ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Object Detection by Implementation of Separable Convolution Gaussian Filter

Chandoji N. Bansode, Aparna Shinde

M.E Student, Dept. of E&TC Engineering, D Y Patil College of Engineering, Pune, India.

Asst. Professor, Dept. of E&TC Engineering, D Y Patil College of Engineering, Pune, India

ABSTRACT: In this paper the method uses XILINX SPARTAN 3 FPGA platform to implement architecture of separable convolution Gaussian filter with more ease. The proposed approach takes less time, less number of resources and faster processing with help of FPGA. Complexity of circuit reduces, less number of clock cycles are used and because of that power consumed is less. After Gaussian filtering it is possible for us to determine the exact shape of object with the help of edge detection algorithm. Separable Gaussian filtering is a technique which can be used for large number of tasks. It has applications of image smoothing, line parameter estimation, and texture analysis and edge detection. It converts 2D convolution into two separate 1D convolution, so reduces complexity of operation.

KEYWORDS: SPARTAN 3 FPGA, XILINX, MATLAB, Gaussian convolution Filter.

I. INTRODUCTION

Current developments of computer systems tend to reduce the size of the hardware. This is a conclusion drawn from Moore's law [1]. The hardware specifications and capabilities of a small laptop ten years ago are comparable to today's mobile devices, such as the IPhone 3GS. As a result, embedded computer systems are also becoming increasingly pervasive. For instance, today's cars include embedded systems to monitor a wide range of multi-media features such as audio, video, voice control, and navigation [5]. Another area where embedded systems play an important role is digital image processing with applications such as automated surveillance systems [3], traffic light controller systems [4]. In earlier times, those systems were mostly built with Application Specific Integrated Circuits (ASICs) which are not reprogrammable (or reconfigurable). A malfunction in one ASIC often results in a complete replacement of the faulty component. The ASICs lack of flexibility to be reprogrammed is promoting their counterpart, namely the FPGA (Field Programmable Gate Array) chips. Recently, FPGA technology has become a viable target for the implementation of algorithms in image processing applications. FPGA's generally consist of a logic block based system, which usually includes lookup tables, flip-flops and some amount of Random Access

There are various forms by which noise can be introduced into the original image, such as image acquisition stage, image transmission stage, also during image transmission stage. Our requirement is to have noise free image with highest acknowledgement of data. So to remove noise from image we use Gaussian filter. Gaussian filter is useful for various image processing works such as image blurring, image segmentation and edge detection [2]. The major advantage of Gaussian filter is that it is separable filter. It converts complex 2D convolution into two separate 1D convolution, it reduces number of resources required in the process.

Gaussian filter is used in multimedia applications like image and video analysis. The noise from video frames can also be reduced using Gaussian filter. The Gaussian convolution mask is used to multiply it with input image. The input image depends upon the weights of Gaussian mask and neighboring pixels [6]. If we want to apply large number filters then it requires more computing resources. Gaussian mask requires number of multiplications and additions, so it will take that number of multipliers and adders. To perform large and complex computations we require more computational power and faster processing time, because of that we use FPGA, by using parallel processing of FPGA the performance take less time, it provides the flexibility in the system and speed for doing calculations [3,4].

Real-time processing applications of input images required huge need of memory size and computational power. Therefore to achieve balance between area, processing speed and flexibility, I have decided to implement Gaussian filter on Xilinx SPARTAN 3 FPGA platform [3,7]. The overall working has been done with the help of Xilinx platform Studio software and Matlab. Matlab is used to convert image format into format which is suitable for FPGA processing. This work use the advantage of parallelism of FPGA and more throughput.

II. Related work

A. Convolution

Convolution is an important image processing operation which is used to filter an image by calculating the sum of products between the applied image and a smaller image like array called the "convolution kernel or convolution filter". A convolution operation has various rolls in image processing. It can achieve blurring, sharpening, noise reduction, edge detection and other useful imaging operations depending on the selection of weights of pixels in the convolution kernel [6]. A two dimensional convolution on image can be represented by the following equation:

$$h(m,n) = \sum_{i=0}^{height-1} \sum_{j=0}^{width-1} g(i,j)\, f(m-i, n-j) \qquad \text{eq. (1)}$$

Here f represent the input image and g is the convolution kernel. The width and height of the convolution kernel decides the range of double summation. A convolution operation is calculated by setting the center pixel of the convolution kernel with the pixel at the same position in the input image. Multiplying the weights of input image pixels with the pixels aligned by the convolution kernel and then by adding the results provide the value of the specific pixel in the output image [7]. A 2D convolution using a 3x3 input image and 3x3 kernels would show in fig.1.

