Algorithm to Image Identification and Artix-7 FPGA Implementation for Adequate Edge Detection.

H K Raghu Vamsi K, Kasi Geethanjali, Chinthagada Naveen Kumar, P Sekhar Babu, L.V. Santosh kumar Y

Abstract: ***For any image identification based applications, edge detection is the primary step. The intention of the edge detection in image processing is to minimize the information that is not required in the analysis of identification of an image. In the process of reduction of insignificant data in the image, it may lead to some loss in information which in turn raise some problems like missing of boundaries with low contrast, false edge detection and some other noise affected problems. In order to reduce the effects due to noise, a modified version of popular edge detection algorithm "Canny edge detection algorithm" is proposed. Artix 7 FPGA board set up is used to implement, by using Xilinx platform the image that is obtained as output is displayed on monitor which is connected with FPGA board using connector port DVI. MATLAB Simulink is used for algorithm simulation and then it is executed on FPGA board using Xilinx platform. The results provide good motivation to use in different edge detection applications.***

Keywords: Canny edge detection algorithm, FPGA board model Artix 7, Xilinx platform, VGA monitors, DVI connector.

I. INTRODUCTION

The method of EDGE detection is the generalized pre-processing step in algorithms of Image processing. In current algorithms for edge detection, Canny is in top for several years due to its optimal performance. Its best performance is because of the fact that Canny performs hysteresis threshold. Threshold and low threshold should be calculated based on Statistics of the entire image. Unfortunately, this is not done in Canny edge detection algorithm. The discontinuities are sudden change in pixel intensities characterizing the boundary of different objects as well as objects in the scene. In many implementations of the Canny algorithm, Boundary is the 1st step in various system visionary algorithms.

Revised Manuscript Received on January 15, 2020

***** Correspondence Author

H K Raghu Vamsi K *, Department ECE, Raghu Engineering College, Autonomous, Visakhapatnam, India.

Kasi Geethanjali, Department Department ECE, Raghu Engineering College, Autonomous, Visakhapatnam, India.

Chintagada Naveen Kumar, Department Department ECE, Raghu Engineering College, Autonomous, Visakhapatnam, India.

P Sekhar Babu, Department Department ECE, Raghu Engineering College, Autonomous, Visakhapatnam, India.

L V Santosh Kumar Y, Department Department ECE, Raghu Engineering College, Autonomous, Visakhapatnam, India.

Therefore, these arithmetic lights determine vertical, flat, step and corner edges. The condition of the edges detected by the calculators are lighting conditions, noise, density and edge strength in the scene. As a result, reduction of localization of boundaries is attained. So arithmetic is needed to be aware of such regular changes in intensity. So there is a fake edge detection problem. The problem is due to noise without minimal contrast edges, calculation time etc. S/W and H/W Implementation is done by using Simulink, the generator (XSG) in Xilinx system, are the tools used in this paper to attain results.

This paper explains about various edge detection algorithms and its utilities in section I in brief. In section II Literature survey, the past works is given in detail. In section III, the process of System Design is discussed in detail. Then in section IV Simulated results are presented. Section V gives the conclusion of the paper.

II. LITERATURE SURVEY

Implementation on Canny standards and ASIC-based platforms, Canny-Deriche filter[1] consists of four networks. Transducer detects the edge of a 256 x 256 image in 6 seconds, much more from the requirements of real-time applications. But [2]The design improves the Canny-Deriche filter, with the set up of [1], processing is possible at 24 frames / second 33 MHz, off-chip SRAM memory used is composed of final (LIFO) stack. Canny-Deriche filter Demigny proposed a novel algorithm [3] which Reduce memory dimensions and doubles the computational economy. 2 FPGA implementations in [5] VHDL or Verilog and they use system directly soft hardware design level and hardware design tool. The parallel implementation of [6] increase the number of memories compared to access and processing time of proposed algorithm which further increase computational difficulty. All these set ups calculate the low threshold and minimal threshold offline & uses the exact one on static threshold in every image. Recently, GPGPU appeared as a strong and available parallel Computing basement for the applications on image processing. In authentic Canny method, calculation of low threshold bank on the overall stats of the picture which reduce execution time by a factor of [3][2].

