ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

A Filter Image Edge Detection Algorithm Based on PCNN

Xiangyu Deng[1, a], Haiyue Yu[1, b, *] and Xikai Huang[1, c]

[1]School of Physics and Electronic Engineering Northwest Normal University, Gansu Lanzhou, China

[a]dengxy000@126.com, [b]yuhaiyue2021@163.com, [c]15266372275@163.com

Abstract

As a basic task in image processing, edge detection is widely used in advanced vision tasks. Traditional edge detection algorithms, such as Canny and Sobel, start with the gradient of the image, select the place where the pixel gradient changes obviously, and complete the edge and contour extraction. It has the characteristics of fast speed and immediacies, but the extracted edge is easily affected by noise. To solve this problem, this paper proposes a filter image edge detection algorithm based on PCNN. Starting from the spectrum component information of the image, noise interference is filtered first, and then the noise is further filtered through PCNN model iteration to obtain the optimal binary image of the image, and finally the edge information is extracted through Gauss high-pass filter. Experimental results show that the proposed algorithm is superior to the traditional edge detection operators under the evaluation of PSNR, MSE, FOM and ELE, and can meet the requirements of practical tasks while maintaining the edge of image details.

Keywords

PCNN; Filter; Traditional edge detection operator; Image edge detection.

1. INTRODUCTION

Edge detection [1]is mainly used to determine the edge of an image by extracting the regions in the image where the pixel gradient changes significantly, and identifies the clusters where the luminance changes from bright to dark and from dark to bright to form the final contour, which contains important information in an image. The reason why people can distinguish objects in pictures is mainly based on their edge and contour features. Edge detection has become one of the basic hot research issues in computer vision, digital image processing and other fields [2-4]. As the most basic hot content in digital image processing technology, image edge detection technology can be more intuitive to show the edge or structure of the object, and plays an irreplaceable role in intelligent transportation [5], medicine [6] [7], agriculture and forestry [8]. With the continuous update and development of digital image technology, edge detection algorithms are mainly divided into two categories: traditional edge detection algorithm [9] and neural network-based edge detection algorithm [10]. Traditional edge detection algorithms based on the gradient characteristics of the image, the calculation of the eigenvalues of the local window to achieve edge detection. The first-order differential algorithm calculates the directional gradient value of the image, and the common first-order differential algorithms include Sobel algorithm [11], Roberts algorithm [12] and Prewitt algorithm [13]. The method of second-order differential operator calculates the second-order differential of the gray value as the reference of the edge, the common second-order differential algorithms include: Laplacian algorithm [14], LOG algorithm [15] and so on. For the edge detection algorithm based on neural network, CNN network is generally used to sense the edge from

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

multiple scales, independently supervise the output of different scales, and finally integrate to get the final edge result. The operator of traditional edge detection algorithm has the advantages of fast operation speed and easy to implement. But the image has poor anti-noise performance and poor connection of edge effect. In order to achieve the best detection effect, the neural network edge detection algorithm based on neural network has more and more complex structure design, higher layer number of network stack and more parameter number of calculation. Therefore, the convergence of network training is very slow, and the cost of computer resource consumption increases. In view of the above problems, as a basic task in image processing, it is not only necessary to avoid the loss of details, retain important edges, accurately identify target areas, but also to quickly and efficiently identify image edges.

The contribution of this paper can be summarized in the following two points: (1) A filter image edge detection algorithm based on PCNN is proposed, which combines PCNN model and filter for the first time. First, the median filter is used to effectively reduce the noise of the target image. The median filter can reduce the influence of Gaussian noise on the guaranteed image to a large extent, and effectively keep the image edge details from being blurred; Secondly, the image after noise reduction is segmented iteratively by PCNN model, and the binary image is obtained. Finally, Gaussian high-pass filter is used to filter the binary image, extract the high frequency information in the image, and obtain the edge contour of the image. (2) Compared with the traditional edge detection operator, the anti-noise ability of the image is improved, the recognition accuracy of the image is guaranteed, and the training is completed in a short time with a small amount of resources.

2. EDGE DETECTION PRINCIPLE OF DIFFERENT OPERATORS

2.1. Edge detection principle

Image edge detection refers to the process of detecting the edge points and edge line segments from an image, and then describing the direction of edge [16] [17]. Generally speaking, the edge is the place where the signal *changes dramatically*, and in advanced mathematics, the point where the derivative value of a function is at its maximum is the point where the function *changes most dramatically*. (x, y) is represented by the current pixel position, $f(x, y)$ is represented by the gray value of the current pixel position, and the gradient change of the gray image can be obtained through differential calculation. Since the gray image is a discrete two-dimensional function, the gradient change can be obtained by obtaining the partial derivative of the horizontal direction z and vertical direction y respectively, and the calculation formula is as follows:

$$\frac{\partial f}{\partial x} = \frac{f(x+1, y) - f(x, y)}{\Delta x}$$
$$\frac{\partial f}{\partial y} = \frac{f(x, y+1) - f(x, y)}{\Delta y} \tag{1}$$

The gradient of a two-dimensional function $f(x, y)$ is defined as a vector:

$$\nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \tag{2}$$

The magnitude of the gradient is:

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

$$\nabla f = \left|\nabla f(x,y)\right| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \tag{3}$$

Therefore, for a pixel (x,y) in the image, whether the pixel is an edge point can be judged by setting a threshold value. If the gradient amplitude of the pixel calculated by the differential operator is greater than the threshold value, the pixel is the edge point of the image, and the image containing edge information is obtained, while the most important information in the image is retained.

