Eng Int Syst (2021) 4: 195–204

Engineering
Intelligent Systems

Noise Filtering Method of a 3D Target Image Based on Machine Learning

Yanfei Ren*

Department of Mathematics and Information Engineering, Puyang Vocational and Technical College, Puyang 457000, China

In order to effectively improve the quality of three-dimensional target images, this study designed a method for the noise filtering of three-dimensional target images based on machine learning. Firstly, the 3D target image is represented digitally. On this basis, the image data structure, which is convenient for computer processing, is formed through the process of sampling and quantification. Then, additive white gaussian noise and random value impulse noise are combined to form mixed noise, and the model of mixed noise is established. Through the detection process and denoising process, combined with machine learning, the noise of the 3D target image is filtered out. The experimental results show that the PSNR and SSIM values of 3D target images are ideal after the processing of the filter method based on machine learning, and the image texture complexity value is high, which fully demonstrates the effectiveness of this method.

Keywords: Machine learning, Three-dimensional target image, Mixed noise, Filtering, Digital

1. INTRODUCTION

Images are obtained by observing the objective world in different forms and means through various observation systems and can, directly or indirectly, act on the human eye and produce visual perception. As image is the visual basis of human perception of the world, it is an important means for humans to obtain, express and transmit information [1]. Scientific research and statistics show that about 75 percent of the information we get comes from the visual system. Images can be classified into visible images and invisible images according to their forms or production methods. According to the spatial co-ordinates of the image and the degree of light and shade, continuity can be divided into a simulation image and a 3D target image.

Image processing was first used in newspapers when images were sent from London to New York via submarine cables [2]. The development of 3D target image processing technology is closely related to the development of computer technology. With the development of computer technology, large-scale storage and display systems have been developed, which are the basis of image processing.

The first large computers capable of performing image-processing tasks appeared in the early 1960s. During this period, with the use of computers and the development of space projects, people gradually began to study image processing technology. In 1964, at the Jet Propulsion Laboratory in California, computer technology was used to improve images from space probes [3]. Images of the moon transmitted by the Voyager 7 satellite were processed by a computer to correct for the various types of image distortions in the spacecraft's television cameras.

In addition to medical and space applications, image processing techniques are now used in a wide range of applications. In archaeology, for example, image processing can be used to successfully recover blurred images, which are the only surviving records of rare items that have been lost or damaged [4]. In physics and related fields, computer technology often enhances experimental images in fields such

*Email of corresponding author renyanfei08@163.com

as high-energy plasma and electron microscope methods. In addition, image processing technology has been successfully applied in the fields of astronomy, biology, nuclear medicine, law enforcement, national defense and industry.

In the field of image processing, 3D object image processing technology is an important branch. In general, 3D object image processing technology can be used to deal with automatic character recognition, production line detection, military identification, automatic fingerprint processing, X-ray and blood sample classification processing, weather forecasting, environmental identification among many other aspects [5].

After obtaining the 3D target image, the image needs to be processed preliminarily, including image enhancement, image restoration, image compression, image segmentation and object recognition. In the process of image acquisition or transmission, the image quality inevitably declines (degrades) due to the imaging system, transmission media and other related causes. Some of the many reasons for image degradation are:

(1) Image distortion caused by aberration, distortion and limited bandwidth of the imaging system.

(2) Geometric distortion caused by unstable movement of aircraft or satellites carrying remote sensing instruments and rotation of the earth.

(3) In the process of digitization, parts of details will be lost, resulting in image quality degradation.

(4) Motion blur caused by the relative motion between the camera and the subject during shooting.

(5) Defocus blur caused by lens aggregation is not allowed.

Noise interference is one of the most important causes of image quality degradation. Once the noise is included in the image, it will inevitably affect the visual effect of the image to some extent, and even hinder normal recognition in serious cases. Therefore, noise filtering of 3D target images plays a very important role in image processing, and the results after noise filtering have a direct impact on image edge detection, image segmentation, feature extraction, image recognition and other subsequent processing [6].

Until now, the research of 3D target image filtering mainly focuses on the processing of gray images. In recent years, with the continuous improvement of computer performance, multimedia technology, and especially color imaging equipment, color images have become much more widely used. As opposed to a gray image, each pixel value in a color image is composed of multiple component values. For example, in a RGB image, each pixel value is composed of three component values, that is, a three-dimensional vector. Therefore, compared with a gray image, a color image is far more complex [7]. However, a color image can provide more image information than a gray image, so color image processing has received far more attention, gradually become an important research topic. In recent years, great achievements have been made in noise filtering of color images, and many effective filtering methods have been proposed. However, in practical application, these algorithms have different limitations, so it is necessary to research new image filtering methods and improve existing methods.

To this end, this study considers the introduction of machine learning technology in order to design a new 3D target image noise filtering method.

2. DESIGN OF 3D TARGET IMAGE NOISE FILTERING METHOD BASED ON MACHINE LEARNING

2.1 Digital Representation of Three-Dimensional Target Image

Generally, due to the different ways of representation, 3D target images can be divided into continuous images and discrete images; they can also be referred to as analog 3D target images and digital 3D target images. In everyday life, for example, a photograph recorded on chemical film simulates a three-dimensional object image, while an image taken with a digital camera is a digital three-dimensional object image.

