Research on Motion Image Processing System Based on FPGA and Convolutional Neural Network

Xin Nie [1],*, Xiaoyan Yang [2]

[1] Hubei Key Laboratory of Intelligent Robot, Wuhan Institute of Technology, Wuhan, China
[2] School of Science, Hubei University of Technology, Wuhan, China
* Corresponding author.

Abstract: Aiming at the problem of large amount of moving image data and difficulty in transmission, a moving image processing system based on FPGA and neural network is proposed. The hardware composition of the system is studied and the key modules such as FPGA module and storage module in the system are analyzed and designed, and an FPGA-based neural network realization method is proposed. Experimental results show that this method can achieve reliable compression of moving images. The compression effect is related to the number of neurons in the hidden layer. The more the number of neurons, the higher the signal-to-noise ratio of the compressed image.

Keywords: FPGA, convolutional neural network, moving image, image processing system.

1. Introduction

With the development of neural network technology, it has been better used in digital image compression. The self-organization and self-adaptability of neural network enable it to get rid of the limitations of predetermined data coding algorithms, and completely rely on the characteristics of the image itself for coding and compression. Two methods of software processing and hardware processing can be used to compress digital images. Software processing needs to rely on a computer platform, which is not conducive to system miniaturization. In order to achieve the integration of moving image acquisition and image compression processing, FPGA can be used to implement image compression processing [1]. FPGA has the characteristics of flexible design and convenient programming. FPGA-based image processing systems have the advantages of fast computing speed and field programmable, and are widely used in the field of image processing. This paper presents a method for implementing a motion image processing system based on FPGA and neural network. The system uses neural network technology as the image compression processing method, and analyses and designs the FPGA related circuits, which can effectively compress the moving image [2].

2. Discussion of related concepts

2.1. The Basic Structure of FPGA

FPGA is composed of several independent programmable logic modules. Users can connect these modules into the required digital system through programming. FPGA is a high-density PLD, and its integration level can reach more than 30,000 gates/chip. FPGA is composed of three programmable units and a static memory for storing programming data. These three programmable units are composed of three parts: configurable logic module CLB, output input module IOB and interconnection resource IR. The working state of FPGA is all set by the data in the programming data memory. Among them, most of the pins of the FPGA are connected to the programmable IOB, which can be set as input or output terminals as required. Therefore, FPGA devices have more input and output terminals than EPLDs of the same scale, and are more widely used in practice [3]. Each CLB contains two parts, a combinational logic circuit and a storage circuit, which can be used to design a small-scale combinational logic circuit or sequential logic circuit.

2.2. Convolutional Neural Network

CNN is a neural network structure based on a multilayer perceptron. A typical CNN model consists of an input layer, a convolutional layer, a fully connected layer, an output layer, and a classification layer, as shown in Figure 1. The image data is read by the input layer, and the convolution layer is convolved with the input image through multiple convolution kernels to generate multiple feature maps, and then the dimensionality reduction of the pooling layer extracts the feature map information. After several convolutional layers, the feature map is expanded into a vector, which is input to the fully connected layer, and the output is obtained through the matrix operation of the fully connected layer and the output layer, and then the classification probability output is obtained through the SoftMax classification layer.

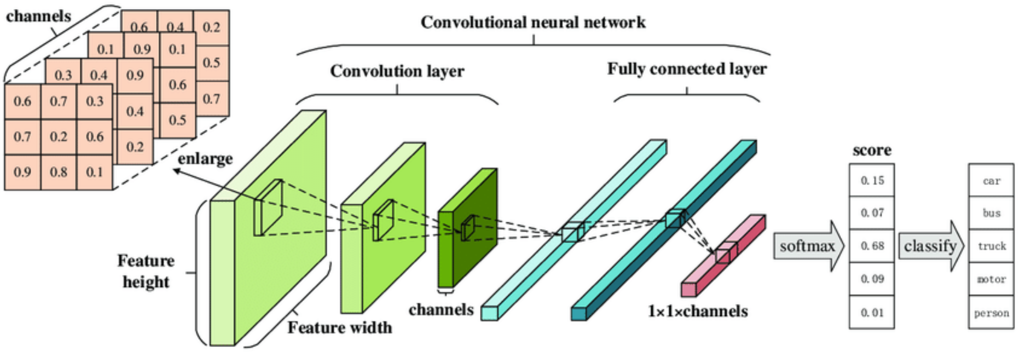

Figure 1. Typical CNN model structure

Neural network model inference reduces a certain accuracy without affecting the accuracy, and the calculation of fixed-point numbers on FPGA is more efficient than floating-point numbers. Therefore, the model parameters are converted from 32-bit floating-point numbers to 16-bit fixed-point numbers. In the method, the 16-bit fixed-point number weight is represented by an 8-bit index. The value of the index table is 28=256. Then the index table is updated through the backpropagation algorithm, and the index and the index table are stored in the external memory [4].