2.2. Canny operator

Canny edge detection algorithm, proposed by John F. Canny [18], is the first derivative of Gaussian function and the optimal approximation operator of the product of signal-to-noise ratio and positioning accuracy. It is a traditional multi-step edge detection method and improves the evaluation standard of edge detection theoretically. It consists of the following four steps, Figure 1 is the flow diagram of the Canny operator.

Step 1: Gaussian filtering of the image

A Gaussian function is a function analogous to a normal distribution in that the image is smoothed using a Gaussian filter with a specified standard deviation σ to minimize noise interference with the image. The expression of the Gaussian filter function is:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{\frac{x^2+y^2}{-2\sigma^2}} \tag{4}$$

The smoothed image expression $M(x,y)$ is:

$$M(x,y) = G(x,y) \times f(x,y) \tag{5}$$

Step 2: Calculate the gradient and amplitude

In images, gradients are used to represent the degree and direction of change in gray values. When calculating the gradient amplitude and gradient direction of the smoothed data array, the Canny algorithm usually adopts 2×2 neighborhood first-order partial derivative difference, and the two arrays E_x and E_y of the x direction and y direction partial derivatives are respectively:

$$E_x = \frac{\partial M(x,y,\sigma)}{\partial x} \times f(x,y)$$
$$E_y = \frac{\partial M(x,y,\sigma)}{\partial y} \times f(x,y) \tag{6}$$

The expression for the gradient amplitude $\left|\nabla M(x,y)\right|$ and gradient direction $\theta(x,y)$ of each pixel point (x,y) in $M(x,y)$ is:

$$\left|\nabla M(x,y)\right| = \sqrt{E_x^2 + E_y^2}$$
$$\theta(x,y) = \arctan\left(\frac{E_x}{E_y}\right) \tag{7}$$

Where $\left|\nabla M(x,y)\right|$ reflects the edge strength and $\theta(x,y)$ represents the direction perpendicular to the edge.

Step 3: Non-maximum suppression

Non-maximum suppression is used to preserve local maxima in the direction of the gradient, filtering out points that are not edges so that the width of the edge is represented with as few pixels as possible.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

Step 4: Obtain the high and low thresholds and determine the edges

Based on the two thresholds, the pixels are divided into three parts: strong edge, weak edge and non-edge according to the gradient amplitude. Then the edge is processed according to the high and low thresholds. It should be noted that the selection of the high and low thresholds needs to be analyzed according to the specific image. If a weak edge is connected to a strong edge, it is classified as a strong edge, otherwise it is suppressed.

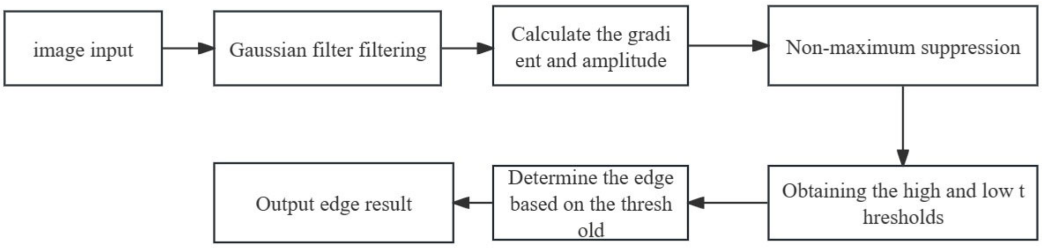


Figure 1. Canny operator detection flow chart

2.3. Roberts operator

Roberts operator [19] is a simple first-order linear differential algebraic operator, which uses the difference between two adjacent pixels in the diagonal direction as the approximate value of the gradient to process the image. The schematic diagram of its algorithm is shown in Figure 2.

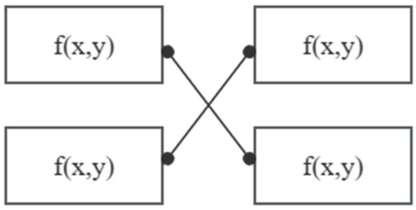


Figure 2. Roberts operator diagram

The gradient size represents the intensity of the edge, and the direction of the gradient is orthogonal to the intensity direction of the edge. The expression of the gradient formula is as follows:

$$\left| grad\, f(x,y) \right| = \sqrt{(f(x+1,y+1)-f(x,y))^2 + (f(x+1,y)-f(x,y+1))^2} \tag{8}$$

Where $f(x,y)$ is the gray value of the changed image.

2.4. Sobel operator

Sobel operator [20] is a first-order differential operator used to detect the edge brightness of pixel gradient images. Sobel operator emphasizes the effect of pixel position. For the first time, two 2×2 square operators are used to calculate the gradients in the x and y directions of images. The expression can be expressed as:

$$\begin{aligned} S_x &= \left[f(x+1,y-1)+2f(x+1,y)+f(x+1,y+1) \right] - \left[f(x-1,y-1)+2f(x-1,y)+f(x-1,y+1) \right] \\ S_y &= \left[f(x-1,y+1)+2f(x,y+1)+f(x+1,y+1) \right] - \left[f(x-1,y-1)+2f(x,y-1)+f(x+1,y-1) \right] \end{aligned} \tag{9}$$

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

The Sobel operator is the average summation difference after weighting the values corresponding to the forward or column, which reduces the ambiguity of the edge. There are many definitions of it, respectively:

1) The Sobel operator that distinguishes horizontal and vertical directions is defined as:

$$S_x = |s_x|,\ S_y = |s_y| \tag{10}$$

2) The Sobel operator considering both horizontal and vertical directions is defined as:

$$S = \sqrt{s_x^2 + s_y^2} \tag{11}$$

3) The Sobel operator of the larger horizontal and vertical directions is defined as:

$$S = \max\{|s_x|,\ |s_y|\} \tag{12}$$

2.5. Prewitt operator

The Prewitt operator [21] is a linear filter, essentially a set of discrete convolution kernels, which can detect the edges in the image by convolution operation with the image, and its calculation expression is as follows:

$$\begin{aligned} P_x &= \left[f(x-1,y-1)+f(x-1,y)+f(x-1,y+1)\right]-\left[f(x+1,y-1)+f(x+1,y)+f(x+1,y+1)\right] \\ P_y &= \left[f(x-1,y+1)+f(x,y+1)+f(x+1,y+1)\right]-\left[f(x+1,y-1)+f(x,y-1)+f(x+1,y-1)\right] \end{aligned} \tag{13}$$

P_x and P_y represent the gray values of pixels calculated by the Prewitt operator in the horizontal and vertical directions respectively, and the conversion template is as follows:

$$P_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad P_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \tag{14}$$

When these two convolution kernels are applied to each pixel of the image separately, they calculate how the intensity values of the pixels around that pixel change, thereby detecting edges in the image. Horizontal convolution kernels can detect horizontal edges in an image (such as the left and right edges), while vertical convolution kernels can detect vertical edges (such as the top and bottom edges). By combining the edges in these two directions, a more comprehensive edge detection result can be obtained.

3. FILTER EDGE DETECTION ALGORITHM BASED ON PCNN

The traditional edge detection algorithm needs to design the gradient operator manually, which is greatly limited by complex mathematical derivation and other factors. In addition, the traditional operator has poor anti-noise performance, easy to be attacked and interfered with by noise, and poor ability to locate image edges. In order to reduce the influence of noise and obtain more accurate image edges, this paper adopts median filter to filter the Gaussian noise interference in the image, and sends the smoothed image as the input image to PCNN network for iteration. According to iterative segmentation, the optimal binary segmentation graph is selected. The PCNN network will further filter out the noise interference of the image. Finally, the optimal binary segmentation is fed into the Gaussian high-pass filter to obtain the image edge. The algorithm flow chart of this paper is shown in Figure 3.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

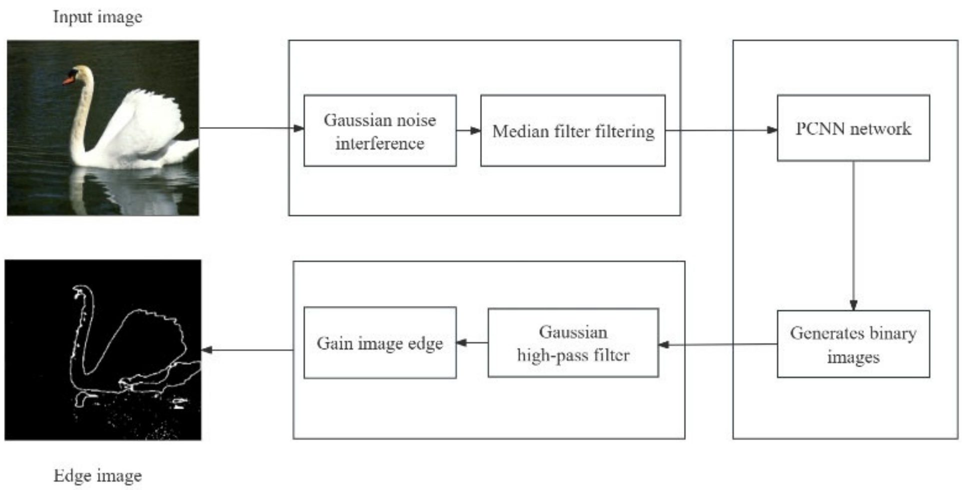


Figure 3. Algorithm flow chart of this paper

3.1. Median filter

Image filtering [22] is to suppress the noise in the target region of the image while preserving the details of the image as much as possible. For an image, the energy of the image is mostly concentrated in the low and middle frequency bands of the amplitude spectrum, while in the higher frequency band, the information of interest is often disturbed by noise, which makes it difficult to obtain the edge information of the image. There are two requirements for the filtering process: first, the important information such as the outline and edge of the image can not be damaged, and the second is to make the image clear and good visual effect.

Median filter [23] is a nonlinear signal processing filter, also known as a statistical sorting filter. The principle is to replace each pixel with the median value in the rectangular field around the center pixel, so that the surrounding pixel value is close to the true value, and has a good filtering effect on pepper and salt noise, pulse noise, etc., can maintain the edge characteristics of the image, and will not make the image produce significant blur. The basic idea diagram is shown in Figure 4.

10	9	4	7	2
4	31	12	23	30
5	43	1	32	8
54	7	3	43	4
23	76	56	45	9

10	9	4	7	2
4	31	12	23	30
5	43	23	32	8
54	7	3	43	4
23	76	56	45	9

Figure 4. Basic idea diagram of median filter

The specific operation of the median filter can be described as: using a moving window with an odd number of points, and the template center coincides with a pixel position in the figure, reading the corresponding pixel gray value under the template and sorting, selecting the gray value of the middle value of the pixel in the middle of the gray scale sequence, and allocating the middle value to the pixel in the center position of the template. The images after the original, noise and median filtering are shown in Figure 5.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

Figure 5. Median filtered image

3.2. Principle and application of PCNN

Pulse-coupled neural network (PCNN) [24] has a strong biological background. It is proposed based on the phenomenon of synchronous pulse bursts in the cerebral cortex of mammals such as cats. PCNN does not need to learn or train, and can extract effective information from complex backgrounds, which is called a new model of the third-generation artificial neural network. PCNN has a wide range of applications in the field of image processing. In image processing, many neurons complete the work synchronously. The number of neurons is equal to the number of pixels in the input image, and the model diagram of each neuron is shown in Figure 6.

