transform [13], adaptive total variation [14], clustering learning [15] and low rank learning [16], etc. Recently, non-local methods, such as NLM [17] and BM3D [18], have achieved promising results in image denoising by capturing the similarity between patches within a broader context. In this section, we emphasise on the most relevant works on kernel filtering, including Gaussian filter, bilateral filter and their extensions.

Gaussian filter (GS) [2] is a conventional low-pass filter, which is commonly combined with other methods to handle high-level noise or coarse texture. For example, the Gaussian mixture model [19–21] integrate two or more Gaussian filters, whose parameters are obtained by learning-based methods. The combination strategy is also applied for image restoration and reconstruction [22]. Another idea is to combine the statistical information, such as Bayesian rules [23, 24], Hierarchical clustering [25] and Monte Carlo methods [26]. These approaches attempt to learn the noise model under a certain probability condition, for the purpose of reducing the error caused by the data discretisation. These methods leverage the robustness of Gaussian kernel to smooth the image, but it is difficult to simultaneously achieve noise removal and feature preservation when using a single Gaussian kernel.

As an improvement of Gaussian filter, Bilateral filter [1] provides better performance in edge-preserving filtering. There are also many other combinatorial techniques based on BF. For example, Yang et al. [27] decomposed the noisy image into components of different frequencies and direction response sub-bands, and then applied the statistical model into BF to obtain the on-the-ground details. To restore the natural information of an image, Mayank et al. [28] combined BF with
the minimum mean square error to reduce noise. Although the technique of the mixed filters could improve the filtering performance of BF, how to select a proper weight for each coordinate filter is an issue that still has not been tackled. In addition, there are improvements [29–32] of selecting parameters and guidance inputs by utilising learning methods, yet such methods may involve high computational complexity and difficulty in model verification, which could not be negligible. Recently, the framework based on joint bilateral filtering [33–36] has demonstrated that a proper design of the guidance input is considered an effective amelioration. In [37], Song et al. designed distinct patches according to image details, and regard the statistics of these patches are used as the guidance input for texture filtering, which could be applied to effectively remove image textures. Mayank and Bhupendra [38] proposed a different guidance input using spatial gradient that could differentiates edges and textures. In the work of Nagashettappa et al. [39], a partial differential equation is regarded as an approximate of the guidance input, and the second derivative of pixel value is employed to reflect the edge information, which could easily enhance edge details. The methods mentioned above share a common target, that is, to better flatten the noisy areas of the guidance image, but this lead to over-blurred structure details. Unlike the previous techniques, our method focuses on retaining the feature details of the guidance input rather than smoothing it, and feeding BF with sufficient information of features to achieve better edges preservation.

3 METHODOLOGY

In this paper, to seek the considerable smoothness and edge preservation in image filtering, we first slightly pre-processed the original input by Gaussian range kernel filtering. Then a local entropy filter was designed to generate a robust guidance for bilateral filtering, and meanwhile a window-aware strategy was adopted to optimise filtering windows. Based on the robust guidance input and the optimal filtering window, the final guided image filtering can obtain more accurate computer vision.

The whole pipeline of our approach is shown in Figure 2.

### 3.1 Bilateral filter and guided bilateral filter

Given an input image $I$, the output $J_p$ at each pixel $p$ obtained by bilateral filter (BF) [1] can be expressed as follows:

$$J_p = \frac{1}{k_p} \sum_{q \in \Omega_p} g_r(\|q - p\|)g_{\sigma}(\|I_q - I_p\|)I_q$$  

(1)

where $I_q$ is the intensity value of the noisy image at pixel $q$ and $\Omega_p$ represents the box window centred at $p$ with the size of $(2r + 1) \times (2r + 1)$. The output $J_p$ at pixel $p$ is a weighted average of all pixels $I_q$ in the neighbourhood $\Omega_p$. The spatial kernel $g_r$ and the range kernel $g_{\sigma}$ are both Gaussian functions, which are written as

$$g_r(\|q - p\|) = \exp\left(-\frac{\|q - p\|^2}{2d_r^2}\right)$$  

(2)

$$g_{\sigma}(\|I_q - I_p\|) = \exp\left(-\frac{\|I_q - I_p\|^2}{2d_{\sigma}^2}\right)$$  

(3)

