Design and Implementation of Image Edge Detection Algorithm Based on FPGA

YuMu Zhang[1,a], Fan Gong[1,b*]

[1]School of Intelligence and Electricity, Guangzhou Institute of science Technology, Baiyun, Guangzhou, China
[a]154808589@qq.com, [*b]corresponding author: 453687893@qq.com

ABSTRACT

The purpose and significance of this paper is to study Laplace, Robert, prewwit and Sobel edge detection algorithms respectively, The advantages and disadvantages of these four algorithms are illustrated by comparison. In order to achieve this purpose, this paper proposes four detection algorithms that can be implemented in FPGA, Some improved methods are proposed for the traditional Laplace algorithm and Sobel algorithm, It is verified by video image test. The verification conclusion is that Sobel algorithm has the best detection effect among the four algorithms, and Laplace algorithm has the worst detection effect, At the same time, it is also verified that the improved Laplace algorithm can improve the detection accuracy, The improved Sobel algorithm can improve the detection speed.

Keywords: *FPGA; Video processing; image processing*

1.INTRODUCTION

One of the basic features of image is edge,which plays a very important role in image analysis and visual analysis. The main reason is that when people want to use effective information to identify objects, they need to use edges. Edge is an important feature extraction method for pattern recognition and image analysis.In an image, the critical point of the image refers to the meaning of "edge".Different brightness on the image represents different regions.When one region changes towards another, there will be "edge", that is, the "critical" of two different regions. There are many pixels at the "critical" point, and these different pixels form many coordinates. Using the gray value of the image to detect the edge of the image is the practice of most commonly used edge detection algorithms. These commonly used edge detection algorithms have one thing in common is to detect the gray value of the image, that is,the gray change in the neighborhood of each pixel.The calculation formula used in the algorithm is mainly the first or second derivative, and its law is used for edge detection [1].

Now there are many different methods to achieve edge detection. Many classical detection algorithms have been designed, and many researchers have put forward many improved methods for the classical detection algorithms, so it has always been a research hotspot in the image field.These improved methods hope that the designed detection algorithm has more accurate positioning,less noise,and will not miss detection and false detection.

2.DESIGN OF EDGE DETECTION ALGORITHM

2.1. Laplace edge detection and its improvement

Laplace algorithm is called the simplest isotropic differential algorithm, which is characterized by rotation invariance. The Laplace transform of a two-dimensional image function is the isotropic second derivative [2], which is defined as:

$$\nabla^2 f(x,y)=\frac{\partial^2}{\partial x^2}f(x,y)+\frac{\partial^2}{\partial y^2}f(x,y) \tag{1}$$

The equation is expressed as a discrete form more suitable for digital image processing:

$$\begin{cases}\Delta^2 f(x,y)=f(x+1,y)+f(x-1,y)+f(x,y+1)+f(x,y-1)-4f(x,y)\\ \Delta^2 f(x,y)=f(x-1,y-1)+f(x,y-1)+f(x+1,y-1)+f(x-1,y)+f(x+1,y)\\ \quad +f(x-1,y+1)+f(x,y+1)+f(x+1,y+1)-8f(x,y)\end{cases} \tag{2}$$

According to the above formula, the following three common templates can be obtained, as shown in Figure 1.

A. El-Hashash et al. (Eds.): IEIT 2022, ACSR 100, pp. 816–821, 2023.
https://doi.org/10.2991/978-94-6463-058-9_128

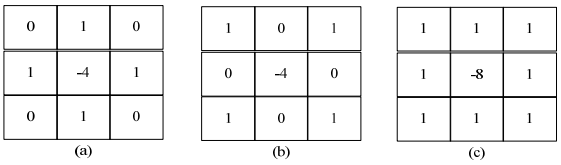

Figure 1: Laplace algorithm template.

The above three common templates have their own advantages and disadvantages， (a) the template can only extract horizontal and vertical edges; (b) The template can only extract 45 ° and 135 ° edges; (c) The template can extract horizontal, vertical, 45 ° and 135 ° edges. The extraction effect is better in the three templates, but the first two are faster. Through the above analysis, the accuracy of the three templates is not high,so the Laplace algorithm is improved.The improved idea is to improve the accuracy by extracting edges in more directions.Figure 2 is an improved new template.

0	1/8	0	1/8	0
1/8	1/2	1	1/2	1/8
0	1	-7	1	0
1/8	1/2	1	1/2	1/8
0	1/8	0	1/8	0

Figure 2: Improved Laplace algorithm template.