More specifically, the simulated 3D target image refers to the image with infinite density of image points and infinite density of gray value in the 2D coordinate system. In other words, the change of the position of the pixels in the horizontal and vertical directions of the continuous image and the change of the gray value in the position of each pixel are continuous. We can consider that the simulated 3D target image is composed of an infinite number of pixels, and the gray value at each point has an infinite number of possible values. The simulated three-dimensional target image reflects the continuous change of the brightness and color of the objective object with the change of spatial position and direction.

A digital three-dimensional object image is an image formed by the reflection or transmission of light rays. Thus, a digital three-dimensional object image can be regarded as a collection of light intensity of each point in space [8].

For the three-dimensional target image, we can simply regard the light intensity as a continuous function changing with spatial co-ordinates (x, y, z), light wavelength λ and time t, and its mathematical expression is as follows:

$$I = F(x, y, z, \lambda, t) \tag{1}$$

If only the energy of light is considered and the wavelength is not considered, then the three-dimensional target image is visually represented as a gray image, which can be called gray image. The image function is:

$$I_a = F(x, y, z, t) \tag{2}$$

If the grayscale image processed is a still image, that is, the content of the image does not change with time, the mathematical expression of the three-dimensional target image is then:

$$I_b = F(x, y, z) \tag{3}$$

Since the information that can be processed by current computers must be digital signals, and the original information such as photos, drawings and scenes obtained by us are all continuous analog signals, the first step in the processing of 3D target images is to convert the information expressed in the continuous 3D target images into a digital form. In other words, the continuous changing image points in the two-dimensional coordinate system need to be discretized, that is the whole image is divided into the pixels of countless small rectangular regions, and the values representing the brightness and darkness need to be discretized. The luminance or color value space of a pixel should be discretized into a quantized number of finite values, and the image information should be represented numerically. In order to realize the digital processing of the image and obtain a three-dimensional target image, it is necessary to sample and quantify the continuous image, that is, to discretize the space, direction and brightness of the continuous image [9]. Reasonable image sampling and quantization are two important factors related to whether the digital image is close to the original image, and also related to the amount of digital image information.

If the three-dimensional target image is sampled with equal spacing and a lattice with row and column distribution is obtained, then every point in the matrix corresponds to an element in the three-dimensional target image, which is called the pixel point [10]. The correct selection of sampling points and sampling interval is very important, as this will directly affect the quality of the sampled 3D target image, such as the distortion degree between the 3D target image and the original image. Take a one-dimensional signal $G(t)$ as an example; if the maximum frequency of one-dimensional signal is ω, then according to the sampling theorem, we can obtain:

$$G(t) = \sum_{i=-\infty}^{\infty} G(iT)s(t - iT) \tag{4}$$

where: T represents the sampling period, $i = \cdots, -2, -1, 0, 1, 2, \ldots$.

$$s(t) = \frac{\sin(2\pi\omega t)}{2\omega t} \tag{5}$$

Then, by sampling at an interval of $T \le \frac{1}{2\omega}$, $G(iT)$ can be fully recovered to $G(t)$ according to the sampling results.

Similarly, the three-dimensional sampling theorem can be obtained. Suppose the three-dimensional function $f(x, y, z)$ is a spatially continuous analog image function, then the following formula exists:

$$f(x, y, z) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f(mT, nT)s(x - mT, y - nT) \tag{6}$$

where: $m \in i, n \in i$.

On the premise that the number of samples is consistent, the sampling process can also be divided into uniform sampling and non-uniform sampling. In the process of sampling the three-dimensional target image, if the horizontal and vertical directions are equally spaced, it is called uniform sampling, otherwise, it is called non-uniform sampling.

Non-uniform sampling refers to sampling at different intervals in different regions of the three-dimensional target image according to the specific situation of the three-dimensional target image. The selection of the non-uniform sampling interval should be based on the subtle variation of shades contained in the original image. In general, the more detail in the 3D target image, the smaller the sampling interval. Therefore, sparse sampling is generally carried out in the areas with less image details, while intensive sampling is carried out in the areas with greater detail transformation. In this way, the useful information of 3D object images obtained will not be affected, but the total data amount is effectively reduced.

After the completion of sampling, the signal amplitude of the 3D target image should be discretized and layered, and the discrete gray value information should be used to replace the continuous analog gray value information, that is, the quantization processing.

After sampling and quantifying the 3D target image, the image data structure that is convenient for computer processing should be generated. Generally, in a computer, a two-dimensional matrix is used to represent and store a three-dimensional object image.

The imaging process of a 3D target image is as follows: firstly, a 3D target image is divided into a series of small squares, which can be called pixels. The grayscale of each square is represented by the integer grayscale.

