

# Implementation of SOBEL Edge Detector on FPGA Using System Generator

Bhupendra Fataniya<sup>1</sup>, Pratik Trivedi<sup>2</sup>

<sup>1</sup> Assistant Professor, Nirma University, Institute of Technology, Ahmedabad, Gujarat, India (Phone Number: 9725064335; e-mail: bdfataniya@nirmauni.ac.in).

<sup>2</sup> Assistant Professor, Nirma University, Institute of Technology, Ahmedabad, Gujarat, India (Phone Number: 9913876184; e-mail: pratik.trivedi@nirmauni.ac.in).

## ABSTRACT

Edge detection serves as a pre-processing step for many image processing algorithms such as image enhancement, image segmentation, tracking and image/video coding and one of the key stages in image processing and object recognition. This edge detection algorithm is based on MATLAB simulation and FPGA implementation. With the introduction of reconfigurable platform such as Field Programmable Gate Arrays (FPGA) and advent of new high level tools to configure them, image processing on FPGA has emerged as a practical solutions for most of computer vision and image processing problems. The design utilizes powerful design tool System Generator (Sys Gen) and Xilinx Development Kit for hardware-software code sign. Currently the image processing algorithms has been limited to software implementation which is slower due to the limited processor speed. So a dedicated processor for edge detection and segmentation is required which was not possible until advancement in VLSI technology. Now more complex system can be integrated on a single chip providing a platform to process real time algorithms on hardware. To meet the speed and area constraints, it is important to quantify the reduction in processing speed as well as FPGA resources that can be achieved if a component of the image/video processing system is embedded onto a hardware based platform like an FPGA. A flexible field programmable gate array device lets develop the image processing application so that the same logic substrate is reconfigured and reused by several custom coprocessors during the different operation stages of the sequential biometric algorithm. The results obtained with this technology reveal that a reconfigurable FPGA faces both real-time and parallel compute-intensive demands of the image enhancement process. This paper outlines, an efficient FPGA based architecture for Edge Detection using Sobel operator and uses Thresholding method for Segmentation. Sobel edge detection algorithm results in significantly reduced memory requirements decreased latency and increased throughput with no loss in edge detection performance and its property of less deterioration in high levels of noise.

**Keywords:** *FPGA, Sobel Edge Detection, Thresholding*

## Introduction

Edge detection is the name for a set of mathematical methods which aim at identifying points in a digital image at which the image brightness changes sharply or more formally, has discontinuities. The points at which image brightness changes sharply are typically organized into a set of curved line segments termed *edges*. [1] Edge detection is a fundamental tool in image processing, machine vision and computer vision, particularly in the areas of feature detection and feature extraction.

The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. It can be shown that under rather general assumptions for an image formation model, discontinuities in image brightness are likely to correspond to: discontinuities in depth, discontinuities in surface orientation, changes in material properties and variations in scene illumination. In the ideal case, the result of applying an edge detector to an image may lead to a set of connected curves that indicate the boundaries of objects, the boundaries of surface markings as well as curves that correspond to discontinuities in surface orientation. Thus, applying an edge detection algorithm to an image may significantly reduce the amount of data to be processed and may therefore filter out information that may be regarded as less relevant, while preserving (IJRDE) the important structural properties of an image. If the edge detection step is successful, the subsequent task of

---

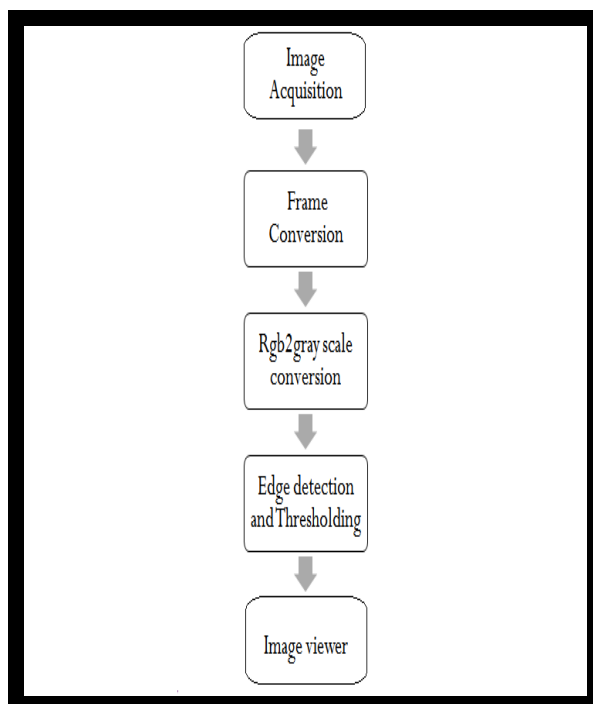
interpreting the information contents in the original image may therefore be substantially simplified. However, it is not always possible to obtain such ideal edges from real life images of moderate complexity.

