International Journal of Multimedia and Ubiquitous Engineering
Vol.11, No.2 (2016), pp.297-308
http://dx.doi.org/10.14257/ijmue.2016.11.2.29

The Real Time Infrared Image Acquisition and Processing System Design Based on FPGA

Fan Jianying, Cui Xin, Fan Zhigang and Feng Yao

The Higher Educational Key Laboratory for Measuring & Control Technology
and Instrumentations of Heilongjiang Province,
Harbin University of Science and Technology,
Harbin, China
cxfjxc@163.com

Abstract

Field programmable gate array (FPGA) has the characteristics of high speed, low power consumption, high integration, flexibility and, small size, etc. In this paper, I design a real time infrared image processing and display system based on FPGA for the requirement of real time infrared image processing, which will realize the transmission, transformation, and storage of the image information, and then complete the infrared image edge detection based on Sobel *algorithm by using this system as a platform. At last, the advantages of infrared image processing with this system compared with other methods will be verified in this paper. The experimental results show that the system spends 11.44ms on processing the colorful image whose resolution is 640×480.The system has realized the real time, high speed, stable and reliable acquisition, processing and display of infrared image and we can realize the infrared target feature extraction, recognition and tracking when we combine the system with other algorithms.*

Keywords*: FPGA; real time; infrared image acquisition and processing; edge detection*

1. Introduction

Compared to visible light image, infrared image has many advantages such as seldom affected by the environment, which can image under the environment of the night, rain and fog, not affected by the surface features, far detection distance and so on. Therefore, infrared image have broad application prospects in the military detection, search and rescue, medical diagnosis, *etc.* Infrared image also has some disadvantages such as low resolution and high noise so we need to establish a real-time infrared image acquisition and processing system to realize the rapid processing of infrared images in order to use infrared image effectively in infrared identification and tracking.

The traditional image acquisition and processing platform always use computer, or use embedded platform based on DSP or ARM + DSP dual-core architecture. The former is large, poor mobility and its speed is limited, so it cannot meet the requirements of real-time [1]. DSP and ARM are all used as processors for serial data processing, leading to slower speed in data processing [2] and limited speed in data transmission. As a result, DSP chip is difficult to meet the requirements in some applications of digital image with complex algorithms, real-time and high data flow [3].

FPGA has the rich resources of register and can change the logic function through the system reconfiguration, which makes the update or modification of the design very convenient [4]. Because the data processing by FPGA is realized on the underlying hardware and has no delay of software running, its speed is much faster than DSP and ARM. FPGA has the incomparable advantages of parallel processing compared to other platforms [5]. We can design any number of execution module in FPGA, which can perform a number of tasks at the same time. So FPGA can accomplish many tasks that the serial structure processor cannot accomplish, such as high speed processing, DSP algorithm implementation, and so on. The advantages of FPGA parallel processing are more obvious in view of the problem of large computation in the digital image processing.

This paper designed a real-time video image acquisition, processing and display platform based on the FPGA chip, Cyclone IV and realized the graying and edge detection of the infrared image. System is programmed and implemented with *verilog* hardware description language. With this method the development cycle is longer, but it has the characteristics of simple structure, saved resource and quick speed, which make the system meet the requirement of real-time better.