A pixel is the smallest unit that constitutes the three-dimensional target image. Each pixel in the three-dimensional target image has its own independent attributes, among which the most basic attributes include the pixel position and gray value. The position is determined by the coordinate value of the row and column where the pixel is located, which is usually represented by the position coordinate (x, y, z) of the pixel. The matrix element is the pixel value on the corresponding pixel point.

To sum up, taking the numerical matrix of digital image $f_{M\times N}(x, y)$ as an example, $M \times N$ represents M rows and N columns of the pixel of the 3D target image in the plane space. The numerical matrix of a 3D object image $f_{M\times N}(x, y)$ is then expressed as follows:

$$f_{M\times N}(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0, N-1) \\ f(1,0) & f(1,1) & \cdots & f(1, N-1) \\ \cdots & & & \\ f(M-1, 0) & f(M-1, 1) & \cdots & f(M-1, N-1) \end{bmatrix} \tag{7}$$

As for the digitized 3D target image, it is necessary to evaluate its quality, which can be divided into two types: subjective evaluation and objective evaluation.

Subjective evaluation is the subjective evaluation of the 3D target image by taking a person as the observer of the 3D target image. The quality of the 3D target image is not only related to the characteristics of the image itself, but also related to the characteristics of the observer and the observation conditions.

The objective evaluation method of 3D target image quality is to measure the quality of a 3D target image restoration by the deviation from the original image. The most common methods are mean square error, mean absolute error and normalized mean square error. The objective evaluation method can reflect the gray difference between the original 3D target image and the recovered 3D target image.

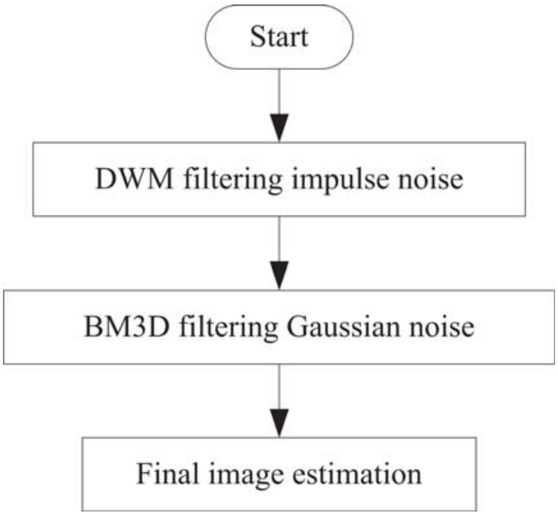

Figure 1 Flow Chart of Traditional 3D Target Noise Filtering Algorithm.

Suppose $f(x, y)$ represents the original 3D target image, $g(x, y)$ represents the 3D target image after degradation, $f'(x, y)$ represents the 3D target image after recovering from a degraded 3D target image, L represents the gray level of the 3D target image, then:

$$MSE = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}[f'(x, y) - f(x, y)]^2}{(L-1)^2} \tag{8}$$

$$MAE = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}|f'(x, y) - f(x, y)|}{(L-1)^2} \tag{9}$$

$$NMSE = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}[f'(x, y) - f(x, y)]^2}{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(x, y)^2} \tag{10}$$

where, *MSE* represents the mean square error, *MAE* represents the mean absolute error, and *NMSE* represents the normalized mean square error.

The composition of each 3D object image system is different, but the common point is to ensure the highest quality of the 3D object image. Therefore, in the process of digital processing of 3D target images, it is necessary to fully consider how to maintain the visual display quality of 3D target images at a high level.

2.2 Design of Noise Filtering Process of a 3D Target Image

The goal of noise filtering is to remove the mixed noise composed of additive white gaussian noise and random value impulse noise.

In order to filter the image noise, the model of mixed noise should be defined. Suppose X represents the noiseless 3D target image, y_G and y_I represent the observed value of gaussian noise and impulse noise to the original image, and $X(i, j)$ represents the pixel value at (i, j) of the noiseless 3D target image. Since the pixel value destroyed by the additive white gaussian noise is the sum of the original pixel value and noise, the damaged pixel $y_G(i, j)$ in y_G can be modeled as follows:

$$y_G(i, j) = X(i, j) + z(i, j) \tag{11}$$

where, $z(i, j)$ represents gaussian white noise and its variance is σ^2.

Unlike additive white gaussian noise, pixels destroyed by impulse noise are replaced by random values. The damaged pixel $y_I(i, j)$ in y_I is modeled as follows:

$$y_I(i, j) = \begin{cases} n(i, j) \times p \\ X(i, j) \times (1-p) \end{cases} \tag{12}$$

where, $n(i, j)$ represents an impulse noise with a random value of 0 to 255, and p is the concentration of impulse noise. Therefore, the damaged pixels in observation value y of mixed noise subject to superposition of gaussian noise and impulse noise are modeled as follows:

$$y(i, j) = \begin{cases} n(i, j) \times p \\ X(i, j) + z(i, j) \times (1-p) \end{cases} \tag{13}$$

Based on the model of mixed noise, the removal of mixed noise is realized. The removal of mixed noise is the main problem in 3D target image processing. As different noises have different characteristics, it is necessary to adopt appropriate removal methods for different noises. In the process of removing mixed noise from 3D target images, different noise removal algorithms need to be combined. The aim of this study is to remove the mixed noise composed of additive white gaussian noise and random value impulse noise [11]. At present, the traditional method does not remove the mixed noise effectively.