This improved algorithm is based on the (c) template and adds eight directions of edge extraction, such as 22.5 °, 67.5 °, 112.5 °, 157.5 ° [3]. In the improved Laplace algorithm, the formula is as follows:

$$\begin{aligned}&\nabla^2 f(x,y)=\frac{\partial^2}{\partial x^2}f(x,y)+\frac{\partial^2}{\partial y^2}f(x,y)=7f(x,y)-[f(x,y-1)+f(x+1,y)+f(x-1,y)+f(x,y+1)]\\&-\frac{1}{2}[f(x-1,y-1)+f(x+1,y-1)+f(x+1,y+1)+f(x-1,y-1)]\\&-\frac{1}{8}[f(x+1,y-2)+f(x+2,y-1)+f(x+2,y+1)+f(x+1,y+2)+f(x-1,y-2)+f(x-1,y+2)\\&+f(x-2,y+1)+f(x-2,y-1)+f(x-1,y-2)]\end{aligned} \quad (3)$$

The above calculation template can improve the edge detection accuracy and avoid the extraction of false edges. Since Verilog cannot perform floating-point calculation, all the numbers in the template are first expanded by 16 times, and then shifted to the right by 4 bits, so there is no floating-point number. For example, 1 / 8 becomes 2, and 1 / 2 becomes 8, as shown in Figure 3.

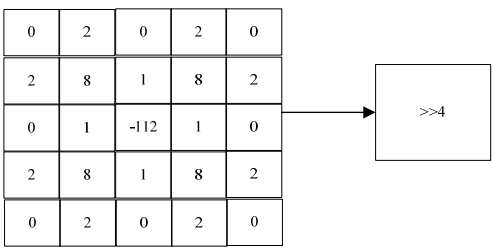

Figure 3: Improved Laplace algorithm template.

The improved Laplace algorithm is implemented in Verilog language. The core code is as follows.

```
des[2,2]=(src[0,1]*2+src[0,3]*2+src[1,0]*2+src[1,1]*8+
src[1,2]+src[1,3]*8+src[1,4]*2+src[2,1]src[2,2]*112+sr
c[2,3]+src[3,0]*2+src[3,1]*8+src[3,2]+src[3,3]*8+src[3,
4]*2+src[4,1]*2+src[4,3] *2)>>4.
```

As shown in the above code, the program directly uses the multiplication operation instead of the left shift operation, mainly the window coefficient 112, which can not be realized by the shift operation. Of course,the left shift operation can be used for multiplication 2 and multiplication 8,and only multiplication 112 is retained,but the multiplication operation is still used,so there is no conversion in this paper. Although this design uses more hardware resources, the advantage is that the detection accuracy will be improved, and the comparison of experimental results is reflected in the test.

2.2. Robert edge detection

The Roberts algorithm uses a gradient detection method, and the mathematical function expression is:

$$g(x,y)=\left\{\left[\sqrt{f(x,y)}-\sqrt{f(x+1,y+1)}\right]^2+\left[\sqrt{f(x+1,y)}-\sqrt{f(x,y+1)}\right]^2\right\}^{\frac{1}{2}} \quad (4)$$

In the formula, the square root operation is used for f(x, y), which is somewhat similar to the implementation process of the human visual system. In fact, the Roberts algorithm is an algorithm that uses the local difference method to edge. The Roberts gradient algorithm uses the difference between two adjacent pixel values in the diagonal direction, so the difference is used to replace the first-order partial derivative. The algorithm form can be expressed as follows:

$$\begin{cases}\Delta_x f(x,y)=f(x,y)-f(x-1,y-1)\\ \Delta_y f(x,y)=f(x-1,y)-f(x,y-1)\end{cases} \quad (5)$$

The two 2*2 calculation templates commonly used in the above algorithm are shown in Figure 4. In practical use, in order to avoid negative values in the calculation, the absolute value is usually used, and the pixels in the image generally use these two templates for convolution operations.

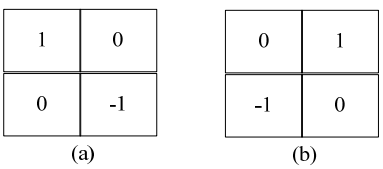

Figure 4: Robert's algorithm template.

Since the template of Robert's algorithm is relatively simple,there is no need to further improve it,but its operation involves the square root, and its resources are very precious for FPGA,so it will inevitably slow down the operation speed, but remove the square root and use the addition directly,the accuracy will drop a lot, and

false edges will appear, which is not advisable,so this article directly uses the above template for design.

