



FPGA Based Design Implementation for Detection of Exudates Using XSG

Nazia Abdul Majeed, Satheesh Rao

M.Tech Student, Dept. of E.C.E., N.M.A.M. Institute of Technology, Nitte, India

Assistant Professor, Dept. of E.C.E., N.M.A.M. Institute of Technology, Nitte, India

ABSTRACT: A methodology for implementing DSP applications on Field programmable gate arrays employing Xilinx System Generator (XSG) is presented. This paper outlines efficient hardware architecture for detection of exudates in retinal images. The proposed design comprises architecture for Sobel and Prewitt edge detection algorithms. The edge map image obtained is enhanced for its perception using contrast stretching. Further the image is segmented to detect the exudates. This design has been implemented on Virtex-II Pro (xc2vp30-7ff896 platform). The code is synthesized within ISE 9.2 development suite. The results obtained via hardware software co-simulation use limited FPGA resources at higher maximum frequency.

KEYWORDS: XSG, FPGA, Edge detection, Segmentation, Exudates

I. INTRODUCTION

Diabetic retinopathy is a disease commonly found in diabetic patients, which is a major cause of blindness and vision defects. It causes damage to the retinal vessels thus the protein and fat gets leaked out from the abnormal blood vessels, resulting in yellowish intraregional deposits termed as Exudates. Vision loss can occur if the exudates extend into the macular area [1]. Therefore it is essential to detect these exudates. But the detection of the exudates in early stage is difficult only by visual inspection. Therefore there is lot of ongoing research to detect this disease in an early stage by using different algorithms. Various edge detection based techniques are employed by researchers to detect blood vessels and exudates.

Edge detection is a significant method in image processing, useful in the field of feature detection and extraction. Edge detection highlights sharp or gradual discontinuity in the pixel intensity. Thus amount of data to be processed is significantly reduced and irrelevant information is being filtered out. There are several types of operators available for edge detection based on first and second order derivatives.

FPGAs are a better platform for real time algorithms on application specific hardware with considerably greater performance than programmable DSPs. The requirement to process image in real time leads this implementation in hardware level. It provides parallelism and thus significantly reduces the processing time. FPGAs are highly used in medical imaging but the drawback is that high level language is used for coding. Xilinx System Generator is a model based DSP design tool from Xilinx that enables the use of Simulink design environment hence makes it easy to handle with respect to other software for FPGA design. This objective leads to the use of Xilinx System Generator (XSG).

XSG processes the image on pixel to pixel basis thus resulting in the modification of pixel neighbourhoods owing to the transformation of the whole or partial image. XSG provides an interface based on the extensive set of xilinx optimized DSP building blocks. It includes slightly higher abstraction level. It uses the Mathworks Simulink with a list of specific Xilinx blockset, which can be used to create optimized designs for Xilinx FPGA's [8]. Some blocks such as the Black box and M-code allow for direct programming in Verilog, VHDL, MATLAB M-code and C code to simplify the integration with an existing design. Additionally it provides the usage of Matlab workspace during simulation. The Xilinx System Generator has the feature of generating User constraints file, Test bench and Test vectors for testing architecture. The integrated design flow of XSG allows simulink design to create the Bit Stream (*.bit) file and generate



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

automatic VHDL or verilog code from Simulink and MATLAB. The bit file can then be downloaded on to the FPGA board.

The proposed hardware implementation method is architecturally based on the XSG tool integrated with ISE 9.2 development suite and MATLAB 2007a. The design focuses on achieving overall high performance, short development time and low cost.

II. RELATED WORK

An extensive work is carried out in the field of feature extraction for real time image processing. Edge detection being fundamental step for any image processing operation, it provokes great concern in research fraternity. Pawar presented hardware implementation of canny edge detection algorithm[2] using XSG on Virtex-5 ML506 platform and developed VGA interfacing for displaying images on screen. Z. Shanshan and W. Xiaohong, proposed simplified hardware approach for vehicle edge detection in traffic analysis [3] using XSG. Yahia et al., presented design to estimate real time sobel edge detection for video processing [4]. The design utilized black blocks in Simulink to integrate the VHDL code for co-simulation. Architecture for Prewitt edge detection algorithm [5] is developed by Pham using system generator on Xilinx Spartan 6LX platform. The FPGA implementation of point processing algorithms [6] are described by Elamaram et al., which can be extended to both spatial and frequency domain applications. An efficient MRI image filtering and Tumour Characterization algorithm [7] is implemented on FPGA using XSG by Christe et al.