The process of traditional image noise filtering algorithm is shown in Figure 1.

Due to the complexity of the traditional image noise filtering algorithm, a new 3D image noise filtering method is designed. The method consists of two parts: a detection process and a denoising process. The step of comparing the minimum

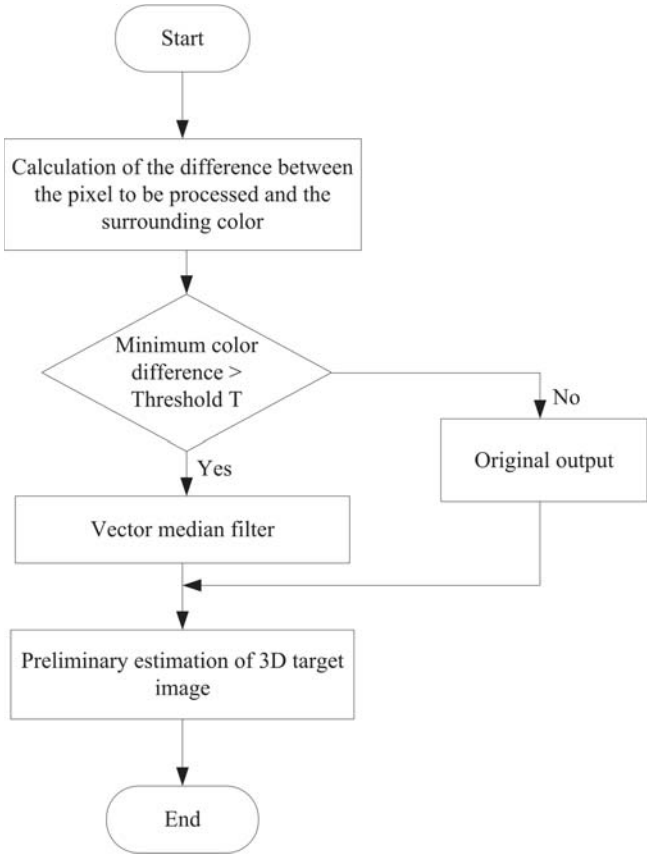

Figure 2 3D Target Image Noise Filtering Process.

chromatic aberration $r_{i,j}$ with the impulse noise threshold T is the boundary line. The process before this step is the impulse noise detection stage, and the process after this step is the impulse noise filtering stage. The algorithm flow is shown in Figure 2.

After the noisy input 3D target image is processed, the image is first processed in the order of pixel blocks to ensure that each pixel is identified in the detection window.

The specific detection process is as follows: according to the way of direction weighting, the pixel to be processed in the detection window is the center, and the chromatism judgment is made on its top, bottom, left and right directions, and the chromatism value pairs of the four directions are obtained. The chromatic aberration values in the four directions are sorted, and the smallest chromatic aberration value $r_{i,j}$ is taken as the measurement of impulse noise detection. The minimum chromatic aberration value is compared with the threshold value of impulse noise. If the minimum chromatic aberration value is greater than the threshold value of impulse noise detection, the pixel point is determined to contain impulse noise components. If the minimum chromatic aberration value is less than or equal to the threshold of impulse noise detection, the pixel point is determined to contain no impulse noise components [12]. The calculation method of the minimum chromatic aberration is as follows: judge the chromatic aberration $d_{i,j}^{(k)} \quad 1 \leq k \leq 4$ in each direction, and the calculation process is as follows:

$$d_{i,j}^{(k)} = \sum_{(s,t)\in S_k} \sqrt{|y_I(i+s, j+t) - y_I(i, j)|^2} \tag{14}$$

In particular, when the three-dimensional target image is a color image, the evaluation chromatic aberration is obtained after the above calculation in the primary color layer of the three-medium image, and then the four-direction minimum chromatic aberration value $r_{i,j} = \min\{d_{i,j}^{(k)} \quad 1 \leq k \leq 4\}$ is used as the measurement index of impulse noise detection. When $r_{i,j} > T$, $y(i, j)$ is the pixel polluted by impulse noise; otherwise, $y(i, j)$ is the pixel not polluted by impulse noise.

The pixels containing impulse noise are marked and denoised by the vector median filtering algorithm in the filtering window. The pixel without an impulse noise component is not processed and is output directly. Through the above processing on all pixel points in turn, the image after the impulse noise component is filtered out in the first stage can be obtained [13].

2.3 The Realization of Noise Filtering of Three-Dimensional Target Image

In this study, 3D target image noise filtering is realized based on the machine learning method, and this realization process is shown in Figure 3.