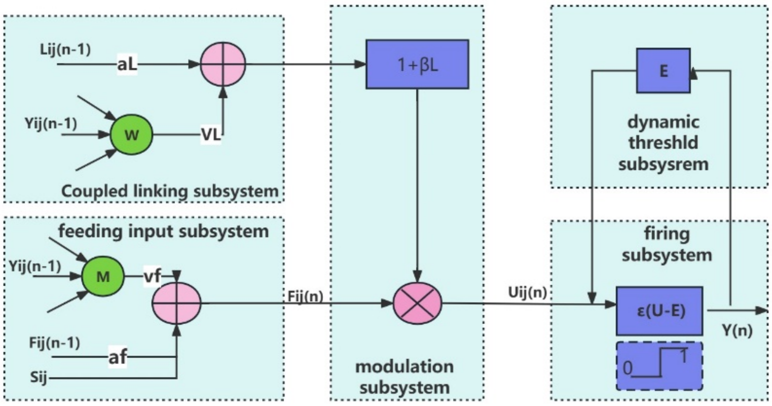


Figure 6. PCNN model diagram

The above subsystems can be described by the discrete equation expression as follows:

$$
\begin{gathered}
F_{ij}(n) = e^{-a_F} F_{ij}(n-1) + V_F \sum M_{ijkl} Y_{kl}(n-1) + S_{ij} \\
L_{ij}(n) = e^{-a_L} L_{ij}(n-1) + V_L \sum W_{ijkl} Y_{kl}(n-1) \\
U_{ij}(n) = F_{ij}(n)(1 + \beta L_{ij}(n)) \\
E_{ij}(n) = e^{-a_E} E_{ij}(n-1) + V_E Y_{ij}(n-1) \\
Y_{ij}(n) = \varepsilon[U_{ij}(n) - E_{ij}(n)]
\end{gathered}
\tag{15}
$$

Among them, each formula represents: feeding input subsystem, coupled linking subsystem, modulation subsystem, dynamic threshold subsystem, firing subsystem. The subscript i,j represents the position of a pixel in the digital image; subscript k,l represents the position of the neighboring pixels corresponding to the center pixel; M,W represents the linking weight matrix between the center pixel and the neighboring pixels; a_F and V_F, a_L and V_L, a_E and V_E, respectively represent the time constant of the iteration decay and linking weight

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

amplification coefficient of the corresponding subsystems. β is the internal activity term linking coefficient of the modulation subsystem, and S_{ij} represents the input of the PCNN and is the normalized gray value of a pixel corresponding to a neuron.

In order to facilitate parameter setting and model construction, the PCNN model adopted in this paper is as follows:

$$
\begin{aligned}
F_{ij}(n) &= S_{ij} \\
L_{ij}(n) &= V_L \sum W_{ijkl} Y_{kl}(n-1) \\
U_{ij}(n) &= F_{ij}(n)(1+\beta L_{ij}(n)) \\
E_{ij}(n) &= e^{-a_E} E_{ij}(n-1) + V_E Y_{ij}(n-1) \\
Y_{ij}(n) &= \varepsilon[U_{ij}(n) - E_{ij}(n)]
\end{aligned}
\tag{16}
$$

After noise reduction, images interfered with noise are sent to PCNN model as input images for iterative segmentation. According to previous research conducted by the team [25], when parameter is set to $V_L = 1$, $a_E = 0.0771$, $V_E = 0.9740$, β=1 and the neighborhood coupled linking is weakly coupled, the binarization image of PCNN iterative output will show a phenomenon of gradual fire suppression. The iterative output of PCNN is shown in Figure 7.