The normalising term $k_p$ is defined as

$$k_p = \sum_{q \in \Omega_p} g_r(\|q - p\|)g_{\sigma}(\|I_q - I_p\|)$$  

(4)

where $\| \cdot \|$ represents the $L_2$ norm, and the parameters $d_r$ and $d_{\sigma}$ control the bandwidth of the spatial and range kernel, respectively. When $g_r(\|I_q - I_p\|) \equiv 1$, Equation (1) represents Gaussian spatial filter (GS), and $g_r(\|q - p\|) \equiv 1$ is Gaussian range kernel filter (GF).
For seeking a better filtering effect, the guided bilateral filter (GBF) utilises the other filtering output $G$ as the input of Gaussian range kernel, which could be expressed as

$$ J_p = \frac{1}{k^2_p} \sum_{q \in \Omega_p} g(\|q - p\|)g(\|G_q - G_p\|)I_q. \quad (5) $$

Constructing a robust $G$ is the core of guided filtering, which is also one of the key schemes to be proposed in this paper.

### 3.2 Gaussian spatial entropy filter

Many studies [36, 37, 40] have shown that an effective constructed guidance image should own a better protection of image features rather than a smoother filtering output. Considering that the image entropy [41] can well measure the intensity energy of pixels, we use the local entropy and the Gaussian spatial kernel to construct a GEF, which behaves like BF with the good preservation of edge structures while avoiding parameter setting for Gaussian range kernel.

Given a neighbourhood $\Omega_p$ centred at a pixel $p$, the local entropy [40] of the pixel $p$ can be defined as

$$ E_p = -\sum_{k=0}^{L-1} \xi_k \log_2 \xi_k $$

where $\xi_k = \frac{1}{N_k}$ is the probability of grey level $k$ appearing in the neighbourhood $\Omega_p$ and $N_k$ represents the number of pixels with the grey level $k$. $L$ is the total amount of grey levels.

According to the image entropy, the output GEF at the pixel $p$ can be expressed as

$$ GEF_p = \frac{1}{\tau} \sum_{q \in \Omega_p} g(\|q - p\|)\xi_q \log_2 \xi_q I_q $$

where $\tau = \sum_{q \in \Omega_p} g(\|q - p\|)E_q$, and $E_q = \log_2 \frac{1}{N_q}$ represents the information entropy of pixels in the window $\Omega_q$. $N_q$ is the number of all pixels in $\Omega_q$. $I_q$ belongs to the grey level interval $[a_q, b_q]$ with $k$, and $a_q \leq b_q \leq [0, 255]$. The Gaussian kernel $g(\|q - p\|)$ is defined as

$$ g(\|q - p\|) = \exp(-\|q - p\|^2). $$

From Equation (7), we can find GEF does not need to set the corresponding parameters except the window radius $r$. Meanwhile, the robust output is also provided by GEF, due to the small fixed spatial bandwidth of Gaussian kernel.

Figure 3 exhibits the robustness of GEF. We can see from the grey histogram shown in Figure 3 that the information distribution of output is stable, when the window radius $r$ is larger than 4. The robustness could be easily inferred from Equation (2). Since $\|q - p\| > 4$, the weight at each pixel $q$ produced by the spatial kernel $g(\|q - p\|)$ is almost zero. Therefore, the large window radius have no effect on the output of GEF at the weight with a zero value.

GEF with robust output and no too many parameter settings can better serve GBF, determined by experiments in Section 5.

In general, a larger size of filtering window will be used in image filtering for a better smoothness, yet a large-size window should lead to the over-smoothness. For this reason, many techniques have been developed to select filtering windows for edge preservation, such as side window [42, 43] and edge-aware windows [44]. However, these improvements of sub-windows do not provide significant structure protection for fine details. To retain low-scale structural details, we propose a simple and effective window-aware technique that can find a non-box window to redefine the intensity-similar information of each central pixel.

### 3.3 Intensity-aware window

Assuming the filtering window $\Omega_p$ centred at each pixel $p$ with the size of $(2r + 1) \times (2r + 1)$, pixels $q$ whose intensities $I_q$ are similar as $I_p$ can be collected according to the following formula:

$$ \|I_q - I_p\| < \varepsilon $$

where $\varepsilon$ is a threshold that can be used to determine the intensity-similar window $\Gamma_p$ at the pixel $p$. By Equation (8), we obtain the window $\Gamma_p$ as follows:

$$ \Gamma_p = \{q \mid \|I_q - I_p\| < \varepsilon, q \in \Omega_p\}. \quad (9) $$

The intensity-aware window $\Gamma_p$ is an irregular region, but it can better reflect the information of the central pixel $p$, as shown Figure 4. Clearly, the intensities of pixels in the region marked with red ‘1’ are more similar to $I_p$.