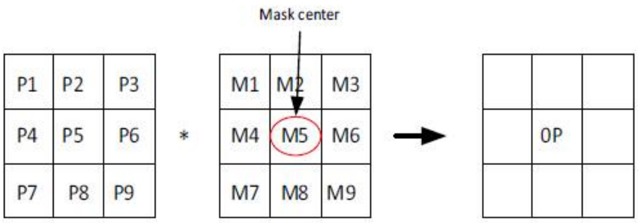

Fig.1. 2D Convolution Operation [1].

OP=P1M1+P2M2+P3M3+P4M4+P5M5+P6M6+P7M7+P8M 8+P9M9

B. Gaussian Mask

Gaussian filters are included in the class linear smoothing filters. The shape of a Gaussian function and its value at every position give the weight of pixel at that position. The Gaussian filter is used for removing noise drawn from a normal distribution [1,7]. It mostly applied in the applications where noise affects more. The equation for zero mean Gaussian function in one dimension is:

$$g(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-x^2}{2\sigma}} \quad \text{eq. (2)}$$

The width of the Gaussian is determined by Gaussian spread parameter σ. The 2D discrete Gaussian function for image processing is:

$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma}} \qquad \text{eq. (3)}$$

Here g is the Gaussian kernel, which determines weight at the location with coordinates x and y. The sharpness or smoothness of the Gaussian functions is determined by the σ parameter which is the standard deviation of the Gaussian distribution. The term 1/2πσ2 denotes normalization constant.The separable Gaussian mask is derived using equations (2) and (3) with mean equal to zero, equal to zero and normalizing factor N = 0.0016 are shown below:

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Table No.1: Horizontal Gaussian mask with mean=0, ∑=1 and N=0.0016

61	100	14

Table No.2: Vertical Gaussian mask with mean=0, ∑=1 and N=0.0016

61
100
14

C. Separable Gaussian Filter

General two-dimensional convolution between an image and a non-separable filter of N X N size takes a total of N2 multiplications and N2-1 additions for each pixel. A separable filter of N X N size can be realized as the convolution between two filters of NX1 sizes and 1XN, respectively. Thus, convolution using separable filter can be per-formed in two steps. The input image is convolved with filter of NX1 size, while the obtained result is convolved with a filter of size 1XN. In this operation, a total of 2N multiplications and 2N-2 additions required, which are less than the non-separable convolution [4]. It is particularly very useful for large scale filters.

III. PROPOSED METHOD

A. *General two Dimensional Convolution Method:*

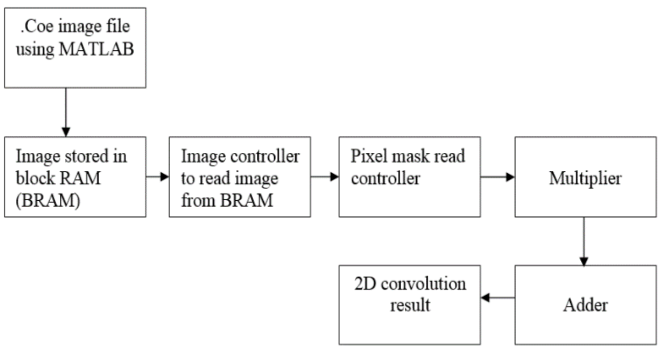

Fig.3. Two-Dimensional Convolution Method [1].

In this method, BRAM is used to store a 128×128 test image by the help of .h file which is generated with Matlab software. It is required because FPGA is not able to read jpeg or jpg image file and it also takes lots of memory space, whereas BRAM has less memory. An image controller is used to get the stored image in the BROM. The converted image and mask pixels are controlled by using pixel and mask read controller block. The multiplier is used to generate an output by multiplying input image pixel with the pixel in convolution kernel. The inputs to the multipliers are represented using n bits. In this operation n was set equal to 8. The multiplier outputs are then given as input to an adder, the output of adder is the two dimensional convolution result between the input image and the 3×3 Gaussian mask [8]. The block diagram representation of two dimensional convolutions is shown in Fig.3.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

B. Separable Convolution Method

Similar to the regular convolution operation presented in the non-separable method, BRAM is used to store a 128×128 input image using .h file and an image controller is designed to obtain the stored image in the BRAM. The obtained image pixels and mask pixels are controlled using the pixel and mask controller module. The inputs to multiplier are obtained from the pixel and mask controller module. The output generated by multiplier block is represented using 2n bits. The multiplier inputs are represented using n bits (n= 8 bits). The output of adder is the horizontal (intermediate) convolution result between the 128×128 input image and the 1×3 vertical Gaussian mask. The adder output obtained is the final result of separable convolution between the 128×128 input image and the 1×3 vertical Gaussian mask. The block diagram representation of separable convolution method is shown Fig.5. The outputs of multipliers are connected to an adder, the adder output is the separable convolution between the 128×128 input image and the separable Gaussian masks 3×1 & 1×3.