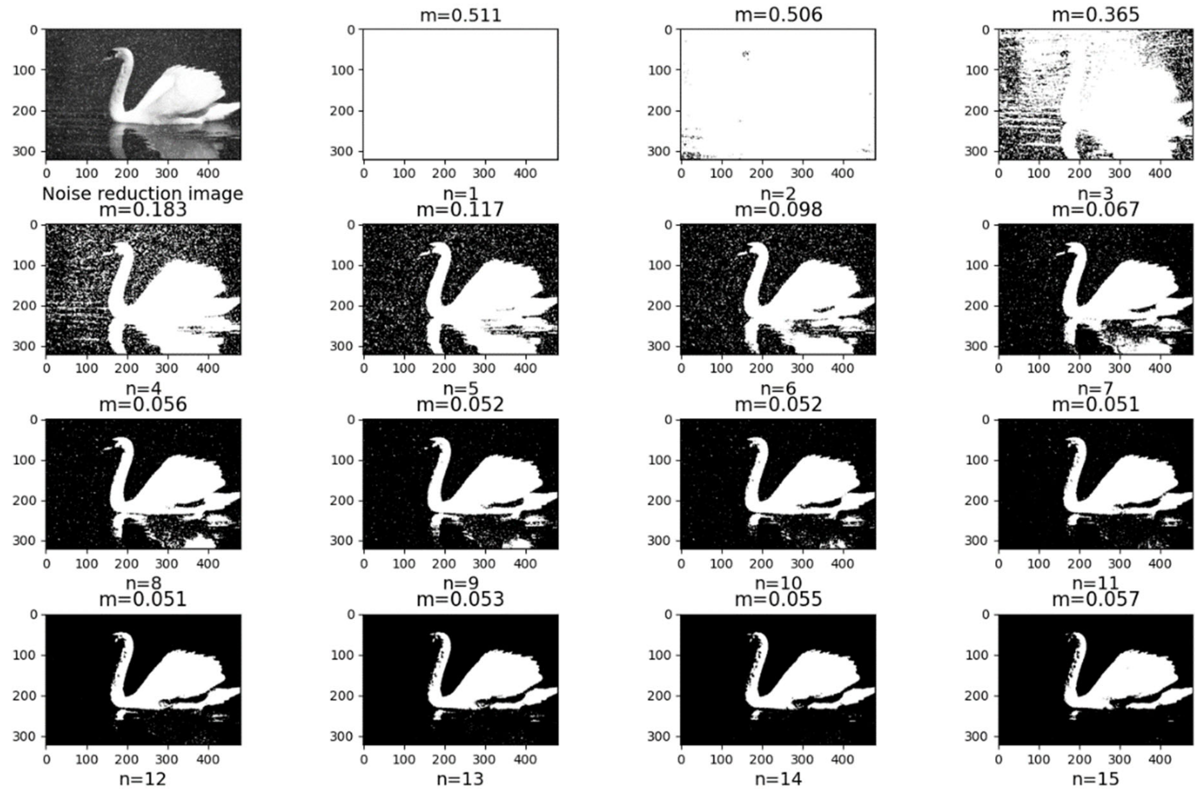


Figure 7. PCNN iteration diagram

In order to minimize the interference of noise on the image and obtain the optimal binarized image in the iterative process, the paper selected the iterative output graph when $n = 12$ on the basis of the combination of mean-square error evaluation index and subjectivity, as shown in Figure 8 for subsequent experiments. In addition, from the PCNN iterative output renderings, we can also see that in the process of PCNN iterative segmentation, the noise reduction image can be further reduced to a certain extent.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

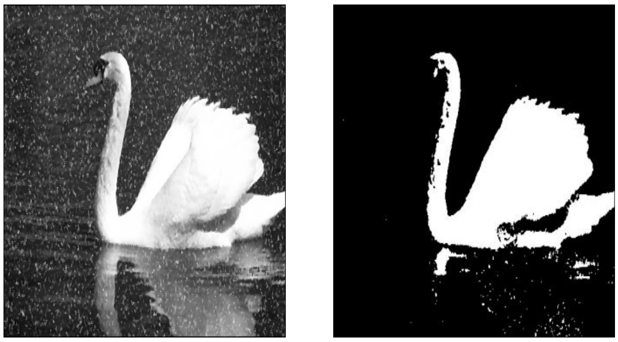


Figure 8. Optimal binary iteration graph

3.3. Gauss high-pass filter

The Gaussian high-pass Filter [26] is a commonly used image processing filter, which is used to enhance the High frequency detail information in the image and suppress the low frequency part. Gauss high-pass filters in different dimensional coordinates are shown in Figure 9, and their generation formula is as follows:

$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2} \tag{17}$$

Where u and v represent the frequency variables respectively, they are usually used to represent the frequency components of the image in the horizontal and vertical directions. $D(u,v)$ is the distance from the frequency variable (u,v) to the center frequency, D_0 represents the cut-off frequency, the cut-off frequency determines the boundary of the filter, and the part beyond the cut-off frequency will be suppressed. In practical applications, the choice of cut-off frequency has a great influence on the filtering effect. A smaller cutoff frequency will enhance high-frequency detail in the image, but may result in edge sharpening and noise enhancement. A larger cut-off frequency will reduce high-frequency detail in the image, but may retain more low-frequency information.

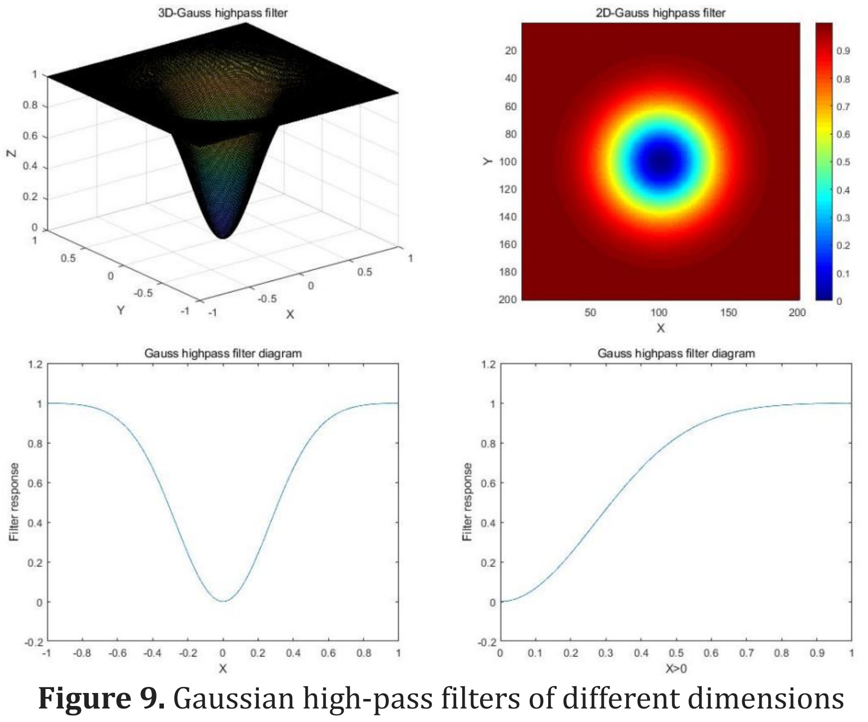


Figure 9. Gaussian high-pass filters of different dimensions

In the spectrum of the image, the low frequency component contains most of the information of the image, while the high frequency component contains the boundary information of the image. Therefore, compared with the general gray level image, the boundary characteristics of

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

the binary image are more obvious, and the image boundary can be obtained more clearly through the high-pass filter. The optimal binary image after PCNN iteration is selected, and the spectrum information of the image is obtained by fast Fourier transform first, and the obtained spectrum information is sent to the Gauss high-pass filter to extract the high frequency components of the image, and then the boundary information of the image is obtained by Fourier inversion. Finally, the obtained image is further sharped. In order to obtain binary images and compare with traditional operators. The process flow chart is shown in Figure 10.