The remainder of the paper is organized as follows. Section 2 briefly presents the related work. Section 3 describes the edge detection algorithms. Section 4 describes the architecture of the proposed system. Section 5 presents the hardware co-simulation setup. Section 6 briefly presents the results. Section 7 draws the conclusion.

III. EDGE DETECTION

Several types of operators are available for edge detection based on first and second order derivatives. In first order derivative input image is convolved by an adapted mask to generate a gradient image. The major classical operators like Prewitt, Sobel, and Robert are the first order derivative operators also called gradient operators which spot edges by looking for maximum and minimum intensity values. Second order derivative include Laplace operator, which is often applied after smoothening the image to reduce noise.

For edge detection, original image is convolved with coefficient of convolution kernel obtained along x and y direction. Results are then added together which yields edges in an image. This can be mathematically formulated as in eq. (1)

$$|G| = |G_x| + |G_y| \quad \text{eq. (1)}$$

A. Prewitt edge detection algorithm

In Prewitt operator, similar weights are assigned to the neighbourhood candidate pixels. Robert operator has even sized mask and the implementation of this algorithm has some drawbacks. Prewitt operator overcomes this drawback. For a given image I, horizontal and vertical gradient of each pixel (x, y) can be computed as shown in eq. (2) and eq. (3) where G_x and G_y represent horizontal and vertical gradients respectively.

$$G_x(x, y) = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix} * I(x, y) \quad \text{eq. (2)}$$

$$G_y(x, y) = \begin{bmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} * I(x, y) \quad \text{eq. (3)}$$

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

B. Sobel edge detection algorithm

Sobel is a gradient based edge detection filter. The operator consists of two kernels which are convolved with the original image to compute horizontal and vertical gradients. The two convolution kernels are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. These can then be combined together to find the absolute magnitude of the gradient at each point and the pixels close to candidate pixels are assigned higher weights. This algorithm provides better performance over Prewitt Operator. Unlike laplace operator, Sobel operator is insensitive to noise.

For a given image I, horizontal and vertical gradient of each pixel (x, y), are computed as shown in eq. (4) and eq. (5) where G_x and G_y represent horizontal and vertical gradients.

$$G_x(x, y) = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * I(x, y) \quad eq. (4)$$

$$G_y(x, y) = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * I(x, y) \quad eq. (5)$$

IV. THE PROPOSED SYSTEM

The proposed architecture for exudates detection is developed by the integrated design tool XSG. The design flow is as shown in Fig. 1 which describes about the basic idea of the proposed hardware implementation on FPGA platform. The powerful tools utilized for the design implementation are System Generator 9.2i, MATLAB 2007a, Xilinx ISE 9.2 and Virtex II-pro Development board for hardware-software co design.

The process of Xilinx System Generator is in five stages:

1. Transform the 2D input image to 1D.
2. Design the proposed model in Simulink using XSG
3. Perform the co-simulation.
4. Generate and Synthesis HDL code.
5. Hardware/software co-simulation

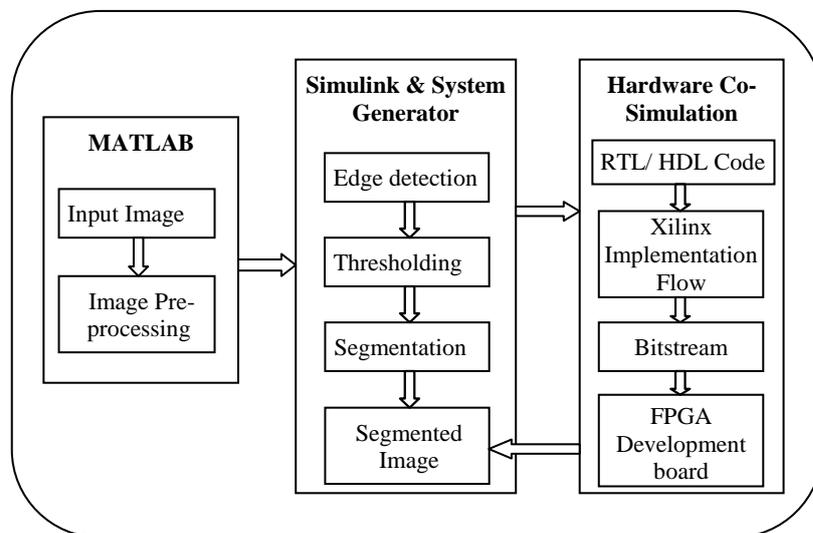


Fig. 1 Design Flow of the proposed model

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

Image Pre-processing: The images are converted from RGB to grayscale, resized and transformed to 1-dimensional vectors using MATLAB code. The pixels are serialized and temporarily stored in the workspace so as to make it in a format suitable for the hardware execution. Thus further processing of data is achieved by using ‘From Workspace’ block available in Simulink.