The realization of 3D target image noise filtering firstly needs to obtain the pre-estimated image through hard threshold calculation, then obtain the final output 3D target image through Wiener filtering, and achieves the image filtering and output through the machine learning process [14]. The 3D target image fragments used in this method are all square blocks with fixed size. The specific realization process is

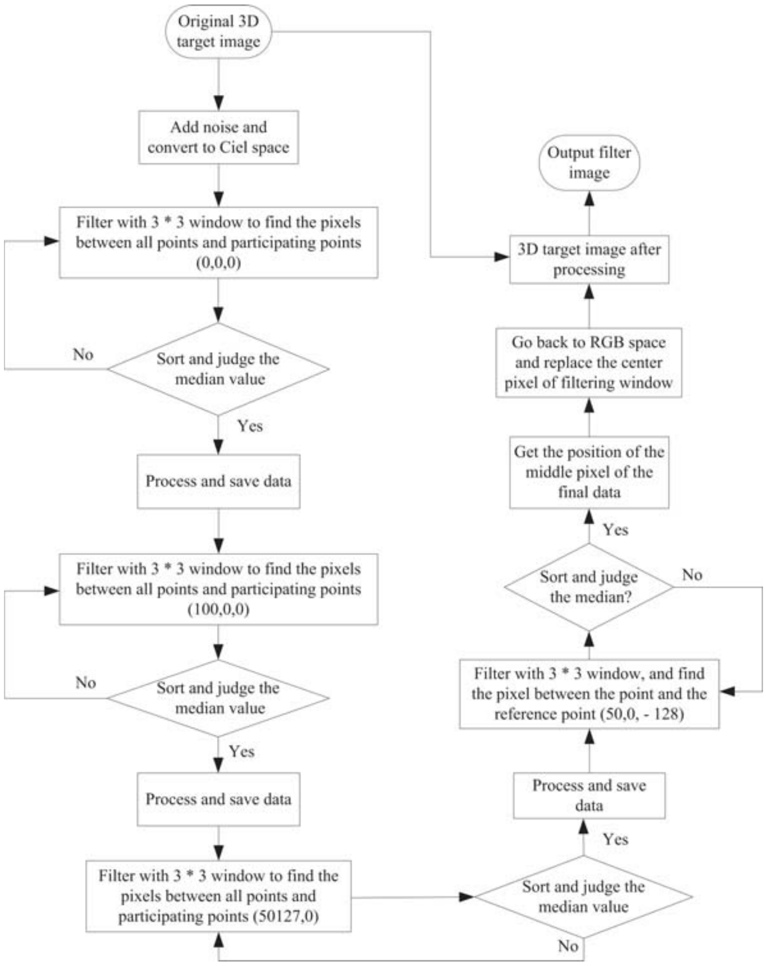

Figure 3 Implementation Process of 3D Target Image Noise Filtering Based on Machine Learning.

as follows:

Step 1: The algorithm is initialized and the original 3D target image is read.

Step 2: Add salt and pepper noise to the 3D target image, and convert the original 3D target image to CIEL*a*b* uniform space according to the conversion formula.

Step 3: 3*3 window is used for initial filtering until the maximum filtering window is confirmed to be 7 or 9 (i.e. 7*7 or 9*9) and the distance l between each pixel and the reference point is calculated.

Step 4: Based on the distance obtained in Step 3, find and calculate the weights of the filter window S_{xy} and each pixel point, calculate the mean values of the pixel point and center and store gray values $A1$ and $A2$ respectively.

Step 5: Confirm the size of $A1$ and $A2$ and judge. When $A1 > 1$ and $A2 < 1$, go to the second step. Otherwise, increase the size of the filter window. If the filter window is $S_{xy} \leq l$, repeat Step 3, otherwise output the result as the final value.

Step 6: Finish the adaptive window algorithm and confirm the position of the noise point in the CIEL*a*b* space, as well as the position of the replacement point.

Step 7: Trace back to the 3D target image space, get the pixel points in the same location as the result in Step 6, and replace them with this pixel.

Step 8: The whole 3D target image is traversed until the noise filtering is completed.

According to the realization flow of 3D target image noise filtering, the specific steps of 3D target image noise filtering are obtained, and the 3D target image noise filtering is realized.

3. EXPERIMENTAL COMPARATIVE ANALYSIS

In order to verify the practical performance of the machine learning-based noise filtering method for 3D target images designed in this study, the following comparative detection experiment is designed.

3.1 Simulation Experiment Environment Design

The experiment was carried out on the MATLAB simulation platform. MATLAB is usually designed in two ways: either by compiling M files, or by utilizing the MATLAB GUI. Most simulation experiments will choose the latter, mainly because the MATLAB GUI has more simple and intuitive interface, it is easy to adjust the layout, easy to modify the program code and has other advantages.

The main steps when designing the interface of MATLAB GUI are as follows:

1. First, define the design tasks, analyze the functions to be realized in the interface design and draw simple sketches.

2. Reasonably arrange the layout of the controls, including the relevant labels of graphic controls such as images and co-ordinates that need to be displayed, and the positions of button controls.

3. Set the properties of each control according to the needs of the function realization. The Tag attribute is one of the important attributes; its value is often a string, and according to the property value of each control, the control can be modified accordingly.