The core design code of Verilog language is as follows:

```
dx=src[1,1]-src[2,2]; // Color1-Color4
dy=src[1,2]-src[2,1]; // Color2-Color3
dx=abs( dx ); // |dx|
dy=abs( dy ); // |dy|
des[1,1]=sqrt(dx*dx+dy*dy );
```

As shown in the above code, dx is the difference between color 1 and color 4, dy is the difference between color 2 and color 3, then take the absolute value of dx and dy, and finally find the square root, the result is des[1,1] color value (edge).

2.3. Prewwit edge detection

The Prewitt algorithm is an algorithm that uses the local difference averaging method to find edges. It represents the average calculation idea of the difference between the pixel values of three pairs of pixels. Usually, the average can reduce or eliminate noise. The algorithm is to average first, then find Difference, the core of the design is to average the difference to find the gradient. Use the difference to replace the first-order partial derivative to obtain the algorithm calculation formula :

$$\begin{cases}\Delta_x f(x,y)=[f(x+1,y+1)+f(x,y+1)+f(x-1,y+1)]-[f(x+1,y-1)+f(x,y-1)+f(x-1,y-1)] \\ \Delta_y f(x,y)=[f(x-1,y-1)+f(x-1,y)+f(x-1,y+1)]-[f(x+1,y-1)+f(x+1,y)+f(x+1,y+1)]\end{cases} \quad (6)$$

Two commonly used templates of the Prewit algorithm are shown in Figure 5. The pixels in the image are convolved with the template to obtain the maximum value as the output pixel, so that the Prewit algorithm can also generate edge images.

1	1	1
0	0	0
-1	-1	-1

(a)

-1	0	1
-1	0	1
-1	0	1

(b)

Figure 5: Prewit algorithm template.

According to the above two calculation templates, the core code is as follows.

```
dx =(src[0,0]+src[0,1]+src[0,2])-(src[2,0]+src[2,1]+
src[2,2] )
dy=(src[0,2]+src[1,2]+src[2,2])-(src[0,0]+
src[1,0]+src[2,0])
dx = abs( dx );
dy = abs( dy );
des[1,1] = sqrt( dx*dx + dy*dy );
```

As shown in the code above, dx is the sum of colors 1, 2, and 3 minus the sum of colors 7, 8, and 9, and dy is the sum of colors 3, 6, and 9 minus the sum of colors 1, 4, and 7, and then Then take the absolute value of dx and dy, and finally take the square root, the result is the color value (edge) of des[1,1].

2.4. Sobel edge detection

Sobel algorithm is often used in edge detection. Discrete algorithm is used to calculate the gray approximation of brightness by differential calculation. The mathematical model of Sobel algorithm adopts the first derivative, and the calculation method of local average is adopted in the algorithm, so the use of this operator can play a certain smoothing effect and eliminate some noise. Sobel algorithm can be represented by matrix, and usually adopts two groups of 3x3 templates, one is horizontal template and the other is vertical template [4]. In this way, both templates need to convolute with the image information. Then, the transverse brightness difference value and the longitudinal brightness difference value are obtained respectively. Generally, the following template can be used to implement Sobel algorithm during calculation.

Horizontal edge transverse formwork:

$$G_x=\begin{bmatrix}1 & 2 & 1\\ 0 & 0 & 0\\ -1 & -2 & -1\end{bmatrix} \quad (7)$$

Vertical edge longitudinal formwork:

$$G_y=\begin{bmatrix}-1 & 0 & 1\\ -2 & 0 & 2\\ -1 & 0 & 1\end{bmatrix} \quad (8)$$

To get the sum of GX and Gy, we need to combine the two templates. The gradient can be calculated by the following formula.

$$G=\sqrt[2]{G_x^{\,2}+G_y^{\,2}} \quad (9)$$

In actual use, absolute values are often used instead.

$$G\approx|G_x|+|G_y| \quad (10)$$

The gradient direction can be calculated by the following formula.

$$\theta=\arctan(\frac{G_y}{G_x}) \quad (11)$$

Sobel algorithm is based on 3 × 3The edge detection algorithm implemented by the matrix of 3, like the Gaussian filtering algorithm, uses the convolution template for convolution operation, but this algorithm is relatively simple to implement. The convolution template here is essentially a weighting coefficient matrix. Because the weighting coefficients of Sobel algorithm are simple to implement and the general coefficients are fixed, even if the multiplication operation is used, the hardware will not call the

embedded multiplier, but the macro module of the hardware will be used for multiplication, which consumes more logical resources of the system. Therefore, the multiplication 2 operation can be changed to the left shift operation. In this way, there are relatively simple addition, subtraction and shift operations left for the implementation of Sobel algorithm. This design can save the hardware resources of the system [5]. The following figure is the calculation template of Sobel algorithm implemented in hardware.

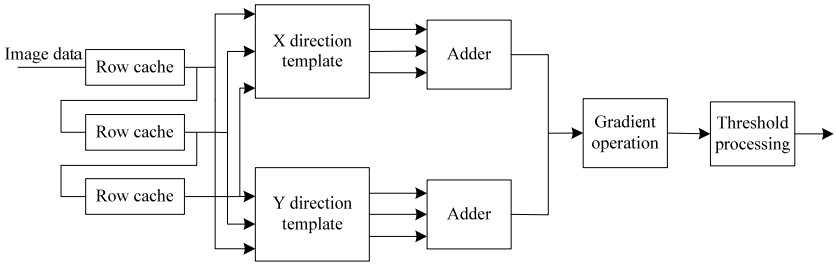

Figure 6: FPGA implementation block diagram of Soble algorithm.

The specific calculation process of the hardware implementation of Sobel algorithm on FPGA is shown in Figure 6. After the data passes through the row buffer module, three rows of image pixels are obtained. These pixels are output at the same time, respectively through two templates in the horizontal and vertical directions, and then through convolution operation, two gradient vectors of the image in the horizontal and vertical directions are obtained, and the gradient amplitude is calculated from the two gradient vectors. The methods used in the algorithm processing module are pipeline design. The shift operation, register cache and delay, addition and subtraction operation of each level are timing operations synchronized with the clock signal. Then, with template simplification, the two-dimensional convolution template can be transformed into two templates, one is a one-dimensional row template and the other is a one-dimensional column template, which can simplify the operation and save hardware logic resources.

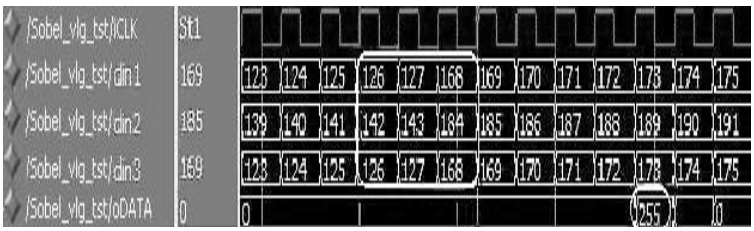

Figure 7: Sobel algorithm simulation diagram.

Figure 7 is the simulation waveform, and 255 is the output result of Sobel algorithm,The algorithm adopts the synchronous sequential circuit design of 5-stage pipeline,The processing process of the algorithm only needs 5 clocks to calculate the final result,This is also the advantage of FPGA parallel processing, which realizes hardware acceleration with the underlying method.

3.DESIGN OF EDGE DETECTION ALGORITHM

The image algorithm module system is shown in Figure 8. When designing the algorithm module system, as long as different algorithms are added, different effects can be achieved, such as median filter, Gaussian filter and edge detection algorithm. Each algorithm module calls the gray processing module, and then carries out algorithm processing.

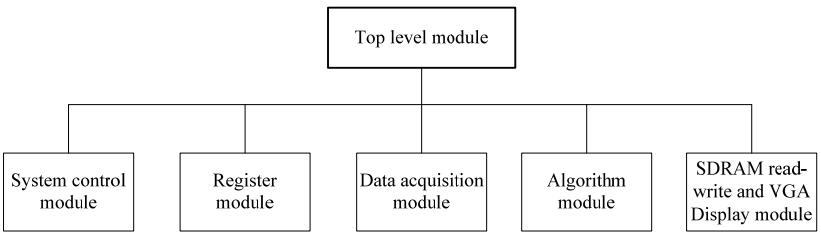

Figure 8: System block diagram of image algorithm module.

In order to compare the source image and the target image, the image algorithm system also makes some modifications to the display system. The source image is displayed in the upper left corner of the screen and the target image is displayed in the upper right corner of the screen. The implementation process is shown in Figure 9.

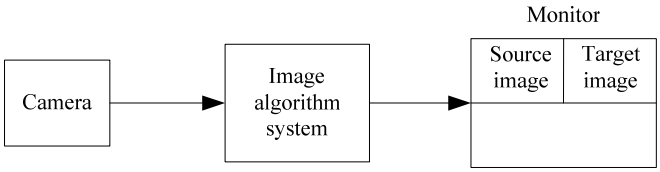

Figure 9: Image algorithm display process diagram.

Edge detection display effect:

The system tests the Laplace algorithm, Robert algorithm, Prewwit algorithm and Sobel algorithm respectively, and the results are shown in Figures 10,11,12,13,and 14.

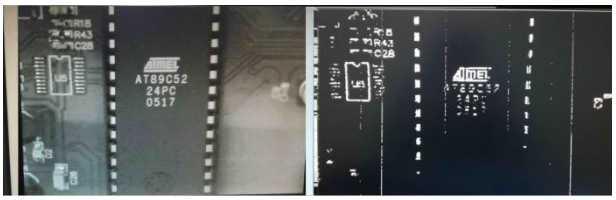

Figure 10: Laplace algorithm renderings.

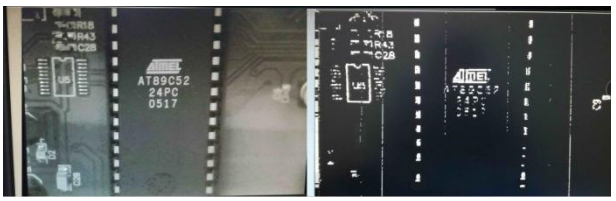

Figure 11: Improved Laplace algorithm renderings.

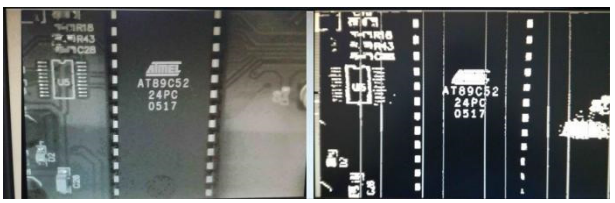

Figure 12: Robert algorithm renderings.

Figure 13: Prewwit algorithm renderings.

Figure 14: Sobel algorithm renderings.

Experimental conclusion:

- As shown in Figure 10 and Figure 11, the normal Laplace algorithm is able to detect major edge lines, but with a whole bunch of random little noise. The edge detection of the improved Laplace algorithm is stronger, indicating that the accuracy has been improved and the purpose of improvement has been achieved, but there is still a lot of random noise. The algorithm itself is sensitive to noise and cannot be eliminated;

- Robert's algorithm has a good effect on noise suppression, but the detection of edge lines is very weak, and the lines are very dark. Robert's algorithm has strong denoising ability, but the detection effect of object edges is weak. Unless the edge of the source image has a sharp outline, the Robert algorithm The detection effect of the algorithm is not good;

- The Prewwit algorithm has very obvious edge lines, and the detection effect is very good. At the same time, it also suppresses noise well. Compared with the previous two algorithms, the detection effect is much better;

- The Sobel algorithm is generally similar to the Prewwit algorithm in terms of noise suppression and operation speed, but the performance on the edge line is slightly stronger, because the latter is an improved version of the former, and the detection effect is the best among the four algorithms;

4.CONCLUSIONS

To sum up, the image edge detection algorithm based on FPGA designed in this paper meets the expected target requirements and meets the processing effects of various algorithms. The detection accuracy of the improved Laplace algorithm has also been greatly improved, and the improved Sobel edge detection can improve its detection speed. By comparing the four edge detection algorithms, it is proved that the overall detection effect of Sobel algorithm is the best. Through these tests, it is proved that the scheme designed in this paper is indeed feasible.

ACKNOWLEDGEMENTS

This article is one of the phased achievements of provincial key scientific research projects "Research on ARM-based Portable Face Recognition System" (2018KQNCX388).

REFERENCES

[1] Han,Bin.Yu,Xiaoyu.Zhang,leiming(2016).Detailed explanation of FPGA design skills and case

development[M].Beijing: Electronic Industry Press,44-45.

[2] Mou,Xingang.Zhou,Xiao.Zheng,Xiaoliang(2017).Principle and application of digital image processing based on FPGA [M]. Beijing: Electronic Industry Press,33-34.

[3] Said,Y.Smach,F(2012).Embedded real-time video processing system on FPGA[C].5th International Conference on Internet Surveillance and Protection (ICISP),85-92.

[4] Wei,Xiaohui(2015).Design and research of real-time video image acquisition and VGA display system based on FPGA [D]. Xi'an: Xi'an University of Electronic Science and technology.

[5] Zhang,Haiqing(2010).Research on key algorithms and hardware implementation of image processing system based on FPGA [D].Chongqing: Chongqing University.