2. Overall Design of the System

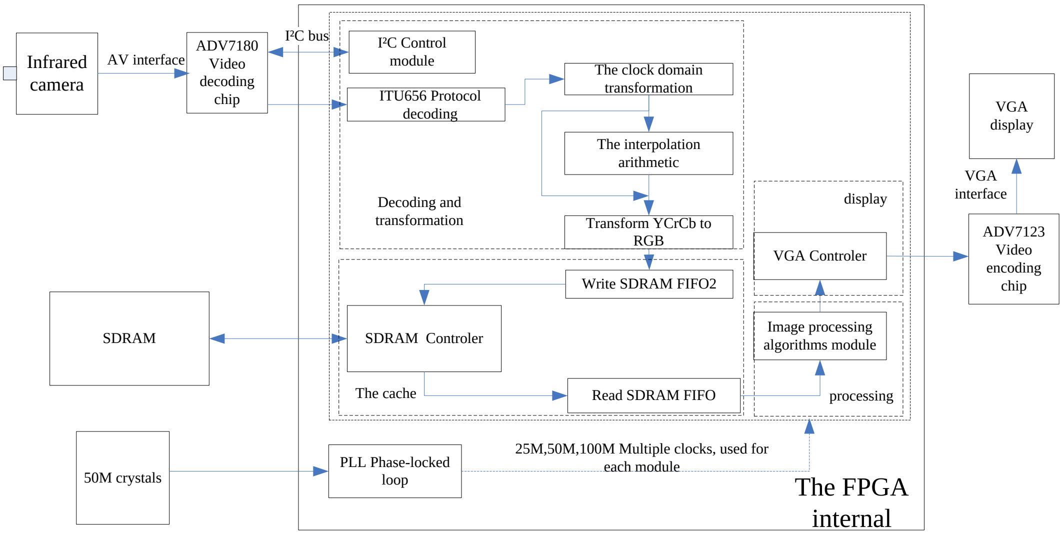


Figure 1. Block Diagram of the System Overall Design

Figure 1 is the block diagram of overall design of the system. The analog video images taken by the infrared camera will be decoded after sampling and A/D conversion of ADV7180.Then the current interlaced scanning video whose odd-even field is separate will be converted to progressive-scan video by interpolation algorithm. At the same time, transforming the current *YCrCb* format video to RGB format video and then writing the video data in SDRAM through two asynchronous FIFO. The system will carry out the read operation after the completion of the write operation and read image data from the SDRAM to the image processing algorithm module for various operations through the buffer of FIFO. Finally, the processed image data will be displayed on the VGA display.

2.1 Video Decoding and Conversion Module

The format of the output video of the infrared imaging system is PAL/NTSC analog video format, but FPGA cannot directly deal with the analog signal. So the PAL analog video output from the infrared thermal camera needs to be done A/D conversion by ADV7180 video decoder chip [6], which can decode the video stream into ITU656 standard digital video signals. The effective data quantity of a frame data is 720(column) × 576(row). Because of the resolution requirements of VGA display, it is necessary to cut the frame of image to meet the requirement of 640 × 480 display resolution. Now, the standard of the image is $YCrCb$ 4:2:2 that Y is luminance signal, Cr is chroma signal and Cb is saturation signal. In this standard, two pixels per row share the same Cr, Cb data, and the different Y data. At this point the (i, j) pixel coordinates data is $P(i,\ j) = [Y_{(i,j)}, Cr_{(i,j)}, Cb_{(i+1,j)}]^T$, ($i = 1,\ 2,\ \cdots,\ 640$, $j = 1,\ 2,\ \cdots,\ 480$),as shown in Figure 2.

Y(i,j)	Cr(i,j)	Y(i+1,j)	Cb(i+1,j)	Y(i+2,j)	Cr(i+2,j)	Y(i+3,j)	Cb(i+3,j)

Figure 2. YCrCb4:2:2 Pixel Coordinates Data Diagram

The output data from ADV7180 is 27MHz interlaced scan data. In order to facilitate real-time image processing and display, it is needed to extract the effective data and transform the time domain for the follow-up treatment of FPGA. There is a phenomenon of aliasing in the real-time processing and display of interlaced scanning data. We must restore the interlaced scanning image data to the progressive scanning image data. In this paper, we use the interpolation algorithm to achieve the progressive scanning restoration of two frames of image in the same field. I took an average of two numbers of the same column locations in two adjacent odd rows and insert the average regarded as the even row of data into the odd field. At the same time, I took an average of two number of the same position of the adjacent two even rows and insert it regarded as the odd row of data into the even field. As shown in equation (1).

$$P(i,\ j) = \frac{P(i,\ j-1) + P(i,\ j+1)}{2} \tag{1}$$

Through the operation above each odd field and even field can be reduced to a frame, which turn the image into a progressive scanning image and eliminate the sawtooth phenomenon. After the processing to remove the interlaced scanning, two fields will be converted from the $YCrCb$ data of each pixel to RGB data by multiplier module, which is convenient for the processing and display later. The formula is shown in equation (2).