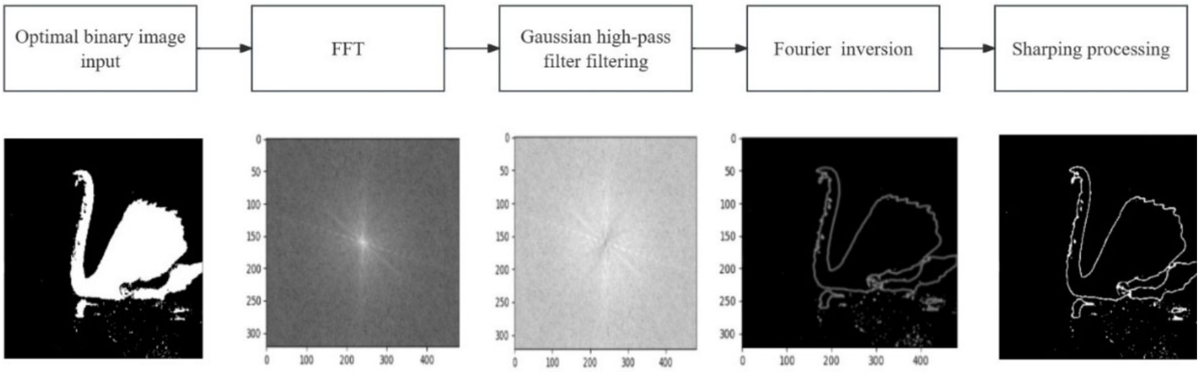


Figure 10. Edge extraction flow chart

4. EXPERIMENTAL RESULTS AND ANALYSIS

In order to verify the superiority of the edge detection algorithm proposed in this paper over traditional operators, the experimental platform was configured as follows: MATLAB R2019a and python platform were used to complete the experimental simulation on a computer with a main frequency of 2.50GHz and a memory of 12GB.

4.1. Evaluation index

The evaluation indexes used in this paper are Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), FOM (figure of merit), and Edge Localization Error (ELE).

MSE [27] is a measure of the degree of dispersion of a dataset collection, which is the sum of the squares of the difference between the predicted value and the true value of each sample divided by the average of the number of samples. A smaller MSE indicates that there is less difference between the predicted values of the model and the actual observed values, that is, the model is a better fit. The advantage of MSE is that the difference values are squared, so larger error values will have a greater impact on the fit, which helps to capture the model's prediction errors more sensitively. Its expression is:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j)-K(i,j)]^2 \tag{18}$$

PSNR [28] is an indicator to measure the quality of an image or video. It quantifies the distortion degree of an image by comparing the difference between the original signal and the compressed restored signal. The calculation expression is as follows:

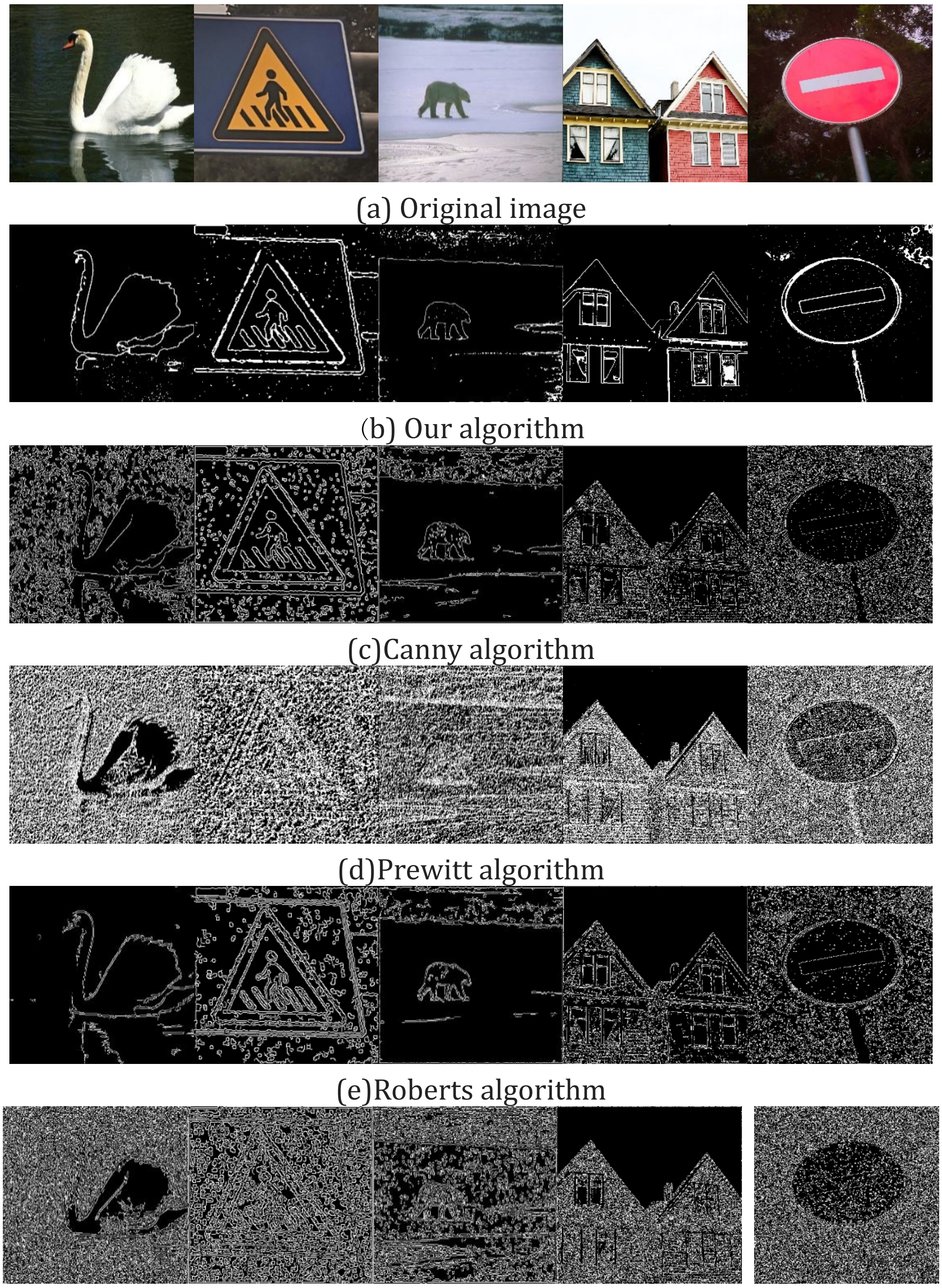
$$PSNR = 10\times\log_{10}(\frac{MAX_I^2}{MSE}) = 20\times\log_{10}(\frac{MAX_I}{\sqrt{MSE}}) \tag{19}$$