The architecture is synthesized with Xilinx tool FPGA settings Artix-7, and where FPGA occupies 64% and 87% of number of slices it have in total, it takes 0.721ms of local memory (including SRAM) Edge detection (read / write time and calculation time) 512x512 images in the database are considered at 100 MHz frequency.

III.DESIGN PROCEDURE OF SYSTEM

In this work, the edge detection in image is done by using algorithm on edge detection named Canny and is first designed and simulated in simulation tool MATLAB, in that Simulink (block based design) is used. Canny is powered by FPGA board model named Artix 7. Edge is the fencing between 2 different areas in the picture that distinguishes black and white color properties relatively. With digital images, a sudden difference in pixel strength between the side by side pixels is also called as an edge.

It is an object issue and there is no other information on the picture where the mechanism is known as edge detection. Block diagram of the system explain basic ideas about the proposed hardware implementation on the FPGA platform. The basic thinking is to perform H/W execution on algorithm of canny edge in FPGA through VGA Interface to Artix7 FPGA board. In entire structure it consists of block of images, pre processing, Edge detection algorithm, FPGA board and output VGA monitor.

1) Input image: The input image is a digital image. It can be considered in different image formats saved in memory. It should be in black and white i.e., gray scale otherwise it is converted into gray scale mode for further evaluation process. MATLAB code is used to do the above mentioned process. The images that are in black and white form were considered as input in Simulink model using MATLAB Workspace block. Next to that image scaling is done, we considered 512x512 as default size to execute in our work. So whatever image is taken it is resized to the 512x512 size and then the process is continued for further extraction process.

2) Canny edge algorithm: Here this algorithm is implemented on the FPGA Artix 7 board by the following steps: As a first step pre-processing is done, then after gradient calculation will be done, after that the minimum suppression, then the dual thresholding and finally the hysteresis process is done one after another. These steps are explained briefly in the below sections.

Artix-7 FPGA board is easy to use and are used to realize this system. This system is terribly rare and requires hardware such as RS232 bus driver, JTAG connector cable, output display monitor. The download process for Artix-7 is simple to understand. This FPGA board gives particularly quick results.

A. Software design: This paper uses an image edge detection algorithm named Canny edge detection algorithm for object edge detection.

1) Image pre-processing: Basically input image of the application is a color image. For edge detection second step is to minimize this as we do not need unnecessary information. The color image is converted to a grayscale image. On this grayscale image filtering is applied because it contains noise. Median filter is recommended for this purpose.

2) Masked Convolution technique, texture gradation and guidance:

Calculation: where G(x,y) is assumed to be a 2 Dimensional masked Gaussian.

I(x,y) is the 1st derivative of image, Gauss is $g_x(x, y)$ and $g_y(x, y)$.

$$G(x,y) = \frac{1}{2\pi\sigma^2}\{e^{(\frac{x^2+y^2}{2\sigma^2})}\} \quad (1)$$

$$gx(x,y) = \frac{\partial G}{\partial x} = \frac{1}{\pi\sigma^2}x\{e^{(-\frac{x^2+y^2}{2\sigma^2})}\} \quad (2)$$

$$gy(x,y) = \frac{\partial G}{\partial y} = \frac{1}{\pi\sigma^2}y\{e^{(-\frac{x^2+y^2}{2\sigma^2})}\} \quad (3)$$

$$Ex(x,y) = gx(x,y) * I(x,y) \quad (4)$$

$$Ey(x,y) = gy(x,y) * I(x,y) \quad (5)$$

$$Grad(x,y) = \sqrt{Ex^2(x,y) + Ey^2(x,y)} \quad (6)$$

$$\theta(x,y) = \tan^{-1}\left\{\frac{Ey(x,y)}{Ex(x,y)}\right\} \quad (7)$$

3) Non-maximum suppression: Non-maximum suppression is used to reduce edge thickness for improvement in localization. If you know the direction of the gradient, pixels that are not maxima are removed. As NMS [2] compare current pixel and its 8 neighbours, along the direction of the gradient. A 3x3 window is adopted during the comparison.

4) Threshold processing and hysteresis: Output non-maximum suppression phase includes several spurs edges due to noise, these responses are called "Streaking" and can be eliminated by using hysteresis and threshold processing. The next step in the flow is to generate a bit file. Download it to the FPGA board and display the output image in VGA monitor. The entire corner identification process can be clearly visualized in figure 1.