We perform BF and window-aware bilateral filtering for the noisy image $cat$ at noise level $\sigma = 10, 20$, and compute the results of PSNR, respectively. Compared with the non-box filtering window, the filtered image obtained by window-aware bilateral filtering has better quality than that by BF with the same parameters (see Figure 5). Furthermore, as can be seen from Figure 5, the optimal PSNR of the filtered image obtained by
The similar intensity-aware technique used to select filtering windows. (a) Box window \( \Omega_p \). (b) The intensity-similar region labelled with the red '1'. (c) The intensity-similar window \( \Gamma_p \).

The relationships between PSNR and \( d_\sigma \). The curves on the upper row show the result of BF for the image with the noise level \( \sigma = 10 \), and the curves on the row below are the results at the noise level \( \sigma = 20 \). The red and black dots denote the maximal PSNR values of window-aware and box filtering images, respectively.

Comparisons of BF with box and non-box filtering windows. (a)–(d) Filtered images with the parameter \( d_\sigma = \{0.07, 0.12, 0.17, 0.22\} \). The filtered image by BF with the box window is on the upper right, and the lower left part shows the result of the intensity-aware window

(WGBF) can be expressed as

\[
J_p = \begin{cases} 
    I_{med}, & \text{if } \min_{q \in \Omega_p} \{ |I_q - I_p| \} > \varepsilon, \\
    \frac{1}{k_p} \sum_{q \in \Gamma_p} g(\|q - p\|)g(\|G_q - G_p\|)I_q, & \text{else}
\end{cases}
\]

where \( \varepsilon \) represents the intensity similarity between pixels in \( \Omega_p \) and the pixel \( p \). The intensity similarity is determined as Equation (8).

For an image with the high-frequency noise or large-scale background textures, a slight pre-processing filtering should be required to improve the guidance image \( G \), which could weaken the difference between the central pixel \( p \) and any pixel \( q \) in \( \Omega_p \), such that the filtering window could be accurately adjudged.

The pseudo code of our algorithm is detailed in Algorithm 1, and our code is publicly available 1.

1 https://github.com/liuchong777/WGBF

3.4 Window-aware guided bilateral filtering

According to the intensity-aware window \( \Gamma_p \) and a robust grey-scale image guidance \( G \) produced by GEF, the output at each pixel \( p \) of our proposed Window-aware Guided Bilateral Filtering using intensity-aware windows is significantly better than that by using box windows at a higher noise level, which demonstrates that intensity-aware windows can enhance the preservation of local details.

Figure 6 shows a visual comparison of the filtered output of BF and the output of window-aware bilateral filtering with different parameters \( d_\sigma \) when the noise level is \( \sigma = 20 \), from which we can see that the box filtering image will be blurred with the increase of frequency bandwidth. However, for the non-box window bilateral filtering, the bandwidth \( d_\sigma \) is much less sensitive to the smoothness provided by the spatial kernel, which can further provide the protection of fine edge details (see cat’s whiskers in Figure 6).

According to the intensity-aware window \( \Gamma_p \) and a robust grey-scale image guidance \( G \) produced by GEF, the output at each pixel \( p \) of our proposed Window-aware Guided Bilateral Filtering

Due to the high preservation of the intensity-aware window for similar features, WGBF may result in some local singular values when encountering strong interference, such as high-frequency noise. In order to address this issue, we adopt a median alternative strategy, by which the median intensity \( I_{med} \) in box window \( \Omega_p \) will replace the singular output \( J_p \) of WGBF to suppress the mutation. Thus, the final optimal output \( J_p \) of WGBF based on the intensity-aware window can be rewritten as

The pseudo code of our algorithm is detailed in Algorithm 1, and our code is publicly available 1.

1 https://github.com/liuchong777/WGBF
4 | EXPERIMENTAL ANALYSIS

4.1 | Parameters analysis

From what has been discussed above, our approach for image smoothing consists of image pre-processing, guidance image filtering and guided window-aware filtering.

In the first stage, Gaussian range kernel filter (GF) is chosen for image pre-processing, because of the capability of edge preservation. The pretreatment stage is intended to slightly smooth the interference and to facilitate subsequent operations. The parameters $d_r$ and $d_d$ for GF are set to relatively small values. For instance, we select the window radius $r = 3$ and $d_d = 0.1$ for a noisy image with the noise level $\sigma = 30$ and most of conventional textured images as shown in Figure 1(Harp). For other scale images, only $d_\sigma$ needs to be fine-tuned.

In the second stage, only the window radius $r$ of GEF needs to be tuned. According to the analysis in Section 3.2, we fix a window radius $r = 4$ for any scale image.

In the third stage, due to the performance of intensity-aware window, the window radius $r$ have a less effect on WGBF, which can be set to 9. The parameter $d_\sigma$ could be the same as the value in first stage. In addition, we define the thresholds $\varepsilon = \lambda = 0.12 + 0.002(\sigma - 10)$ that could basically satisfy different noise levels. When dealing with texture images, the fine-tuning strategy is similar to noise levels. In specific applications, the parameter $d_\sigma$ can be tuned appropriately.

4.2 | Performance analysis of WGBF

Given the same values of $r$ and $d_\sigma$, the bilateral filter (BF), GBF and window-aware guided bilateral filter (WGBF) are performed on a noisy signal input. We can find from Figure 7 that, the signal filtered by BF is obviously not as smoothed as that by GBF and WGBF, and the edge preservation of GBF and WGBF is also far better than BF. From visual perception, the smoothness of GBF and WGBF are the same, while WGBF provides the significant preservation for low-scale features [see the left of ‘Low’ in Figure 7(b) and (c)], due to the contribution of the intensity-aware window. More experimental results of WGBF for image filtering are exhibited in Section 5.

4.3 | Computational complexity

We analysed the computational complexities of our algorithm, BF and its improvements, and calculated the corresponding running time of various methods in the same operating environment. Our proposed algorithm consists of three stages, and the calculation in each stage is equivalent to that of BE. Therefore, the computational complexity is approximately $O(3N \cdot d_r^2)$, where $N$ is the number of the pixels in an input image $I$, and $d_r$ represents the window radius of BF. We compared the computational complexity of related improvements on BF: BF [1], robust bilateral filter (RBF) [8], optimally weighted bilateral filter (WBF) [45], multi-resolution bilateral filter (MBF) [10] and entropy-based bilateral filter (EBF) [12]. MBF has the highest time complexity which is about $O(N^2(L \cdot \log_2 N + d_r^2))$, followed by EBF which is $O(N(d_r^2 + L_w^2 + L_w^2))$, and the others are almost the same. Here, $L_r$, $L_w$ and $L_w$ are the wavelet filter length, the window size of entropy and the window size of the Wiener filter, respectively. We implement the above-mentioned methods on Matlab2019a using an Interel(R) CPU (3.6-Ghz), and obtain the timing data as shown in Table 1. The computational complexity of the proposed method may not be optimal, but subsequent applications show that the experimental results of our method are far better than that of similar improvements of BF.

![Figure 7](image_url) Noisy input smoothed by BF, GBF and WGBF. (a) Filtered by BF. (b) Filtered by GBF. (c) Filtered by WGBF. The parts in left side box of ‘Low’, ‘Medium’ and ‘Large’ correspond to the low-scale, medium-scale and large-scale features, respectively.

**ALGORITHM 1** Window-aware guided image filtering

| Input: Original image $I_i$ |
| output: Filtered image $I$ |
| 1: $I \leftarrow$ Image $I_i$ preprocessed by GF |
| 2: $G \leftarrow$ Obtain the guidance image by GEF |
| 3: for Each pixel $p$ in $I$ do |
| 4: $\Gamma_p \leftarrow$ Determined through the box window $\Omega_p$ |
| 5: $f_p \leftarrow$ Computed by Equation (10) with the guidance input $G$ |
| 6: end for |
| 7: $f \leftarrow$ Return the filtered image |
Image denoising

We introduced the common objective Peak Signal to Noise Ratio (PSNR) [46] and Structure Similarity Index Measure (SSIM) [47] to measure the quality of filtered images. PSNR and SSIM could be used to evaluate a filtered image with a reference, while EPI is adopted as the local similarity for the images without references, such as the real-world images and texture images. 

PSNR is computed by

$$PSNR = 10 \log_{10} \frac{S^2}{MSE} \quad (11)$$

where $MSE = \frac{1}{N} \sum_{x}(x(i) - y(j))^2$ is the mean square error, and $N$ is the number of pixels in the image $x$. $S$ is the maximum grey value of all pixels in the image $x$. When the image $x$ is 8-bit grey level within the scope of $[0, 255]$, we often take $S = 255$.

SSIM is defined as

$$SSIM = \frac{(2\nu_i\nu_j + C_1)(2\sigma_{ij} + C_2)}{(\nu_i^2 + \nu_j^2 + C_1)(\sigma_i^2 + \sigma_j^2 + C_2)} \quad (12)$$

where $\nu_i$ and $\nu_j$ are mean grey values of patch sets $\bar{x}$ and $\bar{y}$ that represent the structure region of $I$ and $J$, respectively. Parameters $\sigma_i^2$ and $\sigma_j^2$ represent standard deviations of the sets $\bar{x}$ and $\bar{y}$. $\sigma_{ij}$ is the covariance between $\bar{x}$ and $\bar{y}$. $C_1$ and $C_2$ are constants, which can be usually set: $C1 = (K1 \ast L)^2$, $C2 = (K2 \ast L)^2$, $K1 = 0.01$, $K2 = 0.03$, $L = 255$.

EPI is expressed as

$$EPI = \frac{\Gamma(x - \bar{x}, y - \bar{y})}{\sqrt{\Gamma(x - \bar{x}, x - \bar{y})\Gamma(y - \bar{y}, y - \bar{y})}} \quad (13)$$

where $\bar{x}$ and $\bar{y}$ are mean values in the region of interest (ROI) of the original image $x$ and filtered image $\hat{y}$ that are performed by the Laplace operator, respectively, and

$$\Gamma(s, t) = \sum_{(i, j) \in ROI} s(i, j)\hat{y}(i, j) \quad (14)$$

where $s(i, j)$ and $\hat{y}(i, j)$ represent the pixel values at the position $(i, j)$ on the original and filtered images, respectively. Since texture filtering would not only remove some texture details but also maintain local structure features, it is an appropriate method which can be used to choose ROI for texture image quality assessment.

5 APPLICATIONS

5.1 Image denoising

Some test images provided by Kodak\(^2\) and the real-world images from PolyU Dataset\(^3\) are selected. The real-world noisy image is captured by a camera with a higher ISO, and the pattern of the real-world noise is different from that of the Gaussian noise, but similar to that of some image textures. We compared our method with bilateral filter based on side window (SWBF) [42] and its recent improvements: entropy-based bilateral filter (EBF) [12], as well as some of the state-of-the-art denoising algorithms: BM3D [18]. All experiments are carried out using the optimal parameter settings and the provided test database.

For the visual contrast, we exhibited the denoised results of the image girl at the Gaussian noise level $\sigma = 30$ and for a real-world image $(2500 \times 1872)$ processed by above methods (see Figures 8 and 9). We also experimented with all the mentioned

\(^2\) http://decsai.ugr.es/cvg/dbimagenes
\(^3\) https://github.com/csjunxu/PolyU-Real-World-Noisy-Images-Dataset
techniques and computed the PSNR, SSIM and EPI of ROI labelled in the purple box region.

The results of PSNR in Table 2 show that BM3D has the unparalleled advantage over other methods in processing noisy images with Gaussian noise, while our method performs better for real-world images. When dealing with high-frequency noise or unnecessary background texture, BM3D tends to highlight the high-frequency region, which may cause the plaque phenomenon [see yellow circle in Figure 8(f)], due to the strategy of enhancement, and it may lead to the lack of artistic effect of images. In contrast, our method focuses on the protection (not 'enhancement') of details, some rough textured background in the noisy image can be better smoothed, while preserving significant details [see the zoomed white flat background and curve textures in Figure 9(h)]. And the results of EPI also show that our method is superior to other similar filters, since these regions of interest existing some important texture features could only be well preserved by our proposed intensity-aware windows.

5.2 | Image detexturing

We chose two texture images with fine and complex texture patterns, and compared our algorithm with some excellent detexturing methods: BF [1], ROG [49], kRTV [50], RGF [51]. Meanwhile, we zoomed the area with more feature details and relatively weak texture as the ROI to compute EPI of each filtered image.

Figures 10 and 11 present the filtered results of images with fine, coarse-scale textures, respectively. From Figures 10 and 11, we can see that the performance of our algorithm is almost close to that of ROG and kRTV in terms of overall texture removal, and is far better than that of BF and RGF. Nevertheless, the zoomed interested region shows that our approach preserve the details better, such as the retention of the hair in Figure 10 and the teeth in Figure 11. The values of EPI also demonstrate that our method achieves the desired retention of significant details on fine and complex scale texture images.

5.3 | Image background smoothing

Background smoothing mainly refers to remove the existing feature details that affect some special application of computer vision. Similar to textures, some of the background details may be unimportant textures, but some are important feature details, such as fish scales, uneven surfaces etc. In this paper, we employed methods: BF [1], RGF [51], ROG [49] and kRTV [50] to smooth the background for obtaining the clean image. Generally, the texture filtering methods kRTV and ROG could smooth the background without the large-scale interruption very well, yet they could also easily blur some low-scale structure details or form some artifacts [see eye areas in Figures 12 and 13(d) and (e)]. Conversely, RGF entirely fails to remove most of the background textures, due to the reinforcement of all details. Since our method emphasizes the preservation of features only based on the similarity of details, we can tune the intensity threshold to control the smoothness for the removal of some similar coarse background details, and preserve the fine-scale features that are different from background details, as shown freckles and water wave in Figures 12 and 13(f) and (l). The intuitive experimental results demonstrate the superiority of our proposed algorithm for background smoothing.

5.4 | Edge extraction

Our algorithm also benefits the pre-processing step of edge extraction, especially when the image contains noise or complex textures. In order to illustrate the effectiveness of our method
in edge extraction, we conduct experiments on both the noisy image (Parrot) and the textured image (Sheep), whose edges are extracted by TV \[52\]. We compare the edges extracted from the original image, the image pre-processed by BF and the image preprocessed by our method. We can find plenty false edges in Figures 14 and 15(b) and (c) caused by noise, grass textures and the sheep’s wool, while the result of our method [see Figures 14 and 15(d)] only retains semantically important edges.

5.5 | HDR tone mapping

A High Dynamic Range (HDR) \[53\] image records the actual scene, whose pixel values representing the brightness are far more than the grey level 255 and are difficult to display on the screen. For computer vision, linear decoding methods usually are used in HDR images, yet it will make some regions too bright or dim. The ways to solve this kind of the balance of

FIGURE 10  Tine texture filtering. The yellow box area is the interested region, which includes many details while less intrusive textures, and the purple box area is the zoomed interested region. The numerical result below each sub-figure (b)–(h) is the corresponding value of EPI

FIGURE 11  Complex texture filtering. The yellow box area is the interested region, which includes lots of details while less intrusive textures, and the purple box area is the zoomed interested region. The numerical result below each sub-figure is the corresponding value of EPI

FIGURE 12  The background (Face freckle) smoothing

FIGURE 13  The background (Water wave texture) smoothing
the colour brightness is called tone mapping, which will decomposes an HDR image into two layers: a base layer and a detail layer. The base layer is compressed with a certain contrast factor and the detail layer remains unchanged. Generally, the base layer needs to select an effective filter to remove noise while maintaining more edge features. Based on the filtered base layer, HDR tone mapping could be expressed as

\[ I = cI_b + I_d \]

where \( I_b \) is the output of the HDR image filtering, and \( I_d \) is the difference between the original HDR image and the filtering image \( I_b \). The compression factor \( c \) belonging to \((0, 1)\) determines the dynamic compression range. We compared our approach with the methods: ROG [49] and BF [1] that could perform well on image edges. As can be seen from the same zoomed region in Figure 16(b) and (d) that the HDR image processed by WGGF is much clearer than the HDR image processed by BF and ROG.

5.6 Multi-scale detail enhancement

Multi-scale detail enhancement requires the outputs of coarse-scale and fine-scale detail enhancements by image filtering with different parameter-scale settings twice, then calculates the average of these two results as the final improvement result. The preservation of the details of the filtering process is essential for the final image enhancement. In the multi-scale detail enhancement experiment, we compared the efficient edge preservation methods: BF [1], ROG [49] and our WGBF. Figure 17(b) and

![Figure 14](image14.png)  
**Figure 14**  Edge extraction of noisy image with noise level \( \sigma = 50 \). (a) Noisy image. (b) Image edge of Parrot without noise. (c) Edge extraction after using BF. (d) Edge extraction after using our proposed method

![Figure 15](image15.png)  
**Figure 15**  Edge extraction of complex texture image. (a) Original input. (b) Edge extraction from the original image. (c) Edge extraction after BF. (d) Edge extraction after our proposed filter

![Figure 16](image16.png)  
**Figure 16**  HDR image tone mapping. (a) Original input. (b) Base layer filtered by BF. (c) Base layer filtered by ROG. (d) Base layer filtered by WGBF.
(c) shows the images of coarse-scale detail and fine-scale detail enhanced by WGBF. Comparing the final enhanced images [see Figure 17(d) and (f)], the clear textures of petal edges is retained in the picture, based on image filtering by our WGBF. In contrast, as can be seen from the zoomed image, BF and ROG cannot more effectively preserve the the fine-scale details and result in low-scale textures to be blurred.

6 | CONCLUSION

Image filtering actually incorporates different topics, while existing research works tend to make improvements only on one specific direction, such as denoising, which makes it hard to be adapted into various types of practical usage. Our work attempts to generalise the task of image smoothing, by combining the image entropy and Gaussian filter in a conceptually simple way, where the proposed local entropy filter strives to protect the detailed information along with the Gaussian filter ensuring the smoothing effect. The adaptability and robustness of the bilateral filter inspire and enable us to realize a more generalised algorithm for smoothing out image interference. To verify the effectiveness of our method, we conduct extensive experiments on images with different kinds of interferences and compare our algorithm with methods aiming at different targets. We prove that our algorithm achieves similar or even better results than techniques designed for a specific task first, then we demonstrate the generalising ability of our method by using noisy textured images as inputs, where our algorithm outperforms the single target algorithms by a large margin. In addition, our designed intensity-aware window can not only improve bilateral filtering but also be applied to other local filters (e.g. Gaussian, mean and other window filters) for details preserving.

Although our method inherits the advantages of the bilateral filter, under this framework it also suffers from the problem of parameter regulation. In our future work, we would like to integrate the intensity-aware window into other excellent algorithms to simplify the tuning process and enable our method to better handle interference with different scales. Besides, we also intend to employ our smoothing algorithm into various types of applications, such as detail enhancement, image extraction and image decomposition.

ACKNOWLEDGEMENTS

This work was supported in part by The National Key Research and Development Program of China (2019YFB1707504, 2020YFB2010702), National Natural Science Foundation of China under Grant 61772267, Aeronautical Science Foundation of China (No. 2019ZE052008), and the Natural Science Foundation of Jiangsu Province under Grant BK20190016.

REFERENCES

1. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: Sixth International Conference on Computer Vision. pp. 839–846. IEEE Cat. No. 98CH36271. (1998)
2. D’Haeyer, J.P .F .: Gaussian filtering of images: A regularization approach. Signal Process. 18(2), 169–181 (1989)
3. Petschnigg, G., et al.: Digital photography with flash and no-flash image pairs. ACM Trans. Graphics 23(3), 664–672 (2004)
4. Buades, A., et al.: A review of image denoising algorithms, with a new one. Multiscale Model. Simul. 4(2), 490–530 (2005)
5. Chaudhury, K.N., Rithwik, K.: Image denoising using optimally weighted bilateral filters: A sure and fast approach. In: IEEE International Conference on Image Processing (2015)
6. Weiss, B.: Fast median and bilateral filtering. ACM Trans. Graphics 25(3), 519–526 (2006)
7. Tiwari, M., Gupta, B.: A consistent approach for image de-noising using spatial gradient based bilateral filter and smooth filtering. In: International Workshop on Pattern Recognition. (2016)
32. Polan, D.F., et al.: Tissue segmentation of computed tomography images.
31. Zhang, Y., et al.: An adaptive bilateral filter based framework for image denoising.
29. Biradar, N., et al.: Echocardiographic image denoising using extreme total variation bilateral filter. Optik - Int. J. Light Electron Opt. 127(1), 30–38 (2016)
28. Tiwari, M., Gupta, B.: A consistent approach for image de-noising using spatial gradient based bilateral filter and smooth filtering. In: International Workshop on Pattern Recognition. (2016)
26. Kottakki, K.K., et al.: Monte carlo Gaussian sum filter for state estimation. J. Visual Lang. Comput. 45, 23–24 (2018)
25. Li, J., Nehorai, A.: Gaussian mixture learning via adaptive hierarchical clustering technique for image denoising. Procedia Comput. Sci. 125, 132–142 (2018)
24. Wang, X.S.L.X., Qi, Jia., J.,: Rolling guidance filter. Comput. Vision ECCV Forum 34(7), 23–24 (2015)
23. Corbetta, M., et al.: Optimization of nonlinear, non-Gaussian bayesian filtering for diagnosis and prognosis of monotonic degradation processes. Mech. Syst. Sig. Process. 104, 305–322 (2018)
22. Zhao, F., et al.: Entropy-based bilateral filtering with a new range kernel. Signal Process. 93(2), 415–427 (2010)
21. Shanthi, S.A., et al.: Image denoising in hybrid wavelet and quincunx domain based on gaussian scale mixture model. Comput. Electr. Eng. 46(C), 384–393 (2015)
20. Raihan, C.S.D.: Particle Gaussian mixture filters(ii). In: 21st International Workshop on Pattern Recognition. (2016)
19. Raihan, D., Chakravorty, S.: Particle gaussian mixture filters(i). In: 21st International Workshop on Pattern Recognition. (2016)
18. Dabov, K., et al.: Image denoising by sparse 3-d transform-domain collaborative filtering. IEEE Trans. Image Process. 16(8), 2080–2095 (2007)
17. Buades, A., et al.: A non-local algorithm for image denoising. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition. (2005)
16. Gu, S., et al.: Weighted nuclear norm minimization with application to image denoising. In: Computer Vision and Pattern Recognition. (2014)
15. Thamulpelvan, P, Sathiaselvan, J.G.R.: A novel profuse clustering technique for image denoising. Procedia Comput. Sci. 125, 132–142 (2018)
14. Yan, H., et al.: Image decomposition combining a total variational filter and a Tikhonov quadratic filter. Multidimension. Syst. Signal Process. 26(3), 1–13 (2015)
13. Luisier, F., et al.: Fast interscale wavelet denoising of poisson-corrupted images. Signal Process. 90(2), 415–427 (2010)
12. Tao, D., et al.: Entropy-based bilateral filtering with a new range kernel. Signal Process. 137, 223–234 (2017)
11. Kaibiche, K., et al.: Restoration of stained old manuscripts via a hybrid wavelet and bilateral filtering system. J. King Saud Univ. - Comput. Inf. Sci. 29(4), 493–498 (2016)
10. Ming, Z., Gunturk, B.K.: Multiresolution bilateral filtering for image denoising. Neurocomputing 140, 299–316 (2014)
9. Chaudhury, K.N.: Acceleration of the shiftable o(1) algorithm for bilateral image denoising. Neurocomputing 164(C), 300–306 (2015)
8. Wang, Z., et al.: Image quality assessment: From error visibility to structural similarity. Trans. Image Proc. 13(4), 600–612 (2004)
7. Yin, H., et al.: Side window guided filtering. Signal Process. 165, 315–330 (2019)
6. Xu, P., Wang, W.: Improved bilateral texture filtering with edge-aware measurement. IEEE Trans. Image Process. 27(7), 3621–3630 (2018)
5. Chaudhury, K.N., Rithwik, K.: Image denoising using optimally weighted bilateral filters: A sure and fast approach. In: IEEE International Conference on Image Processing. (2015)
4. Tanchenko, A.: Visual-PSNR measure of image quality. J. Vis. Commun. Image Represent. 25(5), 874–878 (2014)
3. Wang, Z., et al.: Image quality assessment: From error visibility to structural similarity. Trans. Image Proc. 13(4), 600–612 (2004)
2. Tu, B., Xing, X., Xu, X.: Edge/structure preserving smoothing via relativity-of-gaussian. In: IEEE International Conference on Image Processing. (2018)
1. Wang, S.C., et al.: Fast o(1) bilateral filtering using trigonometric range kernels. IEEE Trans. Image Process. 20(12), 3376–3382 (2011)