

Research and Practice of Traffic Lights and Traffic Signs Recognition System Based on Multicore of FPGA*

Shuhe Wang, Pan Zhang, Zhitao Dai, Yiwen Wang, Ran Tao, Shu Sun

Beijing University of Posts and Telecommunications, School of Computer Science

Received 2012

ABSTRACT

This thesis will present the research and practice of traffic lights and traffic signs recognition system based on multicore of FPGA. This system consists of four parts as following: the collection of dynamic images, the preprocessing of gray value, the detection of the edges and the patterning and the judgment of the pattern matching. The multiple cores system is consist of three cores. Each core parallels processes the incoming images from camera collection in terms of different colors and graphic elements. The image data read in from the camera works as the sharing data of the three cores.

Keywords: Intelligent Transportation; Multicore; Image Processing; SOPC

1. Introduction

Intelligent transportation systems can accurately determine the vehicle encountered in driving the process, such as traffic lights, traffic signs, traffic changes and emergency situations, timely alert, control of vehicle deceleration, or automatically brake to avoid. Traffic lights and traffic signs recognition have importance and necessity. Prompt identification of traffic lights and traffic signs to make a relevant response to operational is difficult, and to develop a smart system that can help drivers safer driving is necessary.

The advantages of the FPGA are reflected in the various functions collection on the same chip, and can freely add functionality at various stages of the design. Combination of software resources and hardware systems, as well as FPGA hardware acceleration features to the electronic system performance is significantly improved. Multi-core technology used in traffic sign recognition can achieve our goals more quickly and efficiently.

The purpose of this paper is to achieve dynamic image recognition and processing on programmable on-chip multi-core system architecture.

a) SOPC Builder to construct the entire hardware platform, including the design of communication between nuclear and the three nuclear structure, connecting the system to complete the functionality required for the processor and peripherals, and providing the operating system and hardware environment [1].

b) Completed the acquisition of the video through the USB camera. Set the USB chip ISP1362 on host-con-

troller status, read the camera device information.

c) The image data processing. The image pixel data collected is stored in the RGB in embedded systems and fulfill its tasks through multi-core parallel computing-intensive pattern recognition [2].

2. The Building and Development of the System

This system is mainly to complete the image recognition processing module based on the multicore SOPC platform, the dynamic image recognition process development steps to shown in **Figure 2-1**. Using Gexin Science And Technology Co., Ltd. GX-SOC/SOPC-CIDE innovative experimental platform for hardware configuration to the corresponding design [3].

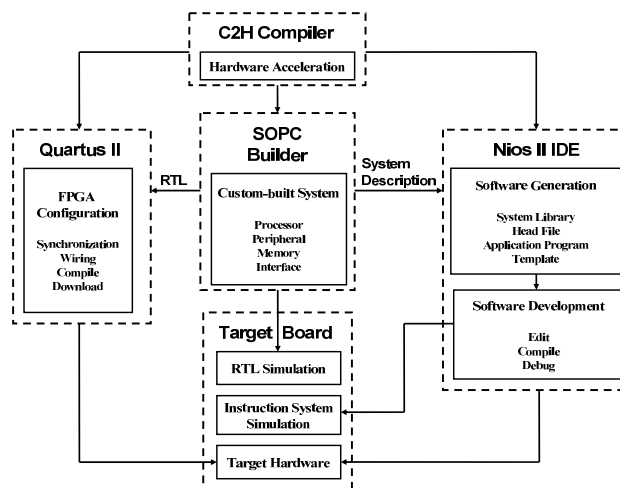


Figure 2-1. SOPC system development model.

*Sponsors: Beijing University of Posts and Telecommunications, Students' Innovative Practice Base.

SOPC Builder is responsible for the Nios II hardware system construction, including selecting and connecting components, determining the processor, configuring the storage devices and designing the different types of interface. SOPC Builder will generate the Nios II system, the Quartus II add the system to the project, fulfilling the framework of the system and downloaded it to the target board. SOPC Builder instores the Nios II hardware information in the .ptf file, the Nios II IDE can access system hardware information by the .ptf file, and generate the appropriate HAL system library and driver. Software algorithm first debugs on a single-core system, and then complements data exchange and communication between the multicore, gradually debugging successfully. After, if necessary, you can use the C2H Compiler to conduct hardware acceleration. Finally, the Nios II IDE will generate the Flash file to download to the target hardware [4].

This is the complete development process of the SOPC system.

3. Hardware Part

Three cores are used in this multi-core systems, The first core completes recognition processing and judgment of red lights and traffic signs with a red element, the second core completes recognition processing and judgment of yellow lights and traffic signs with a yellow element, the third core completes recognition processing and judgment of green lights and traffic signs with a green element as well as for the control of the peripheral response devices. The image data read by the camera will be shared data of the three cores, three cores will all do the binarization processing and pattern recognition for different colors in the image, three cores have timing sequence and will take data exchange communications with the third core to achieve effective control of the peripheral [5].

3.1. Building of the Hardware Platform

1) First, completing single-core system structures is the Core part of the entire multi-core hardware platform. Select IP cores required to add and connect for the system in the SOPC. When the multi-core hardware platform is completed, click the Generate button to generate the Nios II system module. Add and configure the phase-locked loop into the platform at last.