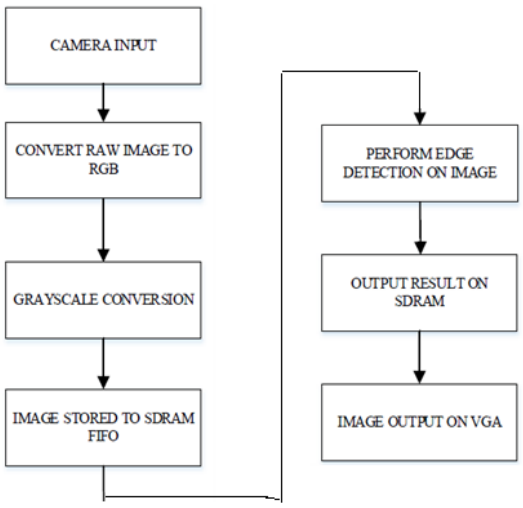

Figure 1: Flow chart of Edge detection

B. Canny Edge Detection Simulink model:

The Canny edge detection Simulink model can be visualized in Figure 3 is done using MATLAB, with the help of Simulink library. Video processing and Image processing tool boxes are used in S/W modeling. The process consists of pictures and scenes, many built-in functions. But every time we need to select an image dataset.

1) Input image considerations: This block basically access the input image of size 512x512 and color of gray scale. If there is any change in color or size, then it converts the image from block set into the standard form what we have considered and then continues to next step

2) Video Viewer: With video viewer block pixel section analysis tool is given for simulation controls like pause, forward, rewind and playback during execution.

3) Input Gateway: Xilinx libraries are used to perform this operation. In Xilinx libraries: Element, datatype, float & index. The input Gateway is taken as the input to the Xilinx tool. This block is mainly used for conversion of Simulink input to the datatype that is compatible in Xilinx.

4) Output Gateway: It has different Xilinx tool blockset library. The main intention of this block is to do the process of conversion from the fixed-point representation in Xilinx data types to datatype that is compatible in MATLAB Simulink.

5) Serialize: This subsystem purpose is conversion of data into serial format as System generator in the Xilinx tool supports only the data transfer in serial format. Figure 2 shows the various steps of block serialization. Transpose the frame, convert from 2 Dimension to 1 Dimension for the process of serialization, and at the end unbuffer the input data and transfer it serially to the Artix 7 FPGA board. Its Transpose provides data in the form of [Mx1] vectors. The vector is then converted to one dimension, next, unbuffer the data and send it serially.

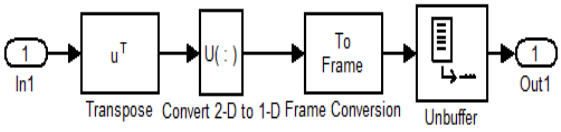

Figure 2 Serialization Simulink model

6) Registration: This block is listed in the following Xilinx block. It is D Flip Flop based register which is used to store the data. Simulink model can be visualized in figure 3 as shown below.

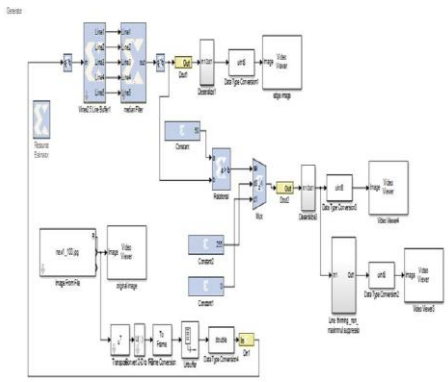

Figure 3: Canny Edge Detection Simulink model

7) Save data type: Here this section converts the input to the output scaling datatype. There are two possibilities for conversion goal. One with the real world value of inputs and the outputs which are to be same and the motto is to have a integer in stored input format and output attained values which are also to be equal.

8) Virtex2-5 line buffer: It is reference block in Xilinx Artix 7 which processes the sequential or serial stream of image pixels which creates 5-line output.

9) Median filter: Here in this blockset 9 various 2 Dimensional filters are provided for the gray scale image filtering. To select a filter, change the mask parameter 2D filter coefficient of 5x5 filter block is stored in block RAM, models are specific optimization of these remaining all coefficients.