A. Architecture of Exudates detection using Prewitt Operator

The 2D convolution operation as shown in eq. (3) and (4) is functionally implemented using delay blocks. Horizontal gradient of an image is computed by moving horizontal kernel over an image and vertical gradient of an image is

computed by moving vertical mask over an image as shown in Fig.2

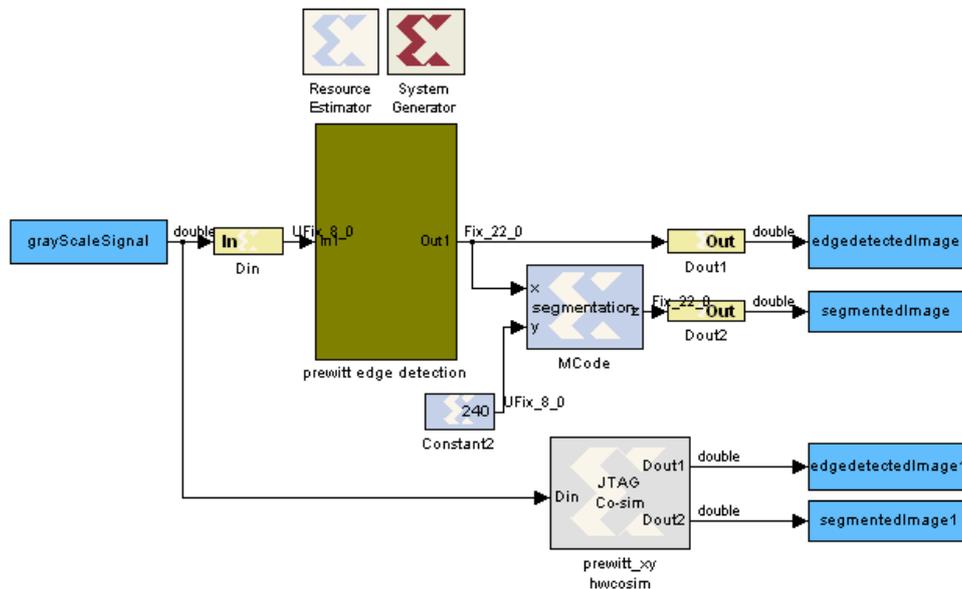


Fig. 2 Architecture of Exudates detection using Prewitt edge Detection

Segmentation is a significant method in many imaging applications. Image segmentation using threshold is essentially employed to detect the exudates which is achieved using Mcode block, on a simple pixel by pixel processing method. Constant block used in the model, sets the threshold level. The results thus obtained are reconverted to two dimensional matrix using MATLAB Workspace in order to represent it as an image. Finally, the exudates seen as yellow patches on the retina are segmented.

B. Architecture of Exudates detection using Sobel Operator

The architecture of Exudates detection using Sobel Operator is as shown in Fig.3 which consists of three stages: Edge detection, contrast stretching and segmentation.