2) Then, we'll finish completing the multi-core system structures. Add the second and third core and clocks and rename them on the basis of the previous single-core systems. Then, add a mutex and a message buffer memory to achieve the communication between the three cores. Finishing setting the connection of multi-core shared buffer and adding the corresponding address, we

will get the three-core system [6].

3.2. The Overall System Architecture As Shown

4. Software Part

This section mainly introduces the USB camera data acquisition module, the image recognition and the achievement of processing algorithms.

4.1. USB Camera Acquisition Modules

The hassel camera is used in this paper for video signal acquisition. The video signal will be encoded to the format as prescribed by the image processing chip inside the USB camera. Here we output the pixel values in RGB format. Besides the head of the packet, the pixel values of the main body of the packet are stored into the form of "R, G , B, R, G, B ... ". The camera resolution is 640 * 480. We do not need to process all the pixel values, but to sample the image information from the acquisition. We sample the data of each frame circularly, and store into the two-dimensional array for the following processing. The system flow chart is to shown in **Figure 4-1**.

4.2. Camera Data Processing Module

First, the hardware system read the RGB values matrix information of the image from the camera. The first step, we change the RGB color space into the HSI color space,

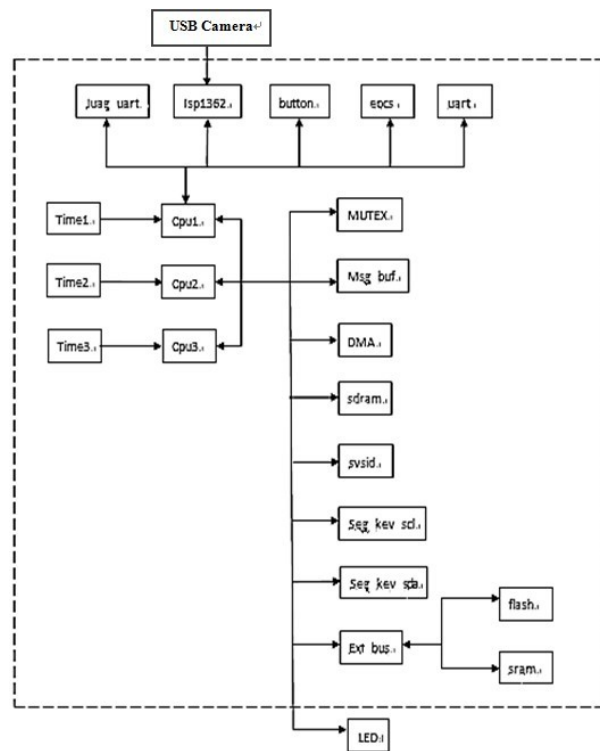


Figure 3-1. Hardware system structure.

then segment and extract color information by the threshold. The second step, we use the SOBEL operator to calculate the gradient and detect the edge of the image. The edge of the objects in the image left, while other content is filtered, then it will generate the 1, 0 matrix. The third step, we use the HOUGH discrete transform to identify the circular area through a small amount of calculation. The fourth step, we detect the RGB values to check the circle left is red or green. So we will determine the red or yellow areas of the traffic lights and of the edge of the traffic signs in the picture. The fifth step, we use the neural network algorithm for traffic sign recognition.

The conversion of color images to grayscale images is to shown in **Figures 4-2(a) and (b)** [7]:

1) Sobel Operator to Seek the Edge

One form of Sobel operator is Isotropic Sobel operator which divides into two parts. One is used for the detection of level edge, another for the detection of vertical edge. It can calculate the gradient value of each pixel. Using the fast convolution function, the Isotropic Sobel operator works simply and effectively. After calculating the gradient, we extract the edges by the threshold.(1) The gradient calculation; (2) The threshold calculation of the adaptive domain, determination of the appropriate threshold is critical to the algorithm. The iterative method is based on the idea of approximation. The results are to shown in **Figures 4-3(a) and (b)**.

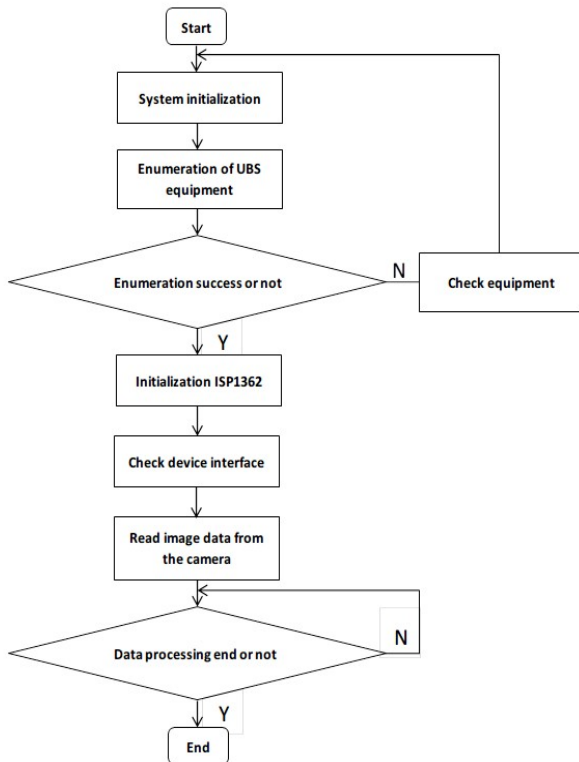


Figure 4-1. System flow chart.