4. Write functional code. The code design includes several main functions: OpenningFcn (initial function), Callback (Callback function), and OutputFcn (output function). OpenningFcn is used to set the initial value of each parameter, which can be set according to the design requirements. Callback is the core function of the design, which mainly refers to the corresponding function of the time when the interface space is triggered. It is certain that without this function, no operation can be carried out in the interface design. OutputFcn (output function) is the information returned to the command line after the function runs.

5. Run and perform the corresponding functional code debugging. If after the completion of the interface design, it is found that the function of the control is not realized or is different from the expected effect, the previous part needs to be repeated to check and correct errors.

In the experiment, the 3D target image (24 bit RGB, 480*480) was selected as the experimental object, and the probability of each pixel being polluted was considered to be equal.

In order to highlight the validity of the experimental results, the method presented in this paper, the image noise filtering method based on edge retention, and the image noise filtering method based on adaptive mean are used respectively for experimental comparison. The experimental platform and environmental equipment are always consistent.

In order to verify the feasibility of different methods, Peak signal-to-noise ratio (PSNR), Structural similarity index (SSIM) and image texture complexity were selected as the evaluation criteria for image noise filtering quality.

3.2 Experiment Process and Result Analysis

(1) Preliminary validation

In the initial stage, the filtering method based on edge retention, the filtering method based on adaptive mean and the filtering method based on machine learning were used respectively to perform noise filtering processing on one image, and the image processing effect was compared, as shown in Figure 4.

It can be seen from the analysis of Figure 4 that after image processing using the filtering method based on edge retention and the filtering method based on adaptive mean, the clarity and brightness of the results are inferior to the images processed by filtering method based on machine learning, this preliminarily proves the effectiveness of the filtering method based on machine learning.

(2) PSNR and SSIM were compared

Since the filtering method based on edge retention and the filtering method based on adaptive mean can both achieve a good filtering effect under low noise density, the image center point with relatively high noise density is selected as the actual pixel point to make the experiment more meaningful. The PSNR and SSIM of output images under different filtering methods are compared, and the results are shown in Table 1 and Table 2 respectively.

According to Table 1 and Table 2, as noise density increases, the PSNR and SSIM values of output images under different filtering methods also change constantly. According to the data analysis, the advantages of the filtering method based on machine learning are obvious, especially when the noise density is high, the PSNR of the output image is high, and the SSIM is large. With the increase of noise density, although the PSNR and SSIM of the image based on machine learning filter decreased, it still had obvious advantages over the other two methods.

In order to more intuitively compare the changes of image PSNR and SSIM values under different filtering methods, the above data is presented in the form of a graph, as shown in Figure 5 and Figure 6.

The analysis of Diagram 5 and Figure 6 shows the PSNR under different filtering methods and how it reduces with the increase of noise density, among them, the image noise filtering method based on edge retention and the image noise filtering method based on adaptive average PSNR value reduction is greater than the method in this paper, and the method of image PSNR is always higher than the other two methods. Meanwhile, the image SSIM value in this method is always higher than that in the other two methods.

The analysis also shows that under high noise density, compared with image noise filtering method based on edge retention and image noise filtering method based on adaptive mean, the filtering method based on machine learning research design showed a good experimental effect, effectively removing the noise, while sharpening the image edge and enhancing the texture detail information as well. (3) Image texture complexity comparison.

In order to further verify the application performance of the machine learn-based noise filtering method for 3D target images, the image texture complexity of output images under different methods is compared. The complexity of image texture can be reflected by the second moment (variance) of the gray level histogram of the pixels in the image, which can reflect the characteristics of image content attributes. The