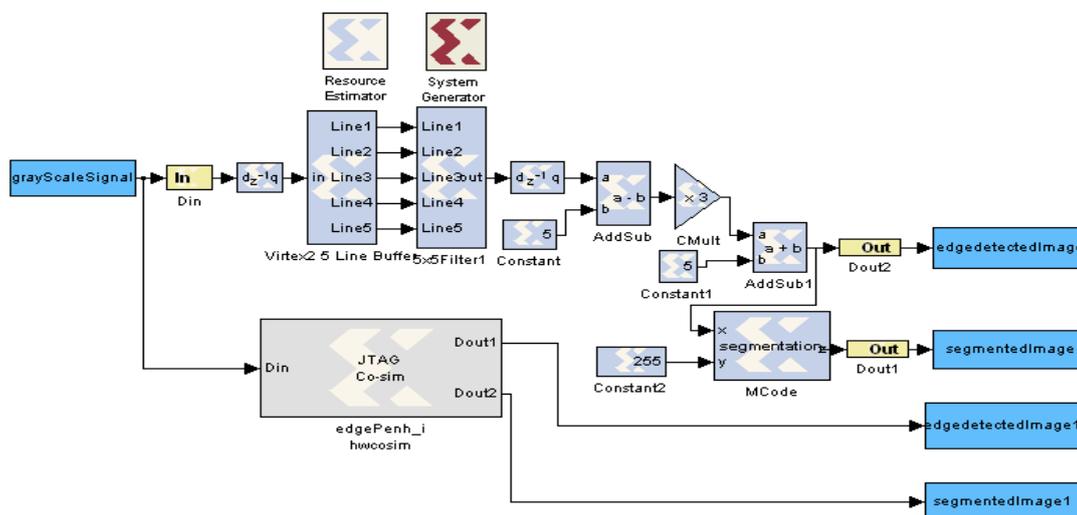


Fig. 3 Architecture of Exudates detection using Sobel Edge Detection

XSG provides a simple solution for edge detection on images [8]. The 2D convolution operation as shown in equation eq.(4) and eq.(5), can be functionally implemented as an n-tap MAC FIR filter consisting of nine programmable coefficient sets. Further high abstracted implementation can be achieved using a 5x5 filter image block.

1) *Edge detection*: The retinal pixels are sequentially streamed into virtex2.5 line buffers via a gateway in block to construct 5 lines of output. Each line is delayed by N samples where N is the length of the line. Line 1 is delayed by 4*N samples, each of the following lines are delayed by N fewer samples, and line 5 is a copy of the input.

This follows 5x5 filter which consists of parallel five n-tap MAC FIR filters and four adder blocks structure, to filter the 256x256 grayscale retinal image. 5x5 filter block provides nine different 2-D FIR filters and offers compile time parameter. The nine filters are Edge, Smooth, Sharpen, SobelX, SobelY, SobelXY, Blur, Gaussian and Identity. The 2-D filters types can be selected by varying the mask parameter on the 5x5 Filter block. SobelXY filter is selected for Sobel edge detection which performs convolution operation with input pixel values in the horizontal and vertical direction. The filter coefficients are stored in a block RAM. Hence, these coefficients can be modified by changing the mask of the 5x5 FIR filter. The architecture needs 5 clock cycles to process per pixel. The MAC processes the pixels at one pixel per clock cycle, since they are clocked five times faster than the input rate. They have to be down-sampled by 5 to match the output rate. Here the Simulink system period is set to 1/5s. The Xilinx Register block in the model is a D flip-flop-based register, having latency of one sample period.

2) *Contrast Stretching*: In the second stage the edge map image thus obtained is enhanced for its perception using contrast stretching algorithm. The contrast of an image is its distribution of light and dark pixels. It is a point process which involves application (addition, subtraction, division or multiplication) of an identical constant value to every pixel in the image. Contrast stretching in XSG is achieved by using Addsub, Constant, and CMult blocks.

3) *Image Segmentation*: The third stage depicts Image segmentation using threshold, essentially employed to detect the exudates. Segmentation using threshold decisions on a simple pixel by pixel basis can be achieved using Mcode block.

The Xilinx Resource Estimator block estimates the FPGA resources required to implement XSG model or subsystem. Additionally Xilinx System Generator token is utilized for FPGA compatibility. Using both Simulink and Xilinx blocks in the model helps for XSG simulation, HDL code generation, and synthesis. The proposed model is simulated in Simulink environment with suitable simulation time and simulation mode. The resultant segmented image is observed. Once the desired software results are obtained, system generator is configured for suitable FPGA platform.

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

V. HARDWARE CO-SIMULATION SETUP

Hardware board is installed using System Generator Board Description Builder to implement the design in Virtex-II Pro development Platform [9]. The clock is set to frequency of 100 MHz. Clocking mode is set to single step clock, which keeps the hardware in the lock step with the software simulation. To commence the co-simulation process, proper simulation time is to be specified, for the system to produce accurate results. Assuming the input image to be a square image, the system run time can be determined using eq. (6):

$$T = W^2 + W + 30 \quad \text{eq. (6)}$$

Where “W” represents pixel width of the image. The automatic code is invoked by pressing Generate button in system generator block dialog box. The code generator produces a FPGA configuration bitstream for the design that is suitable for hardware co-simulation and also comprises of additional interfacing logic that allows Sysgen to communicate with the design using a physical interface between the FPGA platform and the PC.