10) Resource Estimator: It provides quickly the estimation of FPGA. In these estimations addition of these values using Lookup table (LUT), get aggregated estimation of flip-flops (FF), memory, multiplier, 3-state buffer, and Input/Output.

10) Deserialization: Since this block is given as input, the data in parallel form deserializes to Simulink block. Figure 4 shows the deserialization block. The exact opposite of the serialization process. This is the first data is buffered and then converted to 2D format, then the data is transposed.

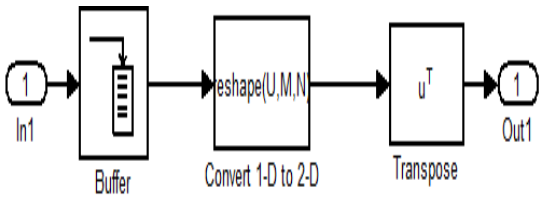

Figure 4: Deserialization Simulink model

11) Embedded block in MATLAB and System Generator token in Xilinx: Embedded block is known as S-Function block. This block is specifically used for the thinning of line & to control the suppression. And generator token is used for FPGA board Artix 7 board compatibility. After the final input is applied to this block it is compiled and the final output is executed and shown on output display monitor.

IV. SIMULATED VIEWS AND IMAGE RESULTS

The system program compiles successfully, The following RTL view diagram. [5] is acquired. The output is shown in VGA monitor which is shown in below figure. We provided real-time images and it shows a grayscale image. We got different outputs for different performance parameters such as lighting conditions and grayscale threshold variation.

Figure [5] and figure [6] the same as MATLAB Simulink is verified with FPGA DE2 board and RTL view in Xilinx. There are few edge detection images acquired by FPGA with high power usage and efficiency when compared to DSP Processor outputs. Specific hardware can be used for the edge to get detected and transfer through the occupying server in less space.

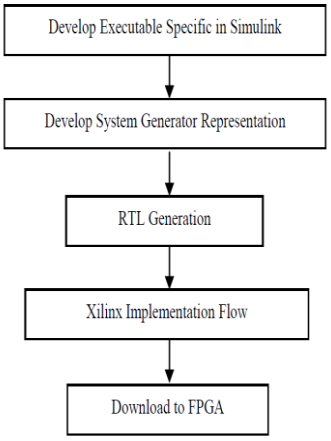

Figure 5: Experiment methodology

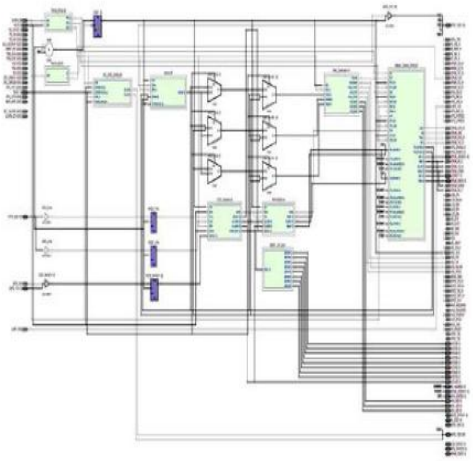

Figure 6: RTL View

Figure 7: Output Edge detection of noisy images using proposed Canny edge detector

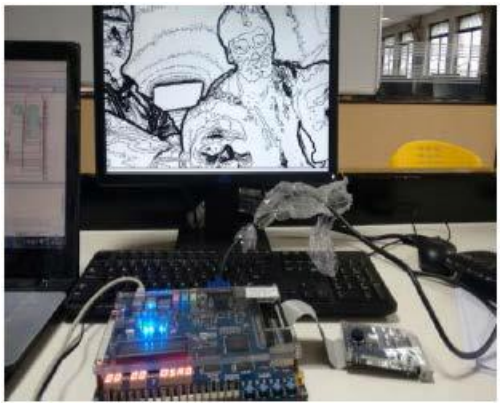

Figure 8: Edge detected image in Altera DE2 board

The Output simulated results were visualized in figure 7 and figure 8 where in the Figure 7 it shows the output Edge detection of noisy images by using the proposed Canny edge detector & in figure 8 it shows the experimental setup while conducting the experiment. In Table 1 the utilization summary of the FPGA device is shown from which the graphical comparative analysis is attained as shown in figure 9 in the view of pie chart.