MAX represents the maximum possible pixel value of the image (usually 255), the unit of PSNR is decibels, the higher the value means the smaller the image quality loss, that is, the lower the degree of distortion.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

FOM (figure of merit) [29] is usually used to compare the advantages and disadvantages of different edge detection algorithms or technologies, and help select the best edge detection method. FOM can be used to measure the sensitivity, specificity, positioning accuracy and consistency of edge detection images. The ability of edge detection algorithm to detect real edge and restrain non-edge region is measured. Its expression is:

$$F = \frac{1}{\max\{I_I, I_A\}} \sum_{i=1}^{I_A} \frac{1}{1 + \alpha d^2(i)} \tag{20}$$



(a) Original image

(b) Our algorithm

(c)Canny algorithm

(d)Prewitt algorithm

(e)Roberts algorithm

(f)Sobel algorithm

Figure 11. Experimental results of different algorithms

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

Edge Localization error refers to the distance error between the edge position detected by the edge detection algorithm and the actual edge position, which is measured by calculating the Euclidean distance between the detected edge position and the real edge position. The smaller the edge Localization error, the more accurate the algorithm is in locating the image edge. Specifically, if the detected edge position is (x, y) and the true edge position is (x_t, y_t), the edge positioning error can be expressed as:

$$E = \sqrt{(x - x_t)^2 + (y - y_t)^2} \tag{21}$$

The edge detection algorithm proposed in this paper is compared with four traditional edge detection operators, namely Canny operator, Roberts operator, Sobel operator and Prewitt operator. When edge detection is carried out by different algorithms, the same operation is carried out on the data set, and the same evaluation index and code are used to calculate all edge detection methods. The comparison of different algorithms is shown in Figure 11, and the experimental results are shown in Table 1.

Table 1. Comparison of data results of five edge detection algorithms

Method	MSE	PSNR	FOM	ELE
Canny	0.186	7.289	0.133	30.623
Sobel	0.528	2.771	0.059	67.835
Roberts	0.147	8.317	0.159	25.413
Prewitt	0.639	1.941	0.060	54.804
Our method	0.127	8.942	0.171	23.508

As can be seen from the visualization in Figure 11, the traditional edge detection operator is vulnerable to the attack of noise, has poor effect on image edge detection, and contains obvious noise. In Table 1, from the data of MSE, PSNR, FOM and ELE, it can be seen that the edge detection algorithm in this paper is significantly superior to the traditional operator, and can completely identify clear edge images and accurately locate detailed edges, with less noise interference.

5. CONCLUSION

Noise degrades image quality and makes it difficult for edge extraction. Because of the poor anti-noise performance of traditional edge detection operator, it is easy to be attacked by noise. In this paper, a filter image edge detection algorithm based on PCNN model is proposed. By using the spectrum component information of the image, the optimal binary image is fed into the Gaussian high-pass filter through PCNN iteration to extract the image edge information. The median filter and PCNN model in the algorithm reduce the noise to the maximum extent, and the algorithm model is simplified, and the consumption of computer resources is small. Experiments show that the proposed algorithm can generate high-resolution and detail-rich edge graphs. Compared with traditional edge detection algorithms, the proposed algorithm has remarkable anti-noise performance, and can completely retain image edge information from noise-polluted images, achieving the best balance between anti-noise and accuracy. The disadvantage is that PCNN model needs to manually set parameters for image iterative processing to adapt to image processing in different fields. For images with different gray distribution, more research is needed on how to achieve the adaptive setting of this main parameter, and the accuracy of the edge detection algorithm also needs to be further improved to apply it to a wider range of fields.