2) HOUGH Transform

Equidistant select from the SOBEL algorithm to obtain the edge points, whose distance between each other is equal (the algorithm take as four points). And then take three points as a group, by three points on the circle to determine a set of center and radius. Be quantified according to the radius of the size of the packet and finally select the most and the second quantity of more than two sets of data, thus determine the approximate center and radius. This method can effectively solve the incomplete circular edge of the image noise problem. (The results are shown in **Figures 4-4A and B.**)

4.3. Speed limit Sign Recognition Based on BP Neural Network

In the selected ring area of HOUGH transform, recognition of traffic signs and speed limit sign will be taken with Artificial Neural Networks. We need to build a three-tier recognition network consists of input layer, hidden layer and output layer. We extract multiple feature of the number in the selected ring area and use them as the input of the neural network [8].



Figure 4-2. (a) before the HIS space; (b) after.

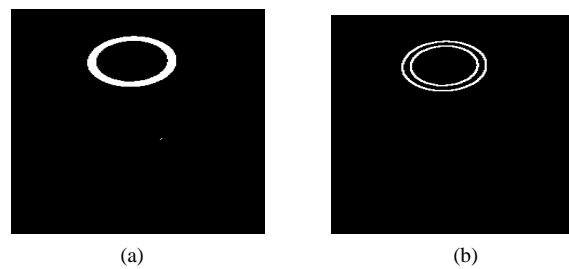


Figure 4-3. (a) before edge processing; (b) after edge processing.

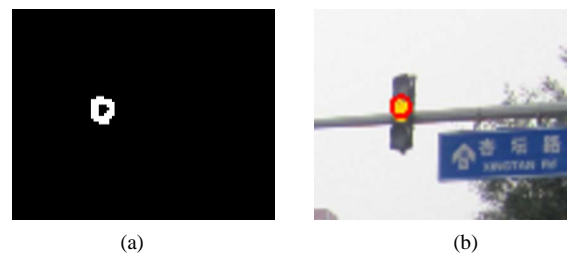


Figure 4-4. (a) before transform;(B) after transform.

After training the neural network recognition speed is better in a variety of pattern recognition programs and it saves the amount of CPU computing and is suitable for limited resources in the FPGA.

5. Summary

In recent years, Programmable System on Chip SOPC technology develops rapidly at home and abroad. This paper is based on SOPC technology for parallel multi-core and dynamic images processing and identifying on the NIOS II platform.

The paper is completed the following tasks:

a) Build up SOPC development platform with Construction of GX-SOC/SOPC-CIDE development board, including the NIOS II system design and integration of the hardware platform.

b) Process Collected data to the threshold value judgment and the binarization processing.

c) Build up the multinuclear platform; fulfill the tasks through multi-core parallel computing-intensive pattern recognition in embedded systems.

The advantage of SOPC technology and multi-core is the high flexibility of the system, which greatly curtailed product development cycle, achieving a decentralized information processing load balancing and found a new way to improve the performance of the equipment.

Though there are still some defects on SOPC system and the development and popularization of the multi-core system is still forward, its advantages can not be ignored.

I believe that with the rapid development of technology, the SOPC technology and multi-core technology will be more mature and broader development and application of space.

REFERENCES

- [1] S. Pan and J. Y. Huang, "SOPC Technique Practical Course," Tsinghua University Press 2005, pp. 10-13.
- [2] Altera Corp, "NIOS II Processor Reference Handbook," Altera, 2005, pp. 23-54.
- [3] Altera Corp, "NIOS II Software Developer'S Handbook," Altera, 2005, pp. 56-78.
- [4] Altera Corp, "Creating Multiprocessor NIOS II System Tutorial," Altera, 2005, pp. 26-46.
- [5] Altera Corp, "NIOS II Software Developer'S Handbook," Altera, 2005, pp. 12-40.
- [6] S. Yehu, O. UMIT and R. KEITH, "A robust video based traffic light detection algorithm for intelligent vehicles," *IEEE Intelligent Vehicles Symposium*, Washington, DC: IEEE Press, 2009, pp. 521-526.
- [7] D. Yang, K. Q. Li and S. F. Zheng, "Automobile Technique in Intelligent Transportation System," *Automotive Engineering Press* 2003, Vol. 25, No. 3, pp. 220-228.
- [8] T.-H. Hwang, I.-H. Joo and S.-I. Cho, "Detection of Traffic Lights for Vision-based Car Navigation System," *PSIVT 2006: Pacific Rim Symposium on Advances in Image and Video Technology*, LNCS 4319, Berlin: Springer-Verlag, 2006, pp. 682-691.
[doi:10.1109/40.285222](https://doi.org/10.1109/40.285222)