2.2.1 Fourier transform

Suppose $x(n)$ is a finite-length signal with a length of N points, that is, the signal is only distributed in the interval [0, N-1], and the rest are 0, then the discrete Fourier transform of the signal is defined as follows:

$$X(k)=\sum_{n=0}^{N-1} x(n)W_N^{kn}, k=0,1,2,...,N-1 \quad (1)$$

In general, x(n) is a sequence of complex numbers. For each value of k, calculating X(k) requires N complex multiplications and $N-1$ complex additions. Therefore, for N values of k, a total of N^2 complex number multiplications and $N(N-1)$ complex number additions must be calculated. That is to calculate all DFT, $N^2+N(N-1)\approx 2N^2$ operation is required. For example, when N is 1024, a total of 2097152 calculations are required. It can be seen that when $N\rangle\rangle 1$, the number of operations is quite large, and it is impossible to perform DFT analysis on the signal, whether it is implemented in software or in hardware, to achieve real-time.

2.2.2 The basic idea of convolutional neural network

The basic idea of the convolutional neural network algorithm is: divide the sequence of length N into several shorter sequences, use the periodicity and symmetry of the rotation factor W_N^m to reduce the number of DFT operations, and the periodicity and symmetry of the rotation factor W_N^m As follows:

$$W_N^{m+lN} = W_N^m = e^{-j\frac{2\pi}{N}(m+lN)} = e^{-j\frac{2\pi}{N}m} \quad (2)$$

$$W_N^{-m} = W_N^{N-m}$$
$$\left[W_N^{N-m}\right]* = W_N^m \quad (3)$$

$$W_N^{m+\frac{N}{2}} = -W_N^m \quad (4)$$

In order to facilitate analysis and comparison, first look at the calculation amount of the discrete Fourier transform of the N-point finite-length signal x (n). According to DFT calculation formula:

$$X(k)=\sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}kn}$$
$$=\sum_{n=0}^{N-1} x(n)\cos(\frac{2\pi}{N}kn) - j\sum_{n=0}^{N-1} x(n)\sin(\frac{2\pi}{N}kn) \quad (5)$$

Although the signal x(n) is generally a real signal in practical applications, assuming it is a complex signal, the calculation amount of the N-point DFT can be known from equation (5). Suppose the length of the signal x (n) is $N=2^r$, and divide it into an even sequence $x_0(n)=x(2n), n=0,1,...,N/2-1$ and an odd sequence $x_1(n)=x(2n+1), n=0,1,...,N/2-1$, so that the discrete Fourier transform X(k) of x (n) can be converted into the discrete Fourier transform $X_0(k)$ and $x_1(n)$ of $x_0(n)$ The discrete Fourier transform $X_1(k)$ is as follows:

$$X(k)=\sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}kn}$$
$$=\sum_{r=0}^{N/2-1} x(2r)e^{-j\frac{2\pi}{N}k(2r)} + \sum_{r=0}^{N/2-1} x(2r+1)e^{-j\frac{2\pi}{N}k(2r+1)} \quad (6)$$
$$=\sum_{r=0}^{N/2-1} x(2r)e^{-j\frac{2\pi}{(\frac{N}{2})}kr} + \sum_{r=0}^{N/2-1} x(2r+1)e^{-j\frac{2\pi}{(\frac{N}{2})}kr}\cdot e^{-j\frac{2\pi}{N}k}$$

3. System Hardware Design

3.1. Overall Hardware Design

The FPGA-based moving image processing system can read the moving image into the system and process it accordingly, and then transmit the processing result to the upper computer for display through the 1394 interface.

The system is composed of FPGA, external memory, 1394 link layer chip, 1394 physical layer chip and upper computer. Its composition structure is shown in Figure 2. FPGA is used to receive moving image data and compress it according to instructions, and then transmit the processing result to the host computer. It contains an image receiving module, an external memory control module, and a 1394 control module, which are used to control the FPGA to interact with the outside world [5, 6]. The external memory is used to store the image data and intermediate processing result data passed by the FPGA during the processing process, and reduce the resource consumption of the FPGA by means of the external memory, and improve its computing efficiency. The 1394 link layer chip and the 1394 physical layer chip together form a data transmission channel, which is used to transfer image processing data and control command data between the FPGA and the host computer. The upper computer is used to display the processing results and control the operating status of the image processing system. According to the function, the system can be divided into FPGA module, storage module, transmission module and power supply module. The detailed design of each module is given below [7].

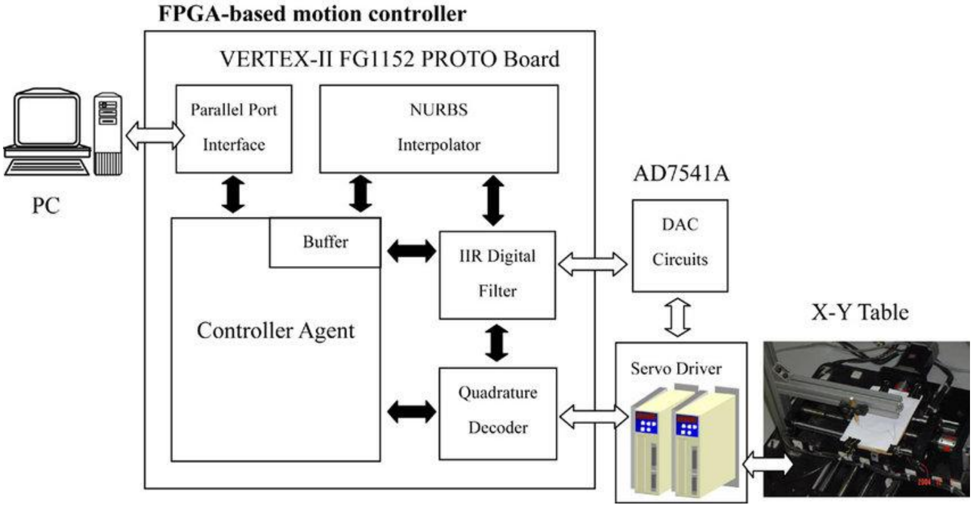

Figure 2. Schematic diagram of system composition

3.2. FPGA Module Design

Configure the FPGA to adopt the fast-active serial way. First, set the start-up mode of FPGA, and then determine the configuration mode of FPGA by driving the high and low levels of the pins. The start-up circuit is shown in Figure 3. The chip uses a standard voltage of 2.5V for power supply. The drive pins are connected to the ground network and power supply through two resistors, and the FPGA configuration mode is set by changing the resistance of the resistor. This design can adapt to different working methods, and can avoid circuit board damage caused by errors, and has better portability. FPGA has dedicated pins for external memory interface, including two types: read and write data pins and read data pins. According to the external memory standard, it can support two working modes: one-way read data and two-way data strobe [8]. When the external memory is DDR, the read data pin is only used in write mode, and the read data pin of FPGA and DDR must be connected. Address signals and control information are sent at a single data rate, and the I/O pins in the FPGA I/O block can be selected to generate addresses and control commands to the external memory. The clock signal of the external memory in the interface circuit is used to obtain the address and control commands, and the clock signal pin needs to be connected to the I/O pin.

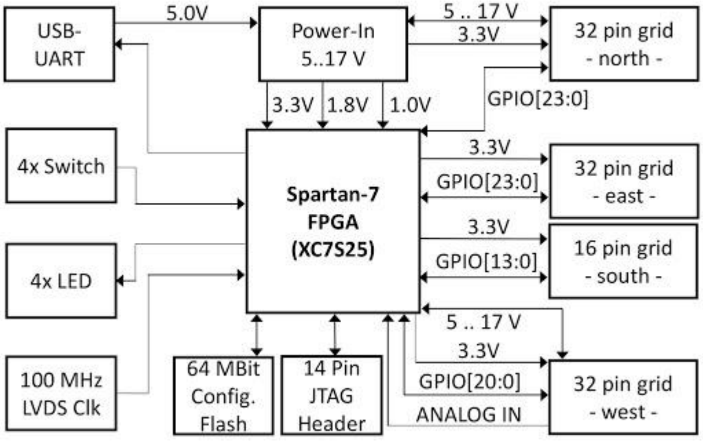

Figure 3. FPGA module circuit diagram

3.3. Data Quantification and Model Compression

The model in this paper uses a structure of two convolutional layers, a fully connected layer, and an output layer. The number of parameters used is 4.76 million parameters as mentioned above, and the fully connected layer accounts for 99% of the parameters, so it is mainly for fully connected Layer for data compression. Data quantization is performed on the parameters of all layers, from the original model 64-bit floating-point number to 16-bit fixed point number, and then using the 8-bit index, a total of 256 shared weights, and then the original weights and sharing are corrected through the back-propagation algorithm the difference in weights [9, 10]. The compression ratio formula is as follows:

$$r = \frac{nb}{r\,\boldsymbol{log}_2 k + kb} \tag{7}$$

Among them, n is the parameter quantity, b is the number of quantized bits, and k is the number of classes (256 classes) that can be represented by the bit. Equation (7) substituting the data to get the compression rate is about 4 times.

4. Experimental Results

The system design uses Xilinx's ZYNQ-7000 xc7z020clg400-1 chip as the experimental platform. The chip has 85,000 logic units, 4.9 MB Block RAM, 220 DSP48 units, and 1 GB off-chip DRAM to meet the needs of this system. The CPU platform uses Core i5 9400f, the main frequency is 2.9 GHz, the GPU platform uses GTX 1060, the GPU main frequency is 1.5 GHz, and the memory bandwidth is 160 GB/s. The experimental results are compared with the CPU, GPU platform and benchmark design. The experimental results are shown in Table 1.

Table 1. Comparison of algorithm operation of different hardware platforms.

platform	Time/ms	Power consumption/W	Problem complexity/GOPS	Performance/(GOPS/s)	Energy consumption ratio/(GOPS/s/W)
i5-9400fCPU	326	28.1	0.88400	2.71	0.096
NVIDIAGTX1060GPU	98	25	0.88400	9.02	0.361
Original design XC7Z020	49	2.61	0.88400	17.78	6.81

The operating frequency of the hardware platform in this article is 100 MHz, the time to recognize each image is 0.27 ms, the power consumption is 1.95 W, and the performance reaches 27.74 GOPS/s, which are 10.24 times and 3.08 times that of the CPU and GPU platforms, respectively, and the benchmark design 1.56 times, energy efficiency ratio is better than CPU, GPU platform and benchmark design. After the data is quantized into 16-bit fixed-point numbers, the recognition rate reaches 95% without loss of accuracy.

5. Conclusion

This paper proposes a moving image processing system based on FPGA and neural network. The overall hardware structure of the system is analyzed and the FPGA module, storage module, transmission module and power module circuit of the system are designed in detail. The FPGA-based neural network realization method is given and verified by experiments. Experimental results show that the system can effectively compress moving images. The compression effect is related to the number of hidden layer neurons. The more the number, the higher the compression signal-to-noise ratio, but the smaller the compression ratio. The number of neurons needs to be selected reasonably. Generally, 8 is more appropriate.

Acknowledgment

The authors wish to thank the wuhan institute of technology. This work was supported by a grant from the Hubei Key Laboratory of Intelligent Robot of China (Grant No. HBIRL202009).

References

[1] Cong, J., Liu, B., Neuendorffer, S., Noguera, J., Vissers, K., & Zhang, Z. High-level synthesis for FPGAs: From prototyping to deployment. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **2011**, 30(4), pp. 473-491.

[2] Sun, H., Zhao, H., Mooney, P., Zhao, H., Liu, D., & Yan, L. A novel system for moving object detection using bionic compound eyes. *Journal of Bionic Engineering* **2011**, 8(3), pp. 313-322.

[3] Kawakita, M., Iizuka, K., Aida, T., Kurita, T., & Kikuchi, H. Real-time three-dimensional video image composition by depth information. IEICE Electronics Express, 1(9) (2004), pp. 237-242.

[4] Mangione-Smith, W. H., Hutchings, B., Andrews, D., DeHon, A., Ebeling, C., Hartenstein, R., & Spaanenburg, H. A. *Seeking solutions in configurable computing. Computer* **1997**, 30(12), pp. 38-43.

[5] Meribout, M., Saied, I. M., & Al Hosani, E. A new FPGA-based terahertz imaging device for multiphase flow metering. IEEE Transactions on Terahertz Science and Technology, 8(4) (2018), pp. 418-426.

[6] Long X, Hu S, Hu Y, et al. An FPGA-Based Ultra-High-Speed Object Detection Algorithm with Multi-Frame Information Fusion. *Sensors* 2019, 19(17), pp. 3707.

[7] Zou, X.F., Li X.Y. Research and application of multifunction testing technology based on FPGA dynamic reconfiguration. *Modern Manufacturing Engineering* **2019**, 000(003), pp. 102-107.

[8] Liang Y, Li K L, Bi F H, et al. Research on LFMCW Radar Velocity Ranging Optimization System Based on FPGA – Science Direct. *Procedia Computer Science* **2020**, 166, pp. 187-194.

[9] Liu X, Zhang M, Luo Y, et al. Machine vision image acquisition hardware system based on FPGA. *Agro Food Industry Hi Tech* **2017**, 28(1), pp. 3490-3493.

[10] Xu B.; Liu L.; Wu X. A new method and simulation of image edge detection based on sobel operator and FPGA Design. *Boletin Tecnico/Technical Bulletin* **2017**, 55(11), pp. 285-292.