Table 1: Summary report on Device Utilization

Logic used	Consumed	Vacant	Use
No. of registered slices	435	24440	2%
No. of LUT slices	185	24440	1%
Full advantage of number of LUT FFs	120	360	33%
No. of IOT bonds	12	360	3%
No. of buffer controllers	1	26	3%

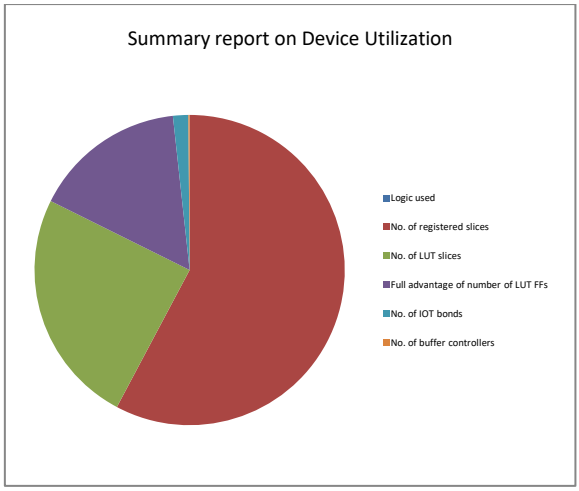

Figure 9: Device utilization summary.

V. CONCLUSION

Algorithm to Image identification and Artix-7 FPGA implementation for adequate edge detection is simulated and executed. In Hardware and software implementation, we found that the system is working properly when used in Simulink models which are basically used for the following compatibility to take place: Main step is to implement the image processing mechanism on the FPGA Artix 7 board as there are some direct functions which are related to image processing elements/ functions in the library of Xilinx blockset for Artix 7 FPGA board which is available for implementation. Artix-7 execution FPGA board is a reliable device to implement our system. Because of that the system H/W got completely minimized to complete the process of implementation and execution. Results show that it could able to detect more cutting edges but vey less missing edges. It can be utilized and is more useful in identifying or filtering many digital imaging applications.

REFERENCES

1. Raman Maini, Dr. Himanshu Aggarwal "Study and Comparison of Various Image Edge Detection Techniques" IJIP, Volume (3): Issue (1).
2. Xiaoyang Li, Jie Jiang, Qiaoyun Fan "An Improved Real-time Hardware Architecture for Canny Edge Detection Based on FPGA" Third International Conference on Intelligent Control and Information Processing July 15-17, 2012 - Dalian, China.
3. Chandrashekar N.S., Dr. K.R. Nataraj "Design and Implementation Of A Modified Canny Edge Detector Based On FPGA" International Journal of Advanced Electrical and Electronics Engineering, (IJAEEE), ISSN (Print): 2278-8948, Volume-2, Issue-1, 2013
4. Tejaswini H. R, Vidhya N, Swathi R Varma, Santhosh B "An Implementation Of Real Time Optimal Edge Detector And VLSI Architecture" International Conference on Electronics and Communication Engineering, 28th April-2013
5. B.Muralikrishna, P.Sujitha, Habibulla Khan "Real Time Edge Detection Modelling With FPGA" International Journal of VLSI and Embedded Systems-IJVES, Vol 04, Article 06103; June 2013.

AUTHORS PROFILE

Mr. H K Raghu Vamsi Kudulla, completed his M.Tech (VLSI-D) and working as Assistant Professor in Raghu Engineering College. My areas of interest include design of Low power circuits.

Ms. Kasi Geethanjali, did M.Tech(VLSI) and working as Assistant Professor in Raghu Engneering College(A). Attending several FDP's and Workshops organized by premier institutions of India and my areas of interest include design of Low power circuits.

Mr. Chinthagada Naveen Kumar, completed his M.Tech (DECS) and working as Assistant Professor in Raghu Engineering College(A). My areas of interest include signal & image processing.

Mr. P Sekhar Babu, completed his M.Tech (VLSI-D) and working as Assistant Professor in Raghu Engineering College. My areas of interest include design of Low power circuits.

Mr. Y L V Santosh Kumar, completed his M.Tech (VLSI-D) and working as Assistant Professor in Raghu Engineering College(A). My areas of interest include design of Low power circuits, Signal & Image Processing and Mixed Signal IC design.