### Proposed Algorithm and its implementation

Our edge detection algorithm is based on Sobel Operator. For getting result using edge detection block some pre-processing and post-processing are required on the image while using Xilinx block set.

The proposed algorithm consists of four major parts:

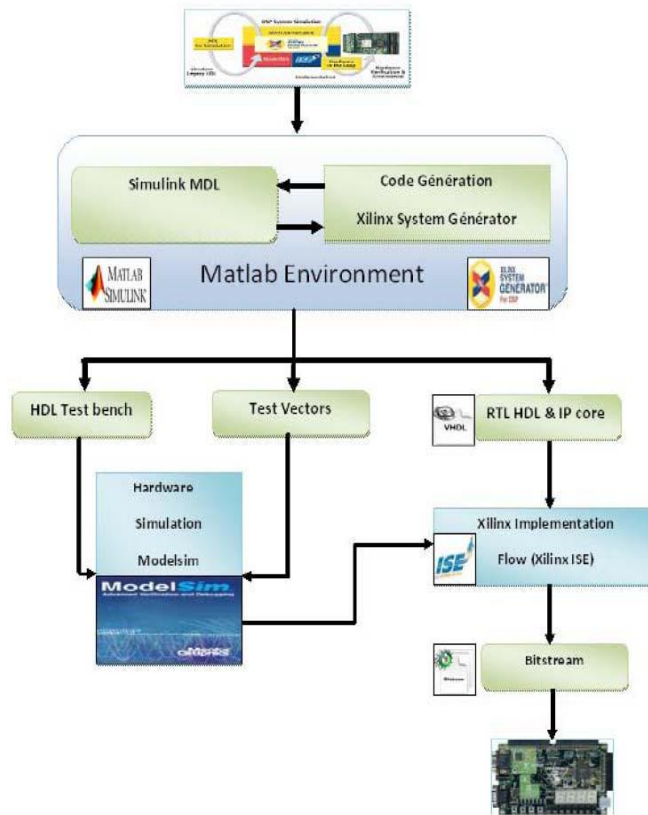
Frame Conversion, Rgb2gray scale Conversion, Sobel Edge Detection, Thresholding.



**Figure1 Flow chart of Edge detection**

Sobel and Canny are two very popular edge detection algorithms, and though they provide similar functionality, the mathematics and methods behind the two algorithms have several differences. Sobel is a gradient based edge detection filter. It works by scanning the image pixel by pixel calculating the approximate gradient of the image intensity at each point. The algorithm then compares the pixel to the pixels around it to judge how the image changes from point to point and how likely there is an edge at that point. Sobel edge detection operates by scanning the input image with a 3x3 convolution matrix, from which the output indicates an edge in either the x or y direction, depending on the matrix. The main advantage of Sobel is that it is computationally fast because of its horizontal and vertical convolutions as it scans the image with a 3x3 filter.

As shown in the figure 2 we read the image into Matlab workspace. Then after that image is converted into 256x256 image for ease of processing. After converting that image the resultant image is given to frame conversion system in the model where 2-D image is converted into 1-D image for separate color processing. The separate color signal is given to gray scale conversion block where image is converted into image. The output of that block is given to sobel edge detection system where Horizontal and vertical masking are done on the image and the output is given to thresholding system. The output of that block is given to 1-D to 2-D conversion System. That output is displayed on the screen using Image viewer block of Xilinx blocks set.



**Figure 2 Design flow for Xilinx system generator**

**Table 1: Comparison between Sobel and Canny detection**

Engineering Criteria	Importance	Sobel	Canny
Detect Edges	5	7	9
Minimal Cost	3		
Minimal False Positives and Positive Negatives	4	6	7
Operate with noise present	4	6	7
Operate in real-time	5	9	7
Development Time	4	9	3
Easily upgradable	4	8	5
Additional Features	2	2	3
Totals		200	174

## Conclusion

In this paper, building of the image edge detection algorithm, using System Generator is achieved on the Virtex5 development board, with satisfactory results. The proposed method is a robust approach to detect edges in the image. It can be applied to different types of images, stored or directly acquired by cameras, and can be considered a powerful initial step for many computer vision applications. System Generator does not replace the way to achieve HDL but simplifies the complicated programming in Xