

Research Article

Local Feature Filtering Method for Dynamic Multiframe Video Sequence Images

Dawei Zhang ¹ and Dan Huang ²

¹Telecom Department, Beihai Vocational College, Beihai 536000, China

²College of General Education, Beihai University of Art and Design, Beihai 536000, China

Correspondence should be addressed to Dan Huang; glncxcb@mail.glnc.edu.cn

Received 10 August 2022; Accepted 13 October 2022; Published 8 November 2022

Academic Editor: Tudor Barbu

Copyright © 2022 Dawei Zhang and Dan Huang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To improve the quality of local feature filtering for dynamic multiframe video sequence images, this study is aimed at designing an improved nontexture class noise filtering algorithm based on noise construction denoising algorithm and gray histogram of pixel points, and then designs a texture noise denoising algorithm based on texture smoothing processing and circular gradient values. The two algorithms are combined to propose a comprehensive filtering and denoising algorithm for horizontal dynamic video images. The experimental test results show that the normalized correlation coefficient, mutual information quantity, peak signal-to-noise ratio, and information entropy of the integrated filter denoising algorithm are 0.950, 0.935, 0.816, and 0.933 after convergence of the training effect, which are significantly higher than those of the commonly used median denoising algorithm and Kalman denoising algorithm. However, the computational time consumption of the proposed integrated filtering and denoising algorithm is higher than that of the comparison algorithms. The experimental results show that the integrated filtering algorithm for dynamic video images designed in this study can achieve better filtering and image reconstruction results in application scenarios with lower requirements for the timeliness of processing results.

1. Introduction

With the socioeconomic development, human beings are demanding higher and higher standards for the viewing and working quality of video images [1]. However, video shooting may result in various types of noise in the video due to camera shake, insufficient light, or uneven distribution, which affects the use and viewing of the video by users or consumers [2]. Therefore, it is particularly important to denoise video images, and this study innovatively classifies image noise into texture noise and nontexture noise and proposes a comprehensive filtering algorithm for the problem of denoising underwater video images, which is difficult for dynamic video filtering, where a denoising algorithm based on noise detection and pixel point gray scale histogram data are used to construct a method for dealing with nontexture noise, texture smoothing processing and improved circular gradient values are used to construct a method to deal with texture noise. Thus, an attempt is made

to improve the processing quality of underwater dynamic video images by reducing the filtering loss of the original information of the images, especially the core structure information while improving the denoising effect of underwater dynamic video images.

2. Related Works

With the rapid development of image processing computation, people demand higher and higher quality for processing noise in picture and video images, and in order to meet consumers' higher quality requirements for image-like data, a large number of computer experts, image processors, and professors of artificial intelligence-related majors at home and abroad have conducted a lot of academic research. Zhang et al. address the dimensional disaster and spatial background caused by high dimension in hyperspectral image classification. For the problem of underutilization of information, a new joint spectral-space classification method

based on edge-preserving filtering was proposed. The algorithm was validated on hyperspectral datasets from Pine Tree and University of Pavia, India. Under the same experimental conditions, the method achieves the highest classification accuracy and the lowest time consumption, showing significant advantages in hyperspectral image classification [3]. Landa et al. designed an adaptive processing method for large datasets to address the problem that image datasets are easily corrupted by noise and distortion. Test results showed that the convergence speed of the method proposed in this study was improved compared with common image denoising algorithms [4]. Cheng et al. proposed a new structure preserving retinal image filtering algorithm (SGRIF) to recover images based on the attenuation and scattering model to address the problem of retinal image quality degradation, which consists of a global structure transfer step and a global edge-preserving smoothing step. Test results show that the proposed SGRIF method is able to improve the contrast of retinal images. Furthermore in two applications, deep learning-based optic cup segmentation and sparse learning-based cup-to-disc ratio (CDR) calculation, the results showed that SGRIF processed images achieve more accurate image segmentation and CDR measurement [5]. Khan et al. explored content filtering techniques from the perspective of static images and analyzed content-based filtering that can help label the images as adult natural or safe images. As the proposed method is chromaticity based on skin segmentation and detection for undesirable content in images, the performance of the method was validated using a certain image dataset and found to be good for image information filtering [6]. Kong et al. proposed an improved cost aggregation method to solve the problem of adaptive cross-region guided image filtering (ACR-GIF) in which parallax is not considered, resulting in degradation of the image quality of the filtered output. The results of the evaluation platform show that the proposed cost aggregation method can significantly improve parallax accuracy with less additional time overhead compared to ACR-GIF, and the proposed stereo matching algorithm outperforms other state-of-the-art local and nonlocal algorithms [7]. Using linear relations and image filtering ideas, Yu et al. constructed an improved image encryption algorithm. Experimental results show that the improved algorithm not only inherits the advantages of the original algorithm but also improves the security robustness to differential cryptanalysis [8]. Qiao et al. designed an optimized SIFT feature extraction algorithm for highlighting stable edge corner point information and improving the acquisition efficiency of stable edge corner points. Experimental results show that the improved SIFT feature extraction algorithm based on image filtering can improve the extraction of stable feature points of edge response while suppressing the extraction of unstable feature points of edge response, thus improving the matching accuracy of images [9]. Chatterjee et al. performed a comparative analysis of three commonly used filtering techniques, i.e., Fourier filter, windowed Fourier filter, and wavelet filter to reduce noise and extract accurate phase information from phase-shifted interferograms. For this purpose, two basic types of noise (additive and multiplicative noise) are intro-

duced in the simulated interferogram and processed using a prefiltering strategy. The practical applicability of the analysis results was experimentally demonstrated and the results showed that high accuracy lens defocus error measurements were obtained using the filtering strategy [10].

Lv et al. discussed the digital holographic microscopy technology based on multiframe full field heterodyne technology and proposed a multiframe video sequence denoising algorithm based on time-frequency spectrum analysis technology. The experimental results show that the algorithm focuses on solving the twin image problem in multiframe video sequences and significantly reduces the random noise in video signals [11]. Ponomaryov et al. designed a new 3D data filter to filter the pulse noise in the color multiframe video sequence data. The input data of the algorithm is in the form of three primary color channels. The algorithm will calculate the fuzzy gradient value of the data from eight directions. The simulation results show that the algorithm has good filtering effect in different color multiframe video sequence data, and the calculation time is significantly lower than the current common filter [12]. Tsang et al. proposed a multiframe video sequence denoising algorithm that makes it difficult to use brute force to solve the problem of insufficient security in multiframe video sequence denoising [13]. The experimental results show that the probability of brute force cracking of multiframe video sequence data after denoising by this algorithm is significantly lower than that of the compared denoising algorithm [13].

In summary, various algorithms have been designed and improved to improve the quality of image denoising and filtering, but it is relatively rare to study the classification of noise according to its information characteristics so as to construct a variety of targeted algorithms combined into a denoising model, and this design approach combined with the idea of classification discussion can serve to improve the filtering effect of the algorithm on certain special types of images, which is the main idea of this study.

3. Video Sequence Image Local Multicategory Noise Integrated Filtering Method Design

3.1. Design of Local Feature Filtering Algorithm Based on the Pixel Difference of Adjacent Frames of Video. Dynamic multiframe video images where the subject is an underwater object contain several types of local noise, which can be classified according to the processing as texture noise generated by liquid flow, traditional nontexture noise due to insufficient or uneven distribution of light, poor performance of the camera equipment, etc. [14]. Now, we first design an algorithm to deal with nontextural noise, if an image has local noise, which can be expressed by Equation (1), the

$$g(x, y) = n(x, y) + f(x, y). \quad (1)$$

In Equation (1), $g(x, y)$ and $f(x, y)$ are the noise-contaminated image and the noise-free image, respectively. $n(x, y)$ is used to describe the noise function, and x, y is the arbitrary pixel coordinates of the noise region. Image denoising is the process of obtaining $f(x, y)$ backwards from

the captured $g(x, y)$ image and the computed $n(x, y)$ function. There are three main types of traditional algorithms to deal with nontextural noise in images, which are the median denoising algorithm, Kalman denoising algorithm, and noise detection-based denoising algorithm [15]. In the following, these three denoising algorithms are compared in order to select the most suitable algorithm for subsequent optimization design. The core logic of median denoising is to replace the pixel's own gray value using the median of the pixel's domain gray value, which is a nonlinear denoising method that works well for filtering speckle local noise and pretzel local noise due to the fact that it does not rely on extreme values in the domain [16]. For the median filter, it has little impact on the output of the original image information because the noise information is difficult if left unfiltered and can achieve relatively balanced results in terms of both noise removal and retention of key image information [17]. In addition, since usually noise points are formed by superimposing normal gray values with noisy gray values, which will be distributed in a random manner in any local area of the whole image, noise adjacent image frames also have up- and downshoot between them, resulting in noise gray values that are all extreme and differ significantly from the gray values of surrounding pixel points [18]. However, in general, the pixel gray levels in an image show a gradient rule, so if a median denoising filtering algorithm is used to deal with underwater nontextural noise, the algorithm will most likely see all the noise as pretzel noise [19]. The following describes how the median filter works using the 3×3 specification pixel map shown in Figure 1.

As we can see in Figure 1, the median denoising algorithm is to arrange the pixel values of Figure 1 in ascending order to obtain the median value, and then replace the pixel values in the middle of the 3×3 size image with the median value and delete the other pixel values [20]. The Kalman denoising algorithm is an efficient recursive filtering algorithm that can be used to detect Poisson noise points in a finite set of video sequences containing noise, and therefore is widely used in various engineering image processing [21]. Specifically, the running states in the Kalman denoising algorithm can be expressed by real vectors, and each time the discrete time increases, the linear operator will generate a new state with noise in the current state and add the control information of the controller [22]. Meanwhile, the output of these hidden states will be generated by another linear operator in the Kalman denoising algorithm. In summary, the operational logic process of the Kalman denoising algorithm can be described by Figure 2.

The third nontexture noise denoising method was chosen for this study because of its better scalability and denoising performance. The core idea of this algorithm is to identify normal signal points and noise points in each frame of the video image using some kind of metric, and then process the two pixel points in different ways. Specifically, the algorithm does not process the pixels identified as signal but outputs them directly, while the pixels identified as noise are processed differently depending on their location, which is the innovation of this study. In this study, in order to reduce the probability of false and missed detec-

tion of noise points and to improve the filtering quality [23]. It is necessary to first calculate the gray histogram data in the original image $h(g)$, g is the gray level, and the purpose is to obtain the full potential range of values of the threshold τT , which is calculated in Equation (2). If the difference between the pixel value at the same position between the original image and its vector frame is defined as $\Delta f_k(i, j) = g$, reassign $h(g) = h(g) + 1(g \in G)G$ as the range of gray level values, and then set to $[0, 255]$.

$$T = \{g \in G | h(g) > 0\}. \quad (2)$$

The Poisson smoothing of Equation (2) is performed using a rectangular window of size $W \times W$, and the value of this window is output x_n , which is calculated in the following:

$$x_n = \sum_{\sigma_i=0}^{w-1} \sum_{\sigma_j=0}^{w-1} \lambda [\Delta_k(i - \delta_{ij} - \delta_j)]. \quad (3)$$

In Equation (3), N represents the total number of samples, n is the sample number and $0 < n < N$, δ_{ij} is the Poisson smoothing coefficient of the location coordinates (i, j) pixels, and the coefficient λ method is calculated according to

$$\lambda = \begin{cases} 1, & \Delta_k(i - \delta_{ij} - \delta_j) > \tau, \\ 0, & \text{other.} \end{cases} \quad (4)$$

Using x_n , the computed variance S_T^2 and the computed mean \bar{x}_t can be found, and the threshold value needed for the algorithm is solved by these two indicators. If the relative rate of change $V_t = \bar{x}_t/S_T^2$, the algorithm threshold τ needs to be determined according to the criteria that make V_t reach the maximum, so that the image noise can be displayed to the maximum extent while ensuring that the overall image information is not damaged. After finding the threshold value τ , the difference between $\Delta f_k(i, j)$ and τ is used to determine whether a point in the image is noisy or not. Then, define a marker matrix, which has the same dimension and size as the image to be tested. And use $m(i, j)$ to represent the value of each pixel point in the original image, with 0 and 1 to represent the test location without noise, the presence of noise, respectively, the matrix initially needs to be initialized as a zero matrix. By comparing the test result with the threshold value, it is judged what the value of this position in the marker matrix should be. After the end of the judgment, then only the judged noise points can be processed, and the processing method is as follows: if the number is less than 9, the pixel value of the last processed point is used to replace the noise point to be processed. If the number is greater than 9, a selective median filtering is started, i.e., only the pixel values classified as noise information are replaced. According to this process, the calculation flow chart of the denoising algorithm based on noise detection is shown in Figure 3.

3.2. Design of Texture Noise Filtering Algorithm Combining Texture Smoothing Suppression and Circular Gradient. The median denoising algorithm, Kalman denoising algorithm,

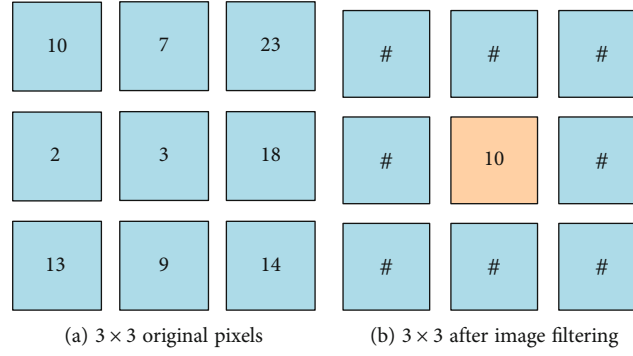


FIGURE 1: Schematic diagram of how the median denoising algorithm works.

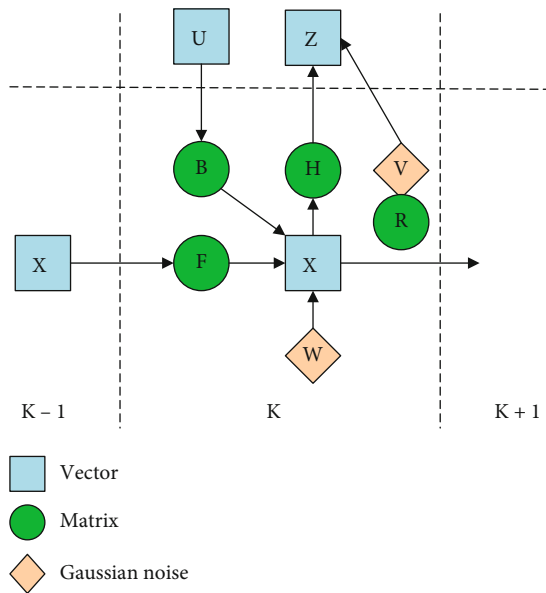


FIGURE 2: Flow chart of the Kalman denoising algorithm.

and noise detection-based denoising algorithm have poor filtering effect on texture noise or lose too much core information of the image due to their own algorithm characteristics. Therefore, this study proposes a filtering algorithm combining texture smoothing suppression and circular gradient for processing texture noise after filtering by noise detection-based denoising algorithm, and its computational flow is shown in Figure 4.

Observing Figure 4, we can see that the algorithm mainly consists of three parts: improved circular gradient value, texture smoothing suppression, and image reconstruction, and the following is a detailed analysis of the computational process content of the algorithm. The role of circular gradient is to reduce the effect of the filtering algorithm on the deletion and dumping of the core information of the image. To design the calculation method of circular gradient, define Ω_p as the central local window of pixel p and I as the input signal of one-dimensional form, then the interval gradient operator of interval gradient algorithm (IG for short) is defined, see the following:

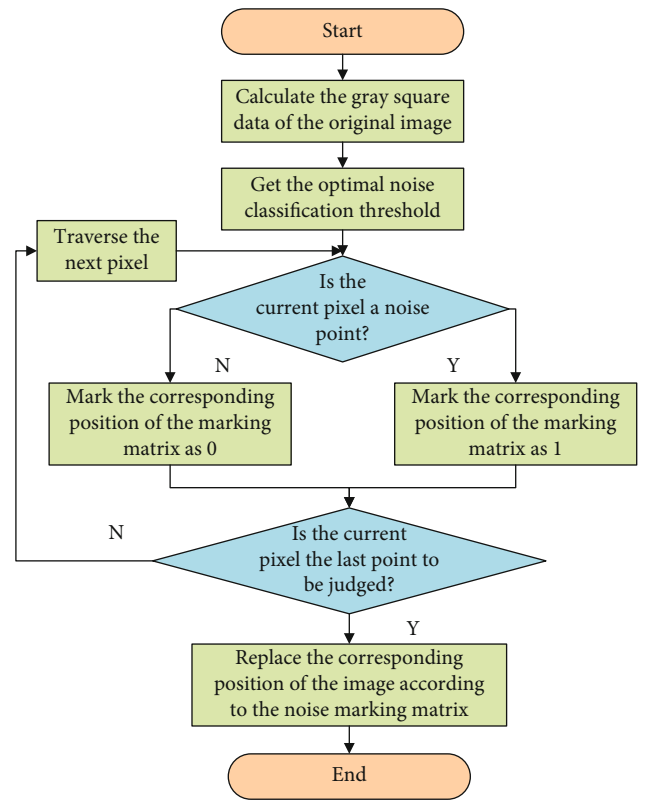


FIGURE 3: Flow chart of noise detection based denoising algorithm.

$$(\nabla_{\Omega} I)_p = g_{\sigma}^r(I_p) - g_{\sigma}^l(I_p). \quad (5)$$

In Equation (5), g_{σ}^r and g_{σ}^l represent the left and right one-dimensional Gaussian filter functions, respectively, whose calculation methods are more common and will not be repeated here. The interval gradient is different from the logic of the traditional gradient calculation, in that the former calculates the color-weighted mean difference located on both sides of the pixel p . Moreover, for structural and textured pixels, the interval gradient can serve to amplify the gradient and cancel the intrablock gradient, respectively, indicating that the operator also has the function of distinguishing image texture. However, since the operator

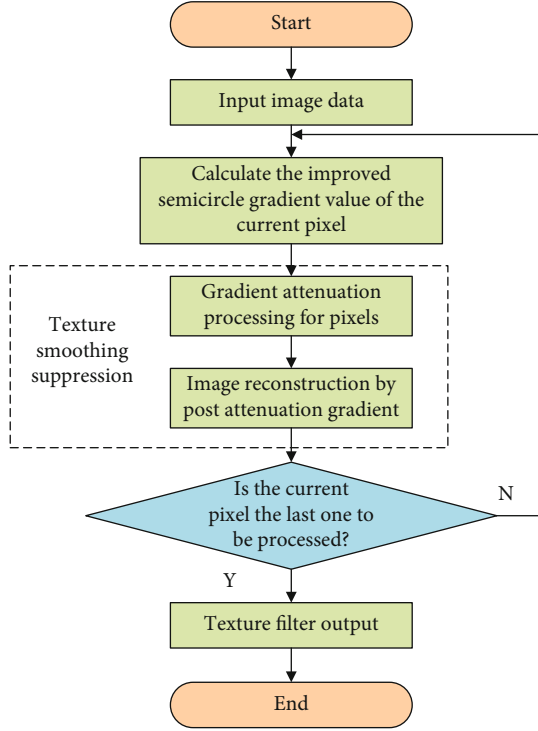


FIGURE 4: Flow of filtering algorithm combining texture smoothing suppression and circular gradient.

calculates the signal data arranged along two axes, respectively, the algorithm can only take into account the local information of the image only. To further improve the texture noise differentiation ability of the algorithm, the interval gradient operator is now improved and the circular gradient operator is designed. With I representing the input image, the formula of circular gradient calculation is shown in the following:

$$u(p) = g_r(R_p) - g_l(L_p), \quad (6)$$

where L_p and R_p represent a circular window located to the left and right of the pixel p , with p as the center and k as the radius. And g_l and g_r are the Gaussian-weighted mean values in the left and right windows, respectively.

By using Equation (6), we can calculate the values of the circular gradient of the image in the direction of x , y , u_x , and u_y . However, in the actual application scenario, the structure direction in the image is mostly uncertain, so we cannot use a fixed and common structure direction, and this research continues to improve the method of determining the structure direction in the image. The method of finding the main direction of the image structure is chosen, and an attempt is made to calculate the circular gradient under the main direction to achieve the goal of increasing the Gaussian and weighted mean difference between the two sides of the structure. The improved directional circular gradient v is calculated according to the following:

$$v(p) = g_r(\phi(R_p)) - g_l(\phi(L_p)), \quad (7)$$

where $\phi(L_p)$ and $\phi(R_p)$ are the left arc and right arc windows after rotation, respectively, and the rotation angle α_p is obtained by the inverse tangent of the directional gradient of y and x dimensions. The rotation angle is also the main direction of the structure of the pixel p . Using α_p , we can calculate the directional gradient of the circular arcs in y and dimensions x , v_y , and v_x and finally calculate the root mean square value of v_y and v_x to get the directional circular arc gradient magnitude of the image v .

Then the texture noise filtering algorithm in the texture smoothing suppression processing method is designed, in order to get the image data with strong gradient texture suppression characteristics, the gradient value of the image needs to be attenuated based on the normalized directional circular gradient magnitude value, and the processing method is shown as

$$G_p = v_p \cdot \nabla I_p, \quad (8)$$

where ∇I_p is the input image gradient of pixel p , G_p is the gradient value of pixel p , and v_p is the directional circular gradient amplitude of pixel p after normalization process. The attenuated gradients are then used to reconstruct the partially filtered image and output a texture suppressed image that has a texture pixel gradient value lower than the structural pixel gradient. The process of reconstructing the image is done specifically by minimizing the objective function $E(J)$, as shown.

$$\min J E(J) = \min J \sum (J - I)^2 + \eta (\nabla J - G)^2. \quad (9)$$

At this point, the image reconstruction problem is transformed into a function optimization problem, and η is the weight coefficient of the control gradient. Equation (9) can be mapped into the frequency domain by the fast Fourier transform, thus speeding up the solution process.

$$J = F^{-1} \left(\frac{F(I) + \eta \left(\overline{F(\partial_x)} F(G_x) + \overline{F(\partial_y)} F(G_y) \right)}{F(1) + \eta \left(\overline{F(\partial_x)} F(\partial_x) + \overline{F(\partial_y)} F(\partial_y) \right)} \right). \quad (10)$$

In Equation (10), F^{-1} and $\overline{F(\cdot)}$ are the discrete Fourier inverse transform operator and the complex conjugate operator, respectively, and $F(1)$ is the δ function after the discrete Fourier transform. Considering that the gradient of the structural pixels of the image will be attenuated by the texture gradient suppression operation, the gradient minimization method with gradient boosting effect is chosen to filter the reconstructed image. The purpose of the gradient minimization process is to control the maximum number of gradient changes in the output image. Let S be the output of the filtering algorithm, and the gradient of the processed output image is described using the symbol ∇S , then there exists $\nabla S_p = [\partial_x S_p \ \partial_y S_p]^T$, and the objective function of gradient minimization can be expressed by

$$\min_S |S - J|^2 + \lambda |\nabla S|_0. \quad (11)$$

In Equation (11), $|S - J|^2$ is used to ensure that the input and output image structures are roughly the same, $| \cdot |_0$ represents the 0th parity, $|\nabla S|_0$ is the smoothing item, and λ represents the smoothing factor, the larger the value, the smoother the image. After introducing the auxiliary variables β , γ , and λ , Equation (11) can be rewritten as

$$\min_{S, \delta} |S - J|^2 + \beta |\nabla S - \gamma|^2 + \lambda |\gamma|_0. \quad (12)$$

In Equation (12), β is used to control the optimization speed of the objective function in terms of the similarity between the auxiliary variable γ and the corresponding gradient ∇S . Equation (12) can be solved by using the alternating variable decomposition method, which requires fixing S first in order to find the auxiliary variable γ , as calculated in the following:

$$\min \left| \nabla S - \gamma \right|^2 + \frac{\lambda}{\beta} |\gamma|_0. \quad (13)$$

Decompose Equation (13) into a set of univariate functions in space, and each δ_i can be approximated by replacing it according to the following:

$$\gamma_i = \begin{cases} 0, & \nabla S_i < \sqrt{\lambda/\beta}, \\ \nabla S_i, & \nabla S_i \geq \sqrt{\lambda/\beta}. \end{cases} \quad (14)$$

Then, fix γ and then solve for the gradient S .

$$\min_{S, \delta} |S - J|^2 + \beta |\nabla S - \gamma|^2. \quad (15)$$

Equation (15) can be minimized by the gradient descent method, and in this solution process, the output image S will be close to γ . From the calculation process of the gradient minimization method, it can be seen that the structures with small gradients in the image will also be smoothly suppressed as the algorithm runs, while the strong gradient textures cannot be effectively suppressed. Therefore, when texture noise should be processed, the processing map where the texture is suppressed should be obtained, and then the gradient minimization filtering process should be performed on this data. In summary, the computational flow of the dynamic underwater multiframe video image local feature filtering model designed in this study is shown in Figure 5.

4. Performance Verification of Video Local Multicategory Noise-Integrated Filtering Method

In order to verify the filtering performance of this research design's video local multicategory noise-integrated filtering method, noise removal experiments were carried out. The data set in the test was obtained from 247 groups of 80 frames of dynamic video shot at different locations underwa-

ter in continuous time, but the shooting position of each group in the video did not move, and each group of video contained a dynamic video with various noises and a video without noise, and the noise in the image was mainly artificially added Gaussian white noise and texture noise brought by the internal liquid flow, both in addition to the noise element, other image information are identical. In the experiments, the median denoising algorithm and the Kalman denoising algorithm are used as control algorithms, and the algorithm logic is programmed in Python. In order to more accurately evaluate the effect of denoising filtering and image restoration of each algorithm, five objective indicators, namely, correlation coefficient, mutual information, peak signal-to-noise ratio, information entropy, and computation time were chosen to evaluate the denoising effect or operation efficiency of the model. After finishing the simulation experiments, the correlation coefficient indexes of each algorithm were first statistically analyzed, as shown in Figure 6.

As shown in Figure 6, the horizontal axis represents the number of iterations of each algorithm, and the vertical axis represents the normalized correlation coefficient of the results of each algorithm run. As we can see in Figure 6, the normalized correlation coefficients of each algorithm show a trend of first growth and then convergence as the number of iterations increases. However, the simplest median filtering algorithm almost completes convergence when the number of iterations is around 30, and the convergence speed is the fastest. The normalized correlation coefficients of the median denoising algorithm, Kalman denoising algorithm, and the integrated filtering and denoising algorithm designed in this study are 0.861, 0.896, and 0.950, respectively, which show that the image processed by the integrated filtering and denoising algorithm designed in this study has the highest correlation with the source image and retains the most information of the source image. The mutual information metrics of each algorithm are then counted and shown in Figure 7.

The vertical axis in Figure 7 represents the normalized mutual information amount, and observation of Figure 7 shows that with the increase of the number of iterations, the change trend of the normalized mutual information amount and the final size ranking of each algorithm are consistent with the conclusion of Figure 6. After the convergence of each algorithm, the normalized mutual information amount of the integrated filtering and denoising algorithm designed in this study has the largest value of 0.935, which again proves that the dynamic video image processed by this algorithm retains the most information of the source image. Next, the statistical results of the peak signal-to-noise ratio data for each algorithm are analyzed and shown in Figure 8.

The vertical axis in Figure 8 represents the normalized peak signal-to-noise ratio. Analysis of Figure 8 shows that the normalized peak signal-to-noise ratio of each algorithm fluctuates more when the number of algorithm iterations is small, and there is no significant pattern in the fluctuation direction. With the growth of the number of iterations, the normalized peak signal-to-noise ratio of each algorithm tends to be stable. When the number of iterations is 100, the normalized peak S/N ratios of the median denoising

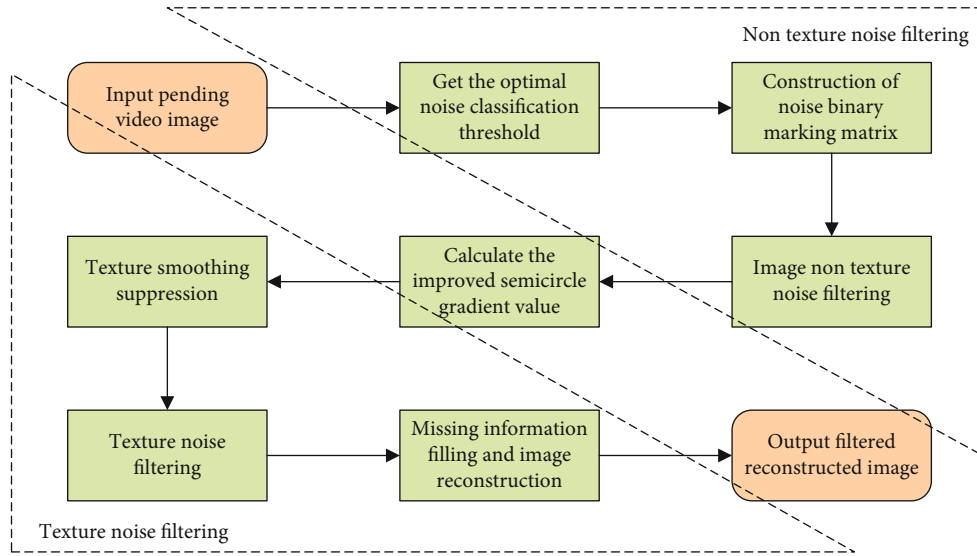


FIGURE 5: Calculation flow of local feature filtering model for dynamic underwater multiframe video images.

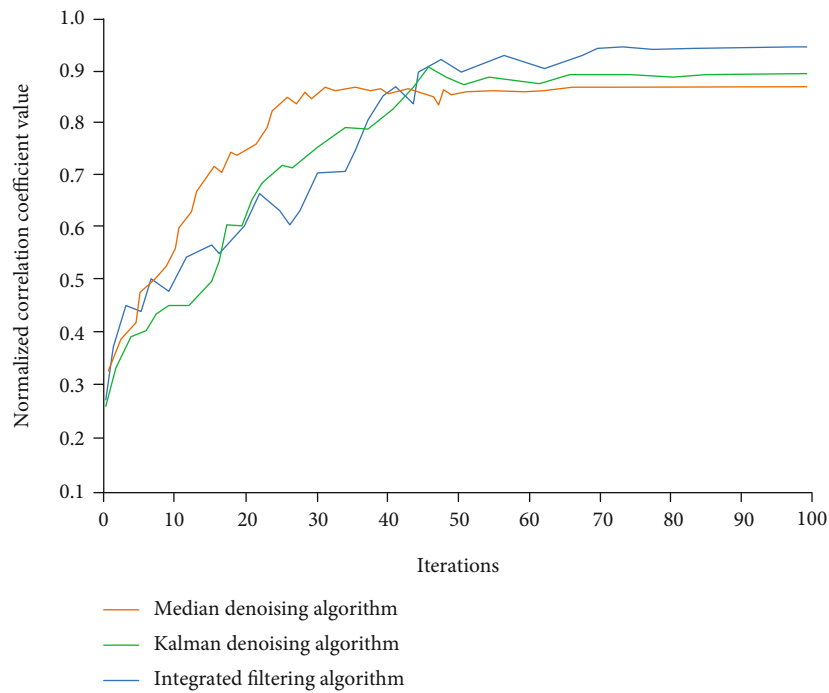


FIGURE 6: Statistical results of correlation coefficients of each filtering algorithm.

algorithm, Kalman denoising algorithm, and the integrated filtering and denoising algorithm designed in this study are 0.816, 0.797, and 0.718, respectively, which shows that the integrated filtering and denoising algorithm designed in this study has the best image denoising recovery effect and the least image distortion. The statistical results of the information entropy data of each algorithm are then analyzed and shown in Figure 9.

The left vertical axis in Figure 9 represents the normalized information entropy data, and the dots in the figure are labeled with the highest normalized information entropy

values that have occurred in the training process of each algorithm. As we can see in Figure 9, with the increase of the number of iterations, the normalized information entropy of each algorithm shows a trend of first growth and then convergence. The normalized information entropy values of the median denoising algorithm, Kalman denoising algorithm, and the integrated filtering and denoising algorithm designed in this study are 0.848, 0.894, and 0.933, respectively. The peak signal-to-noise ratio of has high consistency. Finally, the computational time consumed by each group of algorithms to process different numbers of

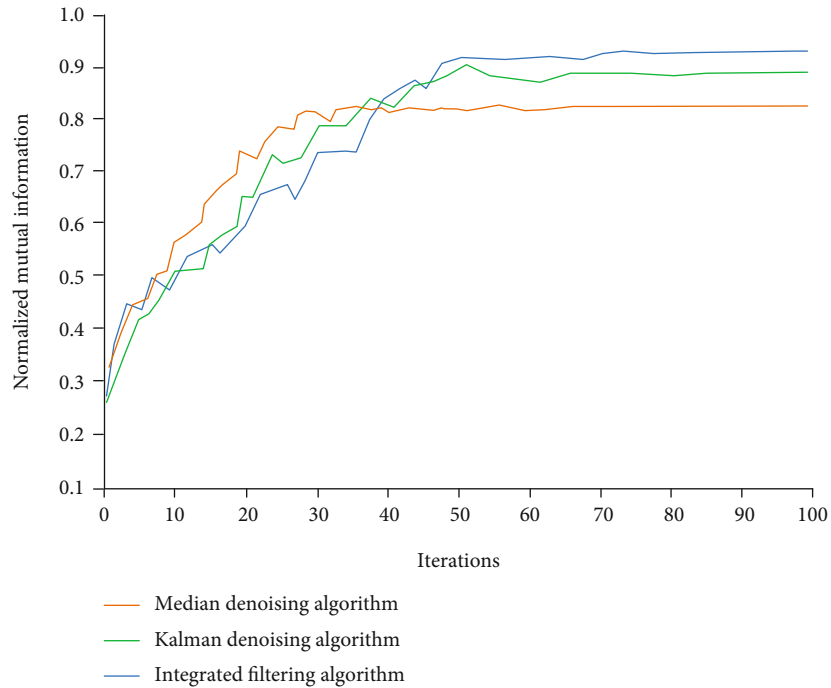


FIGURE 7: Statistical results of mutual information quantity of each filtering algorithm.

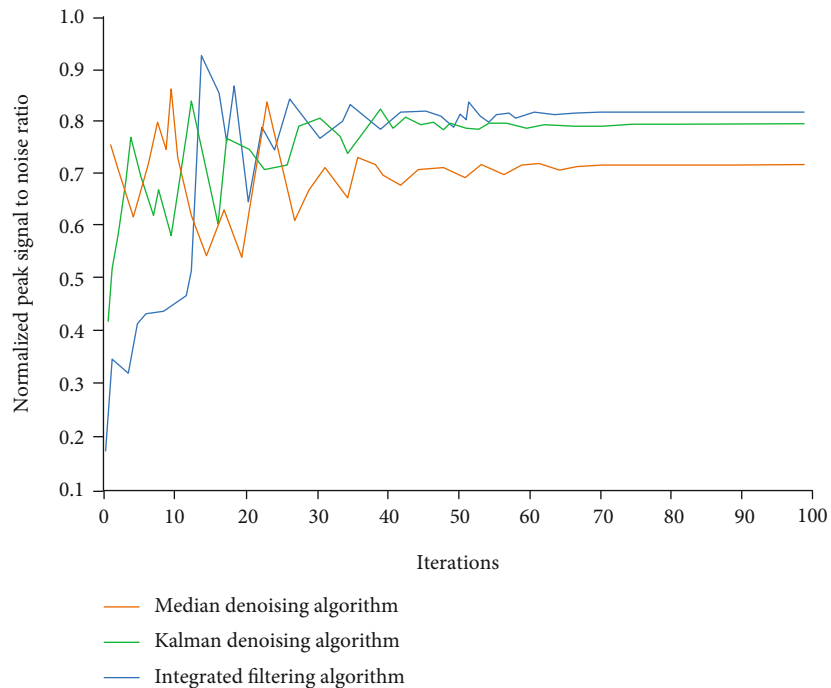


FIGURE 8: Statistical results of peak signal-to-noise ratio of each filtering algorithm.

dynamic videos after the performance convergence is counted, which is shown in Table 1.

As can be seen in Table 1, the computation time of the integrated filtering algorithm is the longest when processing three different numbers of samples, followed by the Kalman denoising algorithm and the median denoising algorithm has the shortest computation time overall. Specifically, when

the number of samples processed is 200, the average computation time of the median denoising algorithm, Kalman denoising algorithm and the integrated filtering, and denoising algorithm designed in this study are 1283 ms, 1519 ms, and 1737 ms, respectively, and the computation standard deviation of the three algorithms are 432 ms, 451 ms, and 683 ms, respectively. This study designs the integrated

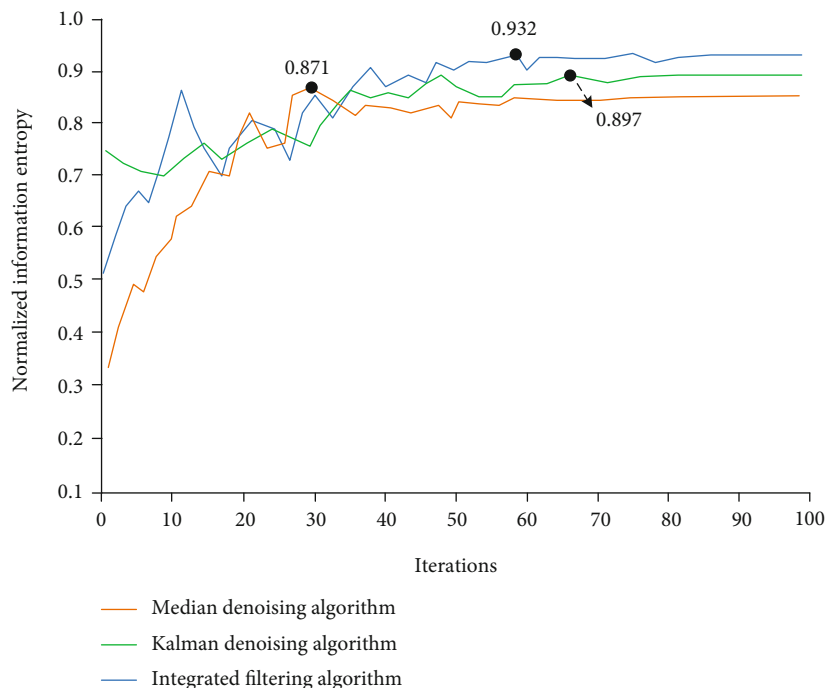


FIGURE 9: Statistical results of information entropy of each filtering algorithm.

TABLE 1: Comparison of computational time consumption of each filtering algorithm.

Filtering algorithm	Number of samples/ group processed	Minimum elapsed time/ms	Maximum elapsed time/ms	Average elapsed time/ms	Standard deviation/ms
Integrated filtering and denoising algorithm	10	84	153	114	38
	50	344	748	563	196
	200	1189	2449	1737	683
Median denoising algorithm	10	27	51	42	12
	50	156	335	216	117
	200	746	1608	1283	432
Kalman denoising algorithm	10	57	127	91	45
	50	253	520	394	160
	200	892	1961	1519	451

filtering algorithm to divide the noise into texture noise and nontexture noise, and then designs a set of algorithms separately and combines the two sets of algorithms to perform integrated denoising of images, which leads to a significant increase in computational complexity so it has the highest computational time consumption when processing the same number of samples.

5. Conclusion

To address the problem of local feature filtering reconstruction of dynamic multiframe video sequence images taken underwater, this study was aimed at designing a comprehensive denoising filtering algorithm fused with noise detection and underwater texture detection, and the designed algorithm is used with the comparison algorithm to process a copy of dynamic video image noise taken underwater. The experi-

mental results show that the normalized correlation coefficients and the mutual information quantity values of each filtering algorithm increase with the number of iterations and then converge. 0.826, 0.887, and 0.935, which shows that the filtering algorithm designed in this study can retain more information structure of the source image under the condition of image denoising. When the number of iterations is 100, the normalized peak signal-to-noise ratio and information entropy of the median denoising algorithm, Kalman denoising algorithm, and integrated filter denoising algorithm are 0.718, 0.797, and 0.816, and 0.848, 0.894, and 0.933, respectively, indicating that the designed filtering algorithm has the highest overall image quality after denoising. However, due to the limitations of research conditions, no more different kinds of underwater dynamic video images can be collected, especially video images with particularly serious noise or poor light conditions to verify the algorithm performance, which is also the scope of future research.