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \frac{1}{256}\begin{bmatrix} 256 & & 0 \\ 256 & -88 & -183 \\ 256 & 454 & 0 \end{bmatrix}\begin{bmatrix} Y359 \\ Cb \\ Cr \end{bmatrix} + \frac{1}{256}\begin{bmatrix} -45941 \\ 34678 \\ -58065 \end{bmatrix} \tag{2}$$

2.2 Buffer Module

Because the memory storage resources of FPGA chip are limited to meet the requirements of the storage and processing of image data, it is necessary to expand the use of chip memory cache. The traditional methods often use SRAM (static random access memory) to design, but its speed is slow and the cost is high. The synchronous dynamic random access memory SDRAM device has the features of low price, large capacity and high speed. Design with SDRAM is more complex, but its usage is flexible and it can achieve higher transmission rate of image data while maximizing the saving of resources [7]. So this paper uses SDRAM to cache the image.

In this paper, we use two SDR SDRAM 1S42S16320D, 8 M×16 bit×4Banks produced by ISSI company. In order to let it work, we need to write control module in FPGA to control the SDRAM. Because the most of the current SDRAM controllers are written in the mode of burst 1/2/4/8 whose speed of reading and writing can barely meet the requirements in the high-speed processing and display of image. Therefore, this paper designs a controller that read and writes in the full page burst mode, which read and write a row of data at one operation. Because the memory cell of SDRAM is composed of capacitors, it will discharge slow after each read and write operation, which will lead to logic faulty. So a data rewriting operation covering full addresses are needed every 64ms, which is called refresh operation. However, page burst mode does not support automatic refresh and we need to send a refresh command to SDRAM for refresh operation after each read and write operation manually.

Because there are two video data to be stored in SDRAM, I define two asynchronous writing FIFO s in the writing end and a reading FIFO in the reading end. The two writing FIFO and a reading FIFO alternating read and write imitating the ping-pong operation to prevent the read and write data overflow of asynchronous FIFO caused by different clocks [8]. Two write FIFO s are deposited in the odd and even rows address. It begin to read operation when each write operation is completed and the read FIFO is not written full, which will write data into the read FIFO. Read FIFO read data to the following module according to the effective signs of the data.

2.3 Real-Time Display Control Module

In this paper, the final result will be displayed real-time. This paper designed a VGA display control module reference VGA protocol. We sent the data to the video coding chip ADV7123 according to the order of coordinate for digital analog conversion and code the data into analog signals, and then control the VGA display to display the image real-time. In this paper, I use 640×480 VGA display whose scan frequency is 60Hz according to the industrial standard of VGA [9]. Because each column pixel needs to stay 40ns, the control frequency is set to 25MHZ.

Column coordinate counting is conducted based on main frequency and the count is 800 in total. In them we pull the row sync signal $hsync$ low at the first 96 count cycle of column coordinates and pull $hsync$ high at following 48 count cycles as a signal front edge of valid display. The following 640 column is an effective display section and the last 18 count cycle is back edge of valid display.

When the column coordinates count to 800, the row coordinate count plus 1.We pull the field synchronous signal $vsync$ low in the first 2 count cycle of row coordinates and the following 33 row coordinate counter cycle is the front edge of effectively displayed data. The 480 count following show the effective display section. There are total 525 row counts. The overlap area of column valid display section and row valid display section is effective display area. Put the above image data into the ADV7123 according to the sign of effective display area and the relative coordinates in effective display area to encode and display.

We need to pull the SYNC pin low when ADV7123 chip work effectively and send the results of the AND operation between $hsync$ and $vsync$ signal to the BLANK pin at the same time. Only when we do these can the ADV7123 chip works normal.

3. The Real-Time Processing of Image

3.1 FPGA Implementation of Image Processing Algorithm

The methods of image processing based on FPGA are essentially different with other means. The RGB information of each pixel can be easily addressed because it can be defined a coordinate when we do image processing with C language and MATLAB language. But when using FPGA and hardware description language for image processing, image data read from the front end orderly cannot be addressed to each pixel by "memory" because the data stream is real time. This is the greatest difficulty to process image by FPGA.

Because most of the image processing algorithm is achieved based on 3 × 3 or greater window and it is necessary to calculate the other values around the window to deal with a certain pixel. However, the data stream is input and read in order in a hardware description language. Therefore, this paper designs shift registers as row buffers to perform the 3 × 3 window operation in order to achieve image processing with FPGA. Each buffer cache a row of data and the data in the row buffer 3 shift to the right in order. The right of the data gets into operation unit, while shifting into the row buffer 2. All operations are performed on three adjacent data every row input into the operational unit. Data shifted to the row buffer 2 continue to shift to the right of the buffer and then participate in a window operation of the next row. Continue to shift row by row to complete the cache and calculation of adjacent three rows at the moment. As shown in Figure 3.

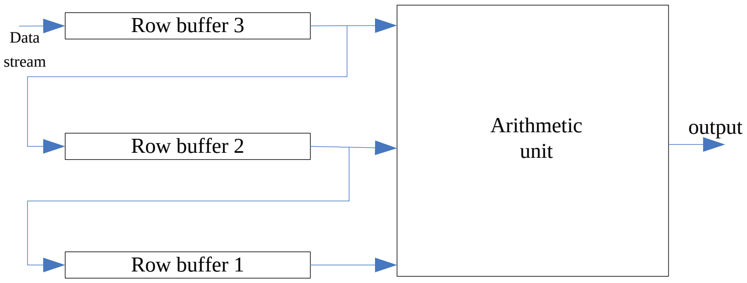


Figure 3. The Method Framework of Window Operation with FPGA

In order to process real-time, it is needed to realize the pipeline operation in the operating window, which require complete one step operation of the three around values in a clock and read the next data at the same time of the current arithmetic. Although a small amount of hardware resources are sacrificed, this method saves the clock, improves the time continuity of the algorithm and improves the efficiency, which meet the principle of “exchange area to rate” better.

3.2 Gray and Edge Detection

This paper realizes the image edge detection using *Sobel* algorithm to verify the performance of running real-time image processing algorithms by this system [10].

Let us gray image at first, because the weight of green in the grey formula reached to 58.7%, which is 2 times as red and 5 times as blue respectively. Considering the FPGA cannot carry out the floating point operation, this paper uses the G instead of R and B component to gray.

In FPGA, design 3 buffer whose length is 640.The data read from buffer each clock was taken multiplication and addition operations with the Gx and Gy ,the template of the *Sobel* operator from two directions, as shown in Figure 4. Then I use parallel adder for aggregation, as shown in Figure 5, in which P1 to P9 are the pixel values of 3 × 3 image window in FPGA.

X1=-1	X2=0	X3=1
X4=-2	X5=0	X6=2
X7=-1	X8=0	X9=1

Gx

Y1=1	Y2=2	Y3=1
Y4=0	Y5=0	Y6=0
Y7=-1	Y8=-2	Y9=-1

Gy

Figure 4. The Two Direction Template of *Sobel* Operator

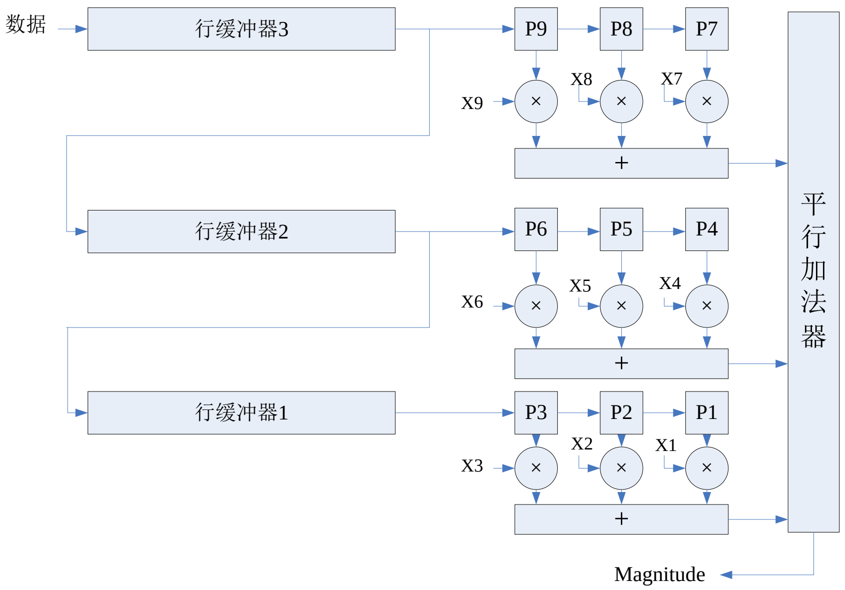


Figure 5. The Implementation Process of *Sobel* Algorithm in FPGA

The calculation formulas of *Sobel* operator of pixels $P5$ in directions of Gx and Gy [11] are

$$Magnitude(Gx) = X1\times P1 + X2\times P2 + X3\times P3 + X4\times P4 + X5\times P5 + X6\times P6 + X7\times P7 + X8\times P8 + X9\times P9$$

(3)

$$Magnitude(Gy) = Y1\times P1 + Y2\times P2 + Y3\times P3 + Y4\times P4 + Y5\times P5 + Y6\times P6 + Y7\times P7 + Y8\times P8 + Y9\times P9$$

(4)

Then we do the square root calculation using the value in the direction of X and Y. The results are as follows

$$Abs\_mag = \sqrt{Magnitude^2(Gx) + Magnitude^2(Gy)}$$

(5)

This paper compares the resulting value with the threshold and judges the finally grey value to processing and display, the formula is below.

$$data\_out = \begin{cases} 0 & Abs\_mag > THRESHOLD \\ 255 & Abs\_mag < THRESHOLD \end{cases}$$

(6)

In the Formula (6), when the square root value of $P5$ point is higher than the threshold, the output is black. Otherwise the result is white [12].

4. Development of Platform and Simulation

4.1 Development of Platform

The image acquisition part of this paper uses infrared thermal imager system *ThermoPro* TP8S of Guide Infrared Company. TP8S uses 384×288 pixels non-cooling focal plane infrared detector and the wavelength range of it is 8-14 μm. The main part of the system is designed on the DE2-115 development board of *Terasic* company. DE2-115 equipped with a FPGA chip cyclone IV EP4CE115F29C7 oriented to low cost launched by *Altera* contains 114480 logical units, 3888K bits storage units and four PLL phase-locked loop.DE2-115 uses 50MHz external crystal oscillator and a external clock interface.

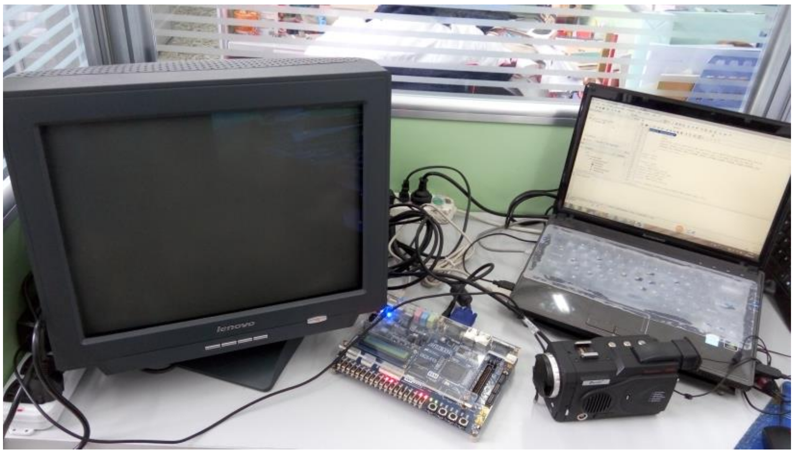
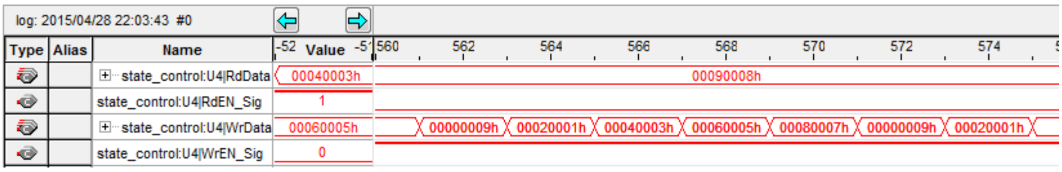


Figure 6. Physical Map of Development Platform

Figure 6 shows the development platform used in this paper. After the completion of the FPGA design, I carry on the synthesis, layout and routing, time series analysis, *etc.*

4.2 System Simulation Experiment Analysis

This paper use two pieces SDRAM chips and parallel extended to 32 *Bits* . The length of read and write one time is 320. Connecting its read and write end each to a FIFO after the design and define a counter inputs counting from 0 to 9 to test the function of write and read. Then we observe the result through the logic analyzer *SignalTap* II of *quartus* II. The results of write and read are shown in Figure 7 and Figure 8 respectively.



log: 2015/04/28 22:03:43 #0

Type	Alias	Name	Value
		⊞ state_control:U4|RdData	00040003h
		state_control:U4|RdEN_Sig	1
		⊞ state_control:U4|WrData	00060005h
		state_control:U4|WrEN_Sig	0

Figure 7. The Results of Write Data Test by SDRAM Controller

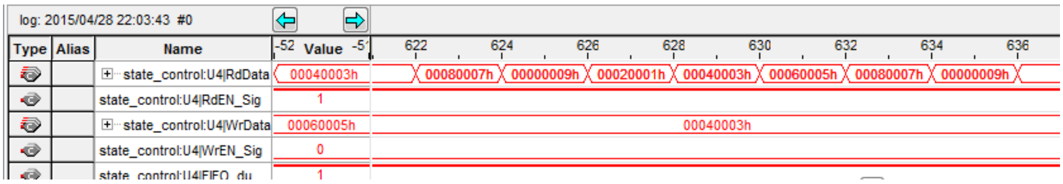


Figure 8. The Results of Read Data Test by SDRAM Controller

The SDRAM controller in the mode of full page burst has been proved to meet the functional requirements by observing the sending of command and result of reading and writing.

In the case of containing the edge detection algorithm module, the system is counted to occupying a total of 2315 logic units, 1326 register resources, accounting for 2% of the total amount of resources. In addition, the system occupies 48 embedded multiplier units, accounting for 9% of the total number of multiplier units. I simulate the system in the way of hardware through the *Quartus II* and JTAG. The results of simulation are shown in Figure 9.

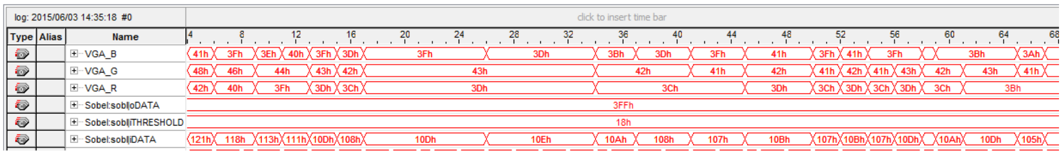

log: 2015/06/03 14:35:18 #0
click to insert time bar
Type
Alias
Name
4
8
12
16
20
24
28
32
36
40
44
48
52
56
60
64
68
VGA_B
41h 3Fh 3Eh 40h 3Fh 3Dh 3Fh 3Dh 3Bh 3Dh 3Fh 41h 3Fh 41h 3Fh 3Bh 3Ah
VGA_G
48h 46h 44h 43h 42h 43h 42h 41h 42h 41h 42h 41h 43h 42h 43h 41h
VGA_R
42h 40h 3Fh 3Dh 3Ch 3Dh 3Ch 3Dh 3Ch 3Dh 3Ch 3Dh 3Ch 3Bh
Sobel:sobl|oDATA
3FFh
Sobel:sobl|iTHRESHOLD
18h
Sobel:sobl|iDATA
121h 118h 113h 111h 10Dh 108h 10Dh 10Eh 10Ah 108h 107h 10Bh 107h 10Bh 107h 10Dh 10Ah 10Dh 105h

Figure 9. Hardware Simulation Results of Image Acquisition

I conduct the verification experiments afterwards for system with edge detection algorithm. The results of hardware simulation are shown in Figure 10.

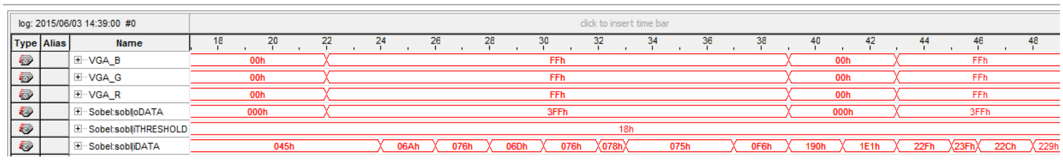


Figure 10. Simulation Results of Edge Detection Experiment

From the results of the simulation, we can know that the acquisition of image data is successful and the edge detection is carried out successfully.

5. Image Processing Experiments and Analysis

After debugging I solidify the correct and reliable program on the configuration ROM chip EPCS64, which can configure the FPGA as soon as given the power. The infrared image and grayscale image that has been interpolated and cached are displayed on the VGA display respectively, as shown in Figure 11, 12.

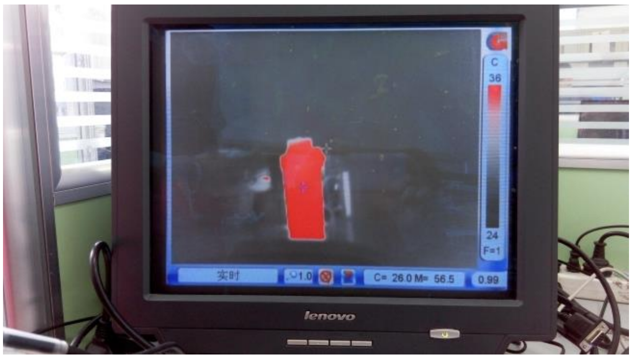


Figure 11. Infrared Image

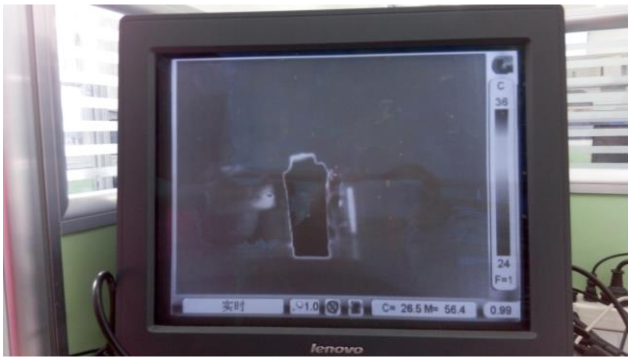


Figure 12. Grayscale Images

We can see from Figure 11 and Figure 12 that the image is displayed clearly, which prove that image has been obtained successful and displayed real-time. The effect of gray scale image is good and the contour features of target are clear, which is convenient for edge detection and feature extraction.

Sobel Algorithm has been used to extract the edge of the gray image in order to verify the performance of the system to achieve real-time image processing. The results are shown in Figure 13. In order to determine the time *Sobel* algorithm module running, I define a flag variable inside the FPGA algorithm module and pull high and low for a number of clock cycles at the starting and the end of processing of a frame of image. The variable is led out to observe the results using the oscilloscope. As shown in Figure 14.

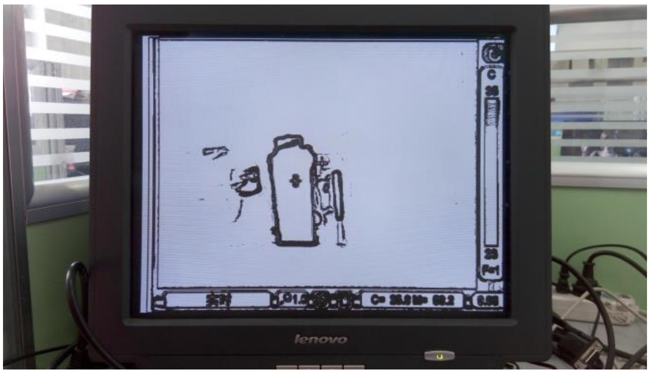


Figure 13. The Result of Edge Detection Results

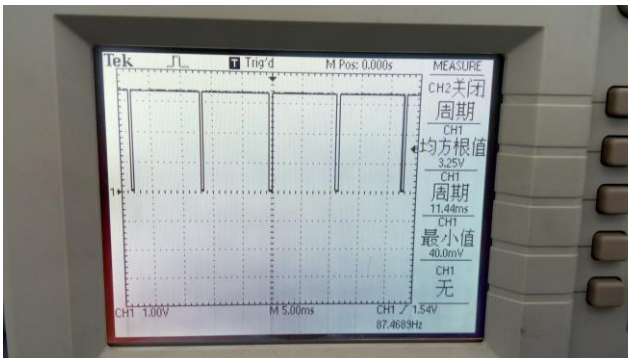


Figure 14. The Cycle of Edge Detection

Edges extracted are clear and there is less noise in them. Dynamic image have not the phenomenon of delay and sawtooth, satisfying the processing requirements of high speed and stable. If going through processing of image filter, it will be suitable for subsequent feature extraction, recognition and tracking.

Algorithm module uses a 27MHz clock given by video decoding chip, convenient for being synchronous with video streams acquired original. The oscilloscope shows that the time for one time edge detection of a picture using this system is 11.44ms.If we adopt a clock of higher frequency output by the PLL, the speed will be faster. At last, this paper takes the comparison experiments of edge detection respectively with Visual Studio 2013 and Matlab2010a on PC platform with 2.27GHz CPU, 2G memory. The experimental results are shown in Table 1.

Table 1. Contrast Experiment of Infrared Image Edge Detection

	PC platform+VS2013	PC platform + Matlab	FPGA platform +Verilog
Infrared image 1	15.16 ms	22.16 ms	11.44 ms
Infrared image 2	15.28 ms	24.80 ms	11.44 ms

6. Conclusion

This paper designs an infrared image processing system based on FPGA and realizes the real-time acquisition, cache and display of the infrared video image in technology and then realizes the gray-scale and edge detection of image on the basis of this system platform. The collected infrared images have been displayed clearly. The results of edge detection are clear and it made little noise, which achieves the desired design goals. By doing comparative experiments I compare the speed of this system with image edge detection by MATLAB and vs2013 on PC platform. The experimental results show that image edge detection taken by this system has a faster speed without delaying. It meets the requirements of real-time, highly efficiency and stability. The system is reliable and stable. The resource occupancy of the system is only 2%.The construction of this system is the basis for the following research, such as infrared target recognition, tracking, and so on.

References

[1] F. Weichang, L. Yuchi and H. Dong, "Dual channel image real-time processing system based on FPGA", Chinese journal of sensors and actuators, vol. 23, no. 8, **(2010)**, pp. 1116-1121.

[2] G. Yongcai, S. Yuwei and G. Chao, "Design and implementation of real time infrared image collection system based on FPGA", Chinese Journal of Scientific Instrument, vol. 32, no. 3, **(2011)**, pp. 515-518.

[3] Y. Fan, Z. Hao and M. Xinwen, "Image processing system based on FPGA", J. Huazhong Univ. of Sci. & Tech., vol. 43, no. 2, **(2015)**, pp. 119-120.

[4] W. Haijun, H. Dongjian and X. Shangzhong, "The Realization of Image Median Filter Algorithm Based on FPGA", Micro computer information, vol. 24, no. 7-3, **(2008)**, pp. 280-281.

[5] Z. Yidan and F. Yibing, "Image acquisition and VGA display system based on FPGA", Journal of Computer Applications, vol. 31, no. 5, **(2011)**, pp. 1258-1259.

[6] Z. Yanbo, W. Zhulin and Z. Zibin, "TV goniometer size field transform system design based on cybic interpolation", Application of Electronic Technique, vol. 40, no. 2, **(2014)**, pp. 85-86.

[7] S. Xiaoping, "Design of an SDRAM-Based FPGA Real-Time Image Acquisition System", Journal of Southwest University, vol. 33, no. 1, **(2011)**, pp. 128-129.

[8] F. Guozhi and W. Kang, "A Kind of Asynchronous FIFO Buffer Design of High Speed Video System", Journal of Harbin University of Science and Technology, vol. 17, no. 6, **(2012)**, pp. 102-105.

[9] Z. Zhuoran and X. Daolian, "Design and Implementation of VGA Image Display System Based on FPGA", Instrument Technique and Sensor, no. 7, **(2012)**, pp. 90-91.

[10] L. Zhongming and X. Weiping, "The Experiment Research of Dynamic Image Detection Performance Evaluation", Journal of Harbin University of Science and Technology, vol. 14, no. 1, **(2009)**, pp. 62-64.

[11] Z. Lihong and L. Zhaodong, "Application of Sobel edge detection based on FPGA", Application of Electronic Technique, vol. 37, no. 8, **(2011)**, pp. 26-28.

[12] Y. Yuemei and F. Dongqin, "Research of high speed image processing system based on FPGA", Ship science and technology, vol. 37, no. 4, **(2015)**, pp. 207-209.