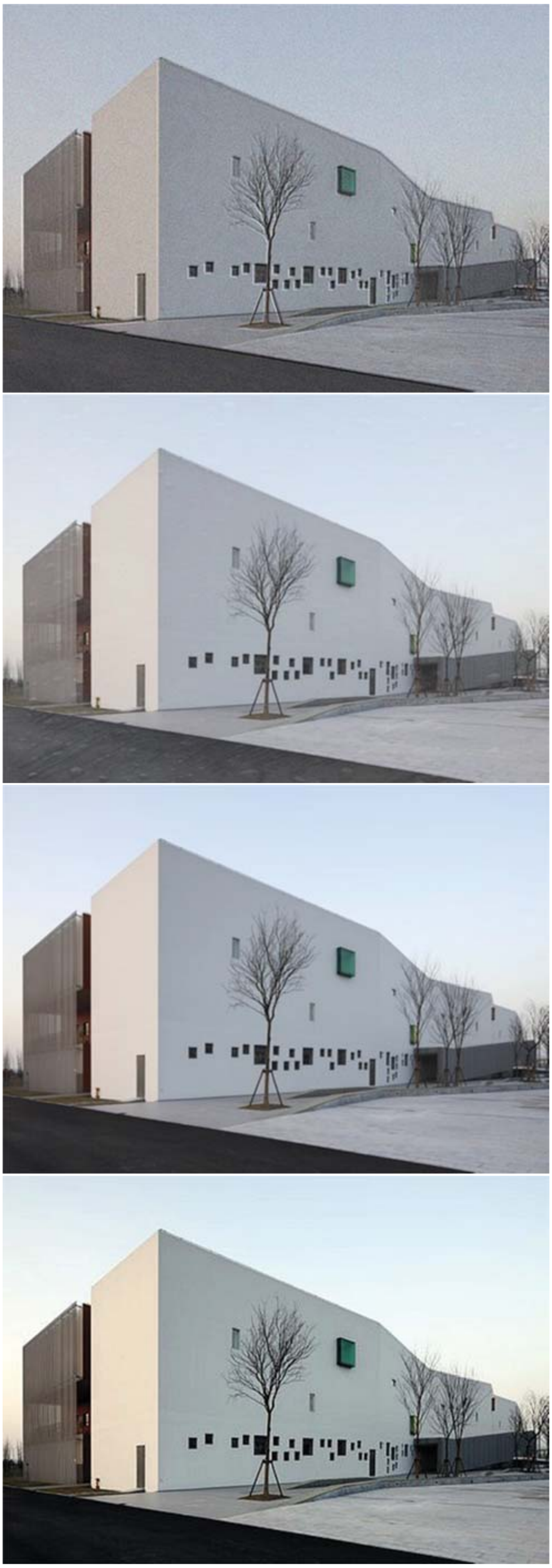

Figure 4 The Results of Image Processing by Different Methods.

Table 1 PSNR Data Compared Under Different Filtering Methods.

	PSNR/dB		
Noise density	Filtering method based on edge retention	Filtering method based on adaptive mean value	Filtering method based on machine learning
20%	43.3773	44.2028	54.2198
25%	41.2300	42.3324	53.5222
30%	39.4704	40.5494	53.3146
35%	38.0131	39.0903	53.6877
40%	36.7092	37.8353	52.0546
50%	34.8512	35.9265	52.2544

Table 2 SSIM Data Comparison Under Different Filtering Methods.

	SSIM		
Noise density	Filtering method based on edge retention	Filtering method based on adaptive mean value	Filtering method based on machine learning
20%	0.4151	0.5559	0.9001
25%	0.2127	0.4189	0.7964
30%	0.1188	0.3181	0.6230
35%	0.0830	0.2471	0.4697
40%	0.0569	0.1881	0.3742
50%	0.0355	0.0933	0.2793

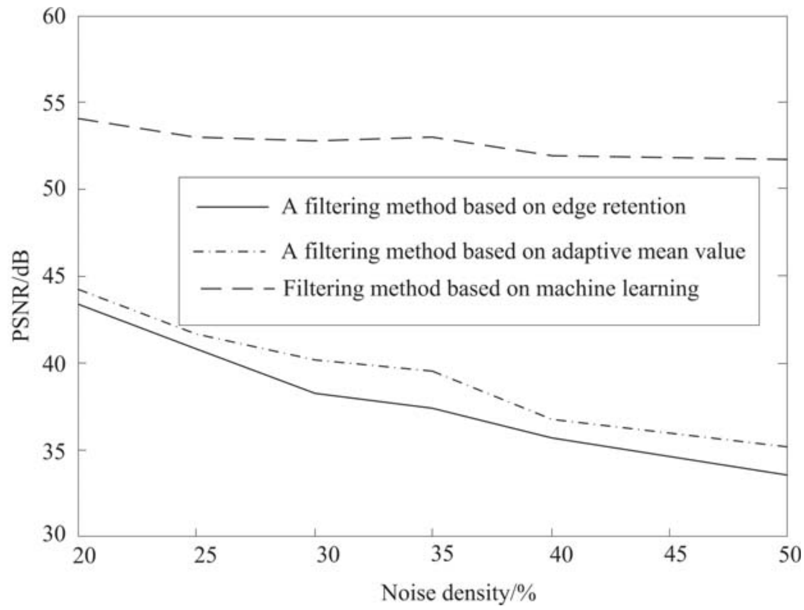

Figure 5 PSNR Curves Under Different Filtering Methods.

higher the texture complexity, the higher the image quality will be after processing.

The texture complexity comparison of 3D target images after filtering by different methods is shown in Figure 7.

The analysis of Diagram 7 shows that the method of image texture complexity based on machine learning fluctuates around 0.8, by contrast, the image noise filtering method based on edge retention and the image noise filtering method based on adaptive mean image texture complexity are significantly lower, reaching a minimum of only 0.61. Therefore, after the noise filtering method of 3D target images based on machine learning designed in this study, the texture complexity of images is high, and the image quality after processing is also improved.