Data Availability

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest

It is declared by the authors that this article is free of conflict of interest.

Acknowledgment

The research is supported by the Guangxi University Middle-Aged and Young Teachers' Basic Research Ability Improvement Project in 2020 "Application Research of Data Mining Technology in University Library Services" (2020KY65009).

References

- [1] Y. Zhang, D. Li, and W. P. Zhu, "Infrared and visible image fusion with hybrid image filtering," *Mathematical Problems in Engineering*, vol. 2020, no. 4, Article ID 1757214, p. 17, 2020.
- [2] B. Meyzia, M. Hamdi, and R. Amelia, "Imaging analysis of thresholding image filtering, brain abnormalities morphology, and dose report CT scan records," *Journal of Physics: Conference Series*, vol. 1655, no. 1, article 012155, 2020.
- [3] C. K. Zhang and M. Han, "Spectral-spatial joint classification of hyperspectral image with edge-preserving filtering," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 44, no. 2, pp. 280–288, 2018.
- [4] B. Landa and Y. Shkolnisky, "The steerable graph laplacian and its application to filtering image datasets," *SIAM Journal on Imaging Sciences*, vol. 11, no. 4, pp. 2254–2304, 2018.
- [5] J. Cheng, Z. Li, Z. Gu, H. Fu, D. W. K. Wong, and J. Liu, "Structure-preserving guided retinal image filtering and its application for optic disc analysis," *IEEE Transactions on Medical Imaging*, vol. 37, no. 11, pp. 2536–2546, 2018.
- [6] R. U. Khan and A. Alkhalifah, "Media content access: image-based filtering," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 3, pp. 415–419, 2018.
- [7] L. Kong, J. Zhu, and S. Ying, "Local stereo matching using adaptive cross-region-based guided image filtering with orthogonal weights," *Mathematical Problems in Engineering*, vol. 2021, Article ID 5556990, 20 pages, 2021.
- [8] F. Yu, X. Gong, H. Li, and S. Wang, "Differential cryptanalysis of image cipher using block-based scrambling and image filtering," *Information Sciences*, vol. 554, no. 4, pp. 145–156, 2021.
- [9] M. Qiao, X. Liang, and M. Chen, "Improved SIFT algorithm based on image filtering," *Journal of Physics: Conference Series*, vol. 1848, no. 1, pp. 012069–012074, 2021.
- [10] A. Chatterjee, P. Singh, J. Dhanotia, V. Bhatia, and S. Prakash, "Comparative analysis of image pre-filtering techniques for shifted noise affected interferograms," *Applied Optics*, vol. 59, no. 10, pp. 3227–3236, 2020.
- [11] X. Lv, B. Xiangli, W. Zhang et al., "Multiframe full-field heterodyne digital holographic microscopy," *Chinese Optics Letters*, vol. 14, no. 5, pp. 050901–050904, 2016.
- [12] V. I. Ponomaryov, H. Montenegro, A. Rosales, and G. Duchon, "Fuzzy 3D filter for color video sequences contaminated by impulsive noise," *Journal of Real-Time Image Processing*, vol. 10, no. 2, pp. 313–328, 2015.
- [13] P. Tsang, W. K. Cheung, and T. C. Poon, "Fast numerical generation and hybrid encryption of a computer-generated Fresnel holographic video sequence," *Chinese Optics Letters*, vol. 11, no. 2, pp. 020901–020905, 2013.
- [14] H. Sahloul, S. Shirafuji, and J. Ota, "3D affine: an embedding of local image features for viewpoint invariance using RGB-D sensor data," *Sensors*, vol. 19, no. 2, p. 291, 2019.
- [15] S. Agarwal and K. H. Jung, "HSB-SPAM: an efficient image filtering detection technique," *Applied Sciences*, vol. 11, no. 9, pp. 3749–3765, 2021.
- [16] S. Rathi and S. Bamal, "Deep learning and globally guided image filtering technique based image dehazing and enhancement," *International Journal of Technical Research & Science*, vol. Special, no. Issue3, pp. 60–68, 2020.
- [17] A. Kumar and S. Raheja, "Edge detection using guided image filtering and enhanced ant colony optimization," *Procedia Computer Science*, vol. 173, pp. 8–17, 2020.
- [18] B. Desai, U. Kushwaha, and S. Jha, "Image filtering -techniques, algorithm and applications," *Applied GIS*, vol. 7, no. 11, pp. 101–106, 2020.
- [19] I. Banerjee, C. Kurtz, A. E. Devorah, B. Do, D. L. Rubin, and C. F. Beaulieu, "Relevance feedback for enhancing content based image retrieval and automatic prediction of semantic image features: application to bone tumor radiographs," *Journal of Biomedical Informatics*, vol. 84, no. 8, pp. 123–135, 2018.
- [20] A. Gattal, C. Djeddi, I. Siddiqi, and Y. Chibani, "Gender classification from offline multi-script handwriting images using oriented basic image features (oBIFs)," *Expert Systems with Applications*, vol. 99, pp. 155–167, 2018.
- [21] X. Wu and D. Qiu, "Digitalization system of ancient architecture decoration art based on neural network and image features," *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 2, pp. 2589–2600, 2021.
- [22] K. Picos, V. H. Diaz-Ramirez, A. S. Montemayor, J. J. Pantrigo, and V. Kober, "Three-dimensional pose tracking by image correlation and particle filtering," *Optical Engineering*, vol. 57, no. 7, article 073108, 2018.
- [23] J. H. Chen, Z. J. Luo, Z. H. Zhang et al., "Polar transformation on image features for orientation-invariant representations," *IEEE Transactions on Multimedia*, vol. 21, no. 2, pp. 300–313, 2019.