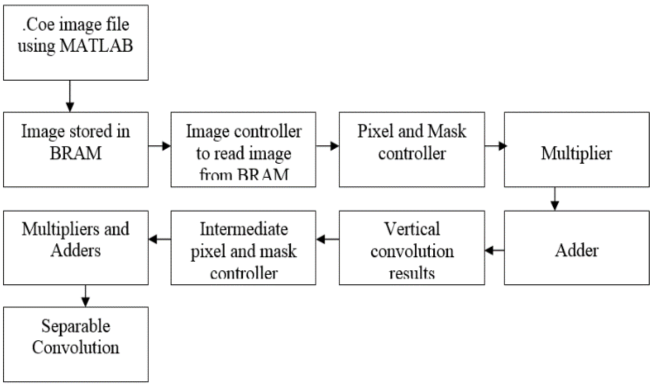

Fig.4. Block Diagram of Separable Convolution [1].

IV. SIMULATION RESULTS

The input image is processed with Xilinx SPARTAN 3 FPGA platform. The image format is first converted into the .h file using MATLAB, then it is stored on BRAM of FPGA. By using parallel processing of FPGA the image get convolved pixel by pixel and get final output as shown in Fig.7. The output image is little bit blurred, but noise get reduced.

Table No.3: Device Utilization Summary of Separable Convolution

Logic Utilization	Used	Available	Utilization
Number of slices	1880	1920	97%
Number of Flip Flops	2118	3840	55%
4 input LUTs	2971	3840	77%
Number used as logic	2418		
Used as shift registers	297		
Used as RAMs	256		
Number of IOs	62		
Bonded IOBs	62	97	63%
IOBs flip flops	64		
Number of BRAMs	4	12	33%
Number of MUX	3	12	25%
Number of GCLKs	4	8	50%
Number of DCMs	1	4	25%

The advantage of separable convolution is that number of resources used are less, so power consumption get reduced and speed of execution increases. This method is very good for complex applications and large mask sizes. The

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

separable convolution method uses less number of resources than general two dimensional convolution. It is clearly understood with the help of table no.1 given bellow.

Fig.5. Input noisy image

Fig.6. Edge detected output

Gaussian filter image can also be used in object detection, where the exact shape of object can determined and noisy part is totally eliminated. It can be used in various applications such as military purpose, navigation system and research area etc. The result of object detection has been shown below.

V. CONCLUSION AND FUTURE WORK

This work represented the implementation of separable Gaussian filter on Xilinx SPARTAN 3 FPGA platform. Output image obtained by separable convolution is similar to the image obtained by 2D convolution, but the performance obtained by separable convolution indicate that it utilizes few resources for number of clock cycles per pixel. It is clear that separable convolution approach is faster and efficient than general two dimensional convolution. Separable convolution approach is very important in complex and large computations, where large mask sizes are used. By applying edge detection on Filtered output it is easy to identify the object.

REFERENCES

[1] F. Alim, F. Talbi, S. Seddiki, I.Mezzah, B. Hachemi "Separable Convolution Gaussian Smoothing Filters on a Xilinx FPGA Platform", Fifth IEEE international conference on innovative Computing Technology, pp. 112-117, 2015.
[2] C. H Cen, P.Y Hsiaol, S.S Chou, L.T.Li, and S. Jie Chen, "A Parameterizable Digital-Approximated 2D Gaussian Filter for Edge Detection in Noisy Image,"ISCAS 2006.
[3] K. Benkrid, W. Vander "High Performance Computing on FPGAS", Springer Science Business Media, LLC 2013.
[4] M.Neggazi, M.bengherabi, A.Amira "An Efficient FPGA Implementation of Gaussian Mixture Models Based Classifier: Face Recognition Application", Systems, Signal Processing and their Applications, pp. 367-371, 2013.
[5] O. Marques, "Practical Image and Video Processing Using Matlab", John Wiley and sons 2011.
[6] D. Charalampidis, "Efficient Directional Gaussian Smoothers", IEEE Geosciences and Remote Sensing, vol. 6, pp. 383-387, 2009.
[7] B.A. Draper, J.R. Beveridge, A.P. Willem Bhom, C.Ross, M. Chawathe, J.Hammes, "Accelerated Image Processing using FPGA", Image Processing, IEEE Transction, 2008.
[8] H. Zhang, G.Hu, "A Multiwindow Partial Buffering Scheme for FPGA Based Two-Dimensional Convolvers", IEEE Transction on Circuits and Systems, vol. 54, pp. 200-204, 2007.
[9] D. Nguyen-Tuong, M. Seeger, and J. Peters, "Local Gaussian processregression for real time online model learning," Adv. in Neur. Inform. Process. Syst., pp. 1193–1200, 2009.
[10] D. J. C. MacKay, "Introduction to Gaussian processes," in Neural Networksand Machine Learning. New York: Springer, 1998, vol. 168, pp. 133–165.
[11] S. Roweis and Z. Ghahramani, "A unifying review of linear Gaussianmodels," Neur. Comput., vol. 11, no. 2, pp. 305–345, 1999.