Completion of compiling design creates a new JTAG co-simulation block. This JTAG Co-simulation block contains all Xilinx blocks within the Gateway blocks of the original system. Gateway In and Gateway Out blocks define the FPGA boundary. During simulation, a hardware co-simulation block interacts with the underlying FPGA platform, automating tasks such as device configuration, data transfers and clocking. Co-simulation process displays the output signal produced by the FPGA hardware. The data received from the FPGA is then acquired by the MATLAB Workspace.

VI. RESULTS AND DISCUSSION

The generated HDL files are synthesized using Xilinx. The synthesis of the code gives the information about the errors, warnings, synthesis report, RTL schematic of that code, device utilization summary of the design, power consumption. The Fig.4 shows the Edge detection and Segmentation results obtained using Prewitt operator and Fig.5 shows the Edge detection and Segmentation results obtained using Sobel operator. Edge detected Image highlights the optic disc, blood vessels and exudates of the input image. Further, Segmented Image locate only the exudates from the edge detected image and thus identifying the abnormalities present in retina.

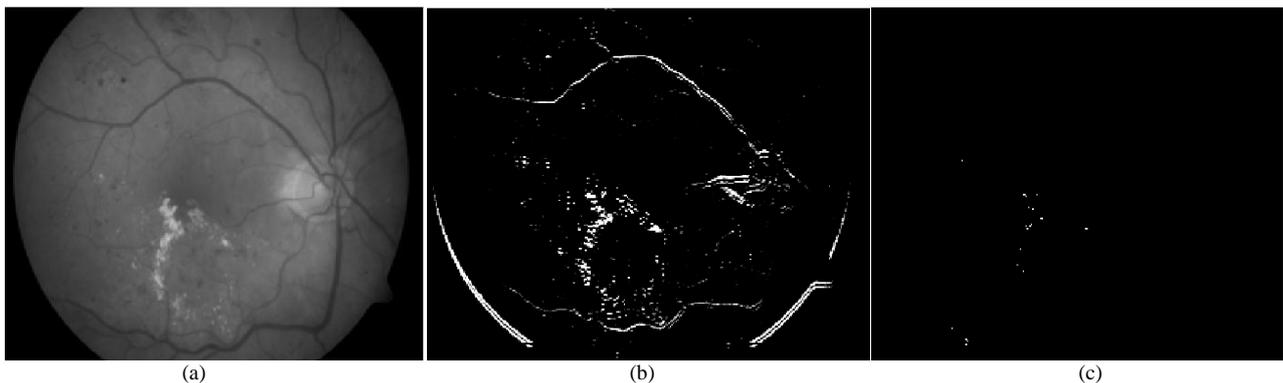


Fig.4: Prewitt Operator (a) Input Image (b) Edge Detected Image (c) Segmented Image

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

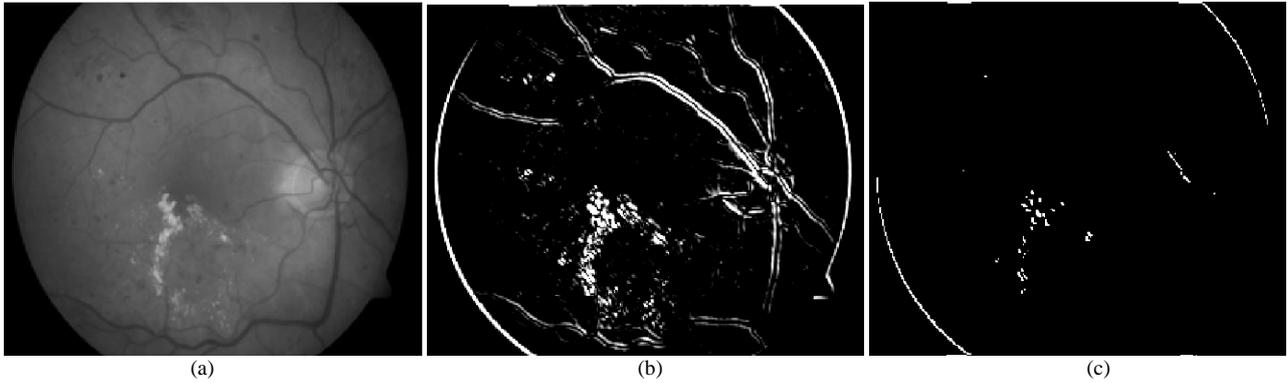


Fig.5: Sobel Operator (a) Input Image (b) Edge Detected Image (c) Segmented Image

Table 1 indicates the device utilization summary after the design is mapped to a Xilinx FPGA Virtex-II Pro xc2vp30.

TABLE 1

DEVICE UTILIZATION SUMMARY

Logic Utilization	Sobel Operator		Prewitt Operator	
	Used	Utilization (%)	Used	Utilization (%)
Number of Slices	350/13696	2%	142/13696	1%
Number of Slice Flip Flops	579/27392	2%	155/27392	0%
Number of 4 input LUTs	390/27392	1%	172/27392	0%
Number of bonded IOBs	47/556	8%	53/556	9%
Number of BRAMs	9/136	6%	0/136	0%
Number of MULT18X18s	6/136	4%	0/136	0%
Number of GCLKs	1/16	6%	1/16	6%

Device utilization summary depicts the information pertaining to device utilization analysis. Detailed reports provide access to reports that are generated as the design is processed. This shows the comparative analysis of resource utilization for the architectures using Sobel and Prewitt operator. Table 2 depicts the comparative analysis of total power dissipation and maximum operating frequency for the two designs which employed sobel and prewitt operator. It is observed that the results obtained via hardware software co-simulation use limited FPGA resources at higher maximum frequency and low power consumption.

TABLE 2

COMPARISON OF POWER DISSIPATION AND MAXIMUM FREQUENCY

Operator	Max Frequency	Total Power(W)
Sobel	323.23Mhz	0.210 W
Prewitt	100.29MHz	0.123W



International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

VII. CONCLUSION

Xilinx system generator has a unique FPGA in the loop co-simulation feature, which greatly simplifies the complicated programming and allows the designers to accelerate simulation while simultaneously testing the design in hardware. The aim of this paper is to prove the role of System Generator in designing a hardware system for the recognition of Retinal exudates and thus identify the abnormalities present in retina. The hardware architecture of edge detection using Sobel and Prewitt Operator followed by segmentation method is successfully implemented in the device Virtex-II Pro (xc2vp30-7ff896 platform). For real time applications, Sobel algorithm produces better results than Prewitt algorithm. The results obtained via hardware software co-simulation use limited FPGA resources at higher maximum frequency and low power consumption.

REFERENCES

1. Singer DE, Nathan DM, Fogel HA, Schachat AP., "Screening for diabetic retinopathy", Ann Intern Med, pp.660-71, 1992.
2. P.H. Pawar and R.P. Patil, "FPGA implementation of Canny edge detection algorithm," International Journal of Engineering and Computer Science, vol. 3, pp. 8704-8709, 2014.
3. Z. Shanshan and W. Xiaohong, "Vehicle Image Edge Detection Algorithm Hardware Implementation on FPGA," IEEE International Conference on Automation and Logistics, Vol.14, pp.184-188, 2010.
4. Yahia Said, Taoufik Saidani, Fethi Smach and Mohamed Atri, "Real Time Hardware Co-simulation of Edge Detection for Video Processing Time", Electrotechnical Conference (MELECON), 16th IEEE Mediterranean, Vol.25, pp.852-855, 2012.
5. P. Nguyen, J. Cho, and S. B. Cho, "An architecture of real time hardware co-simulation of edge detection in image processing using prewitt edge operator," Proc. of ICEIC, pp.1-2, 2014.
6. V. Elamaran and G. Rajkumar, "FPGA implementation of point processes using Xilinx system generator", Journal of Theoretical and Applied Information Technology, Vol. 41, pp.201-206, 2012.
7. Shilpa S.A. Christe, M. Vignesh and A. Kandaswamy, "An efficient FPGA implementation of MRI image filtering and tumour Characterization using Xilinx system generator" International Journal of VLSI design and Communication Systems, Vol. 2 Issue No. 4, pp.95-109, 2011.
8. System Generator for DSP user guides, [Online]. Available: http://www.xilinx.com/support/sw_manuals/sysgen_bklist.pdf, 2010.
9. Xilinx, Inc., XUP Virtex-II Pro Development System Hardware Reference Manual, 2005.