4. CONCLUSION

In this paper, the noise filtering process of 3D target images was designed. Additive white gaussian noise and random value impulse noise were composed into mixed noise. Through the detection process and denoising process, combined with machine learning method, the image noise was filtered out. The experimental results show that the PSNR

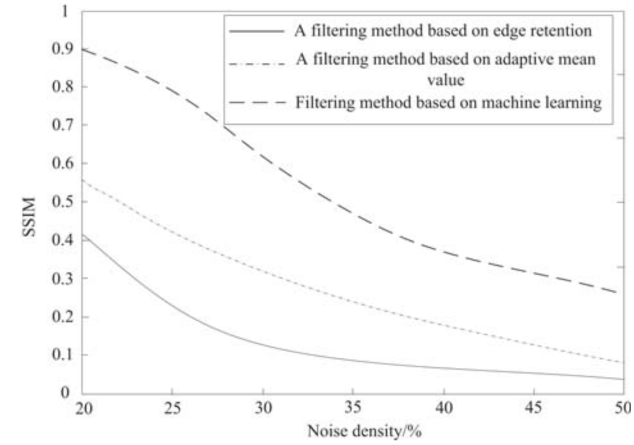

Figure 6 SSIM Comparison Curve Under Different Filtering Methods.

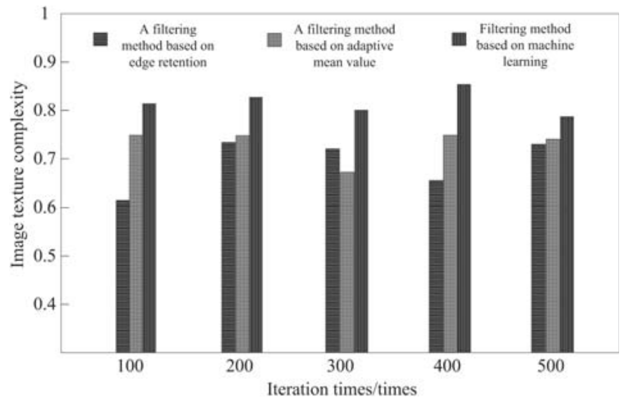

Figure 7 Texture Complexity Comparison in Different Methods.

and SSIM values and texture complexity values of 3D target images are higher after processing by this method, which not only achieves effective image filtering, but also maximizes the protection of image details.

REFERENCES

1. Angeli, Alessia, Ferri, Massimo, Tomba, & Ivan (2018). Symmetric Functions for Fast Image Retrieval with Persistent Homology. *Mathematical Methods in the Applied Sciences, 41*(18), 9567–9577.
2. Nakhoon Baek, & Kuinam J. Kim. (2017). An Artifact Detection Scheme with CUDA-based Image Operations. *Cluster Computing, 20*(1), 1–7.
3. Zihong Chen, Dexiang Zhang, Yao Xu, et al. (2018). Research of Polarized Image Defogging Technique based on Dark Channel Priori and Guided Filtering. *Procedia Computer Science, 131*, 289–294.
4. Gangyan Zeng, & Xiaoyan Zeng. (2018). A Color Image Filtering Approach based on Octant Division of RGB Coordinate System. *Journal of Hunan Institute of Engineering (Natural Science Edition), 28*(04), 28–31.
5. Jianhui Song, Simeng Fan, Yang Yu, et al. (2019). Image Denoising and Enhancement for Vehicle Target Identification. *Transactions of Shenyang Ligong University, 38*(04), 25–29.
6. Wencheng Wangm, Jian Li, Ruilan Wang, et al. (2019). Design and Development of Simulation Platform for Digital Image Processing based on MATLAB GUI. *Experimental Technology and Management, 36*(02), 141–144.
7. Binghong Chen (2019). Noise Reduction of Similarity Multi-chromatographic Image based on Gray Edge Fusion. *Electronic Measurement Technology, 42*(11), 92–96.
8. Jin Zhu, Weiqi Jin, Li Li, et al. (2017). Multiscale Infrared and Visible Image Fusion using Gradient Domain Guided Image Filtering. *Infrared Physics & Technology, 89*, 8–19.
9. David Concha, Raúl Cabido, Juan José, Pantrigo, et al. (2018). Performance Evaluation of a 3D Multi-view-based Particle Filter for Visual Object Tracking Using GPUs and Multi-core CPUs. *Journal of Real-Time Image Processing, 15*(2), 309–327.
10. Zhongyun Hua, & Yicong Zhou (2017). Design of Image Cipher using Block-based Scrambling and Image Filtering. *Information Sciences, 396*, 97–113.
11. Rubel Oleksii, Lukin Vladimir, & Egiazarian Karen (2018). Additive Spatially Correlated Noise Suppression by Robust Block Matching and Adaptive 3D Filtering. *Journal of Imaging Science and Technology, 62*(6), 1–11.
12. Francesc Pons Llopis, Nikolas Kantas, & Alexandros, Beskos (2017). Particle Filtering for Stochastic Navier-stokes Signal Observed with Linear Additive Noise. *SIAM Journal on Scientific Computing, 40*(3), 1544–1565.
13. Ziyun Wang, Ze Tang, & Ju H. Park (2019). A Novel Two-stage Ellipsoid Filtering-based System Modeling Algorithm for a Hammerstein Non-linear Model with an Unknown Noise Term. *Nonlinear Dynamics*, (4), 2919–29915.
14. Sy Dzung Nguyen, Seung-Bok Choi, & Tae-Il Seo (2018). Recurrent Mechanism and Impulse Noise Filter for establishing ANFIS. *IEEE Transactions on Fuzzy Systems, 26*(2):985–997.