ACKNOWLEDGEMENTS

This paper was supported by the National Natural Science Foundation of China (No. 61961037) and the Industrial Support Plan of Education Department of Gansu Province (No. 2021CYZC-30).

REFERENCES

[1] Deng, Xiangyu, et al. "PCNN double step firing mode for image edge detection." Multimedia Tools and Applications 81.19 (2022): 27187-27213.

[2] El-Arafy, Reda A., et al. "Using edge detection techniques and machine learning classifications for accurate lithological discrimination and structure lineaments extraction: a comparative case study from Gattar area, Northern Eastern Desert of Egypt." Arabian Journal of Geosciences 16.11 (2023): 619.

[3] Acharya, Kuldip, and Dibyendu Ghoshal. "Modified von Neumann neighborhood and taxicab geometry-based edge detection technique for infrared images." International Journal of Wavelets, Multiresolution and Information Processing (2023): 2350027.

[4] Zhou, Changsheng, et al. "Multi-scale pseudo labeling for unsupervised deep edge detection." Knowledge-Based Systems (2023): 111057.

[5] Acharya, Kuldip, and Dibyendu Ghoshal. "Modified von Neumann neighborhood and taxicab geometry-based edge detection technique for infrared images." International Journal of Wavelets, Multiresolution and Information Processing (2023): 2350027.

[6] Al-Mansor, Ebtesam, et al. "Medical image edge detection in the framework of quantum representations." Alexandria Engineering Journal 81 (2023): 234-242.

[7] Yadav, Navdeep, et al. "Local diagonal maxima-minima pattern-based edge detection technique for ultrasound and digital radiography images." IETE Journal of Research 69.6 (2023): 3211-3221.

[8] Kondow, Akiko, et al. "Automated contour extraction for light-sheet microscopy images of zebrafish embryos based on object edge detection algorithm." Development, Growth & Differentiation (2023).

[9] Tao, Nana. "Enhanced Canny Algorithm for Image Edge Detection in Print Quality Assessment." Traitement du Signal 40.3 (2023): 1281.

[10] Yin, Zhenyu, et al. "Edge Detection via Fusion Difference Convolution." Sensors 23.15 (2023): 6883.

[11] Kittler, Josef. "On the accuracy of the Sobel edge detector." Image and Vision Computing 1.1 (1983): 37-42.

[12] ROBERTS L G.Machine perception of three-dimensional solids [D].Massachusetts Institute of Technology, 1963.

[13] Prewitt, Judith MS. "Object enhancement and extraction." Picture processing and Psychopictorics 10.1 (1970): 15-19.

[14] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." Proceedings of the IEEE 86.11 (1998): 2278-2324.

[15] TORRE V, POGGIO T A.On edge detection [J].IEEE Transactions on Pattern Analysis and Machine Intelli gence, 1986 (2): 147-163.

[16] Konishi, Scott, et al. "Statistical edge detection: Learning and evaluating edge cues." IEEE Transactions on Pattern Analysis and Machine Intelligence 25.1 (2003): 57-74.

ISSN: 2472-3703 DOI: 10.6911/WSRJ.202402_10(2).0009

[17] Martin, David R., Charless C. Fowlkes, and Jitendra Malik. "Learning to detect natural image boundaries using local brightness, color, and texture cues." IEEE transactions on pattern analysis and machine intelligence 26.5 (2004): 530-549.

[18] Canny, John. "A computational approach to edge detection." IEEE Transactions on pattern analysis and machine intelligence 6 (1986): 679-698.

[19] Kong, Weibin, et al. "Sobel edge detection algorithm with adaptive threshold based on improved genetic algorithm for image processing." International Journal of Advanced Computer Science and Applications 14.2 (2023).

[20] Alam MS, Iftekharuddin KM, Karim MA (2020) Polarization-encoded optical shadow casting: edge detection using Roberts operator. Microwave Optical Technol Lett 6(3):190–193

[21] Balochian, Saeed, and Hossein Baloochian. "Edge detection on noisy images using Prewitt operator and fractional order differentiation." Multimedia Tools and Applications 81.7 (2022): 9759-9770.

[22] Cooper, G. R. J. "Amplitude-balanced edge detection filters for potential field data." Exploration Geophysics (2023): 1-9.

[23] Brioua, Abdelkrim, Redha Benzid, and Lamir Saidi. "Efficient Wavelet Thresholding and Wiener Filtering Association Incorporating a Median Filter Smoother Followed by R-Peaks Recovery for ECG Denoising." Traitement du Signal 40.3 (2023): 1211.

[24] Deng, Xiangyu, Chunman Yan, and Yide Ma. "PCNN mechanism and its parameter settings." IEEE transactions on neural networks and learning systems 31.2 (2019): 488-501.

[25] Deng, Xiangyu, HaiYue Yu, and XiKai Huang. "Time Domain Characteristic Analysis of Non-coupled PCNN." (2023).

[26] Long, Jiale, et al. "Suppressing the Zero-Frequency Components in Single Quantitative Phase Imaging by Filtering the Low-Frequency Intrinsic Mode Function Components." Photonics. Vol. 10. No. 7. MDPI, 2023.

[27] Hodson T O, Over T M, Foks S S. Mean squared error, deconstructed [J]. Journal of Advances in Modeling Earth Systems, 2021, 13(12): e2021MS002681.

[28] Huynh-Thu Q, Ghanbari M. The accuracy of PSNR in predicting video quality for different video scenes and frame rates [J]. Telecommunication Systems, 2012, 49: 35-48.

[29] Abdou I E, Pratt W K. Quantitative design and evaluation of enhancement/thresholding edge detectors [J]. Proceedings of the IEEE, 1979, 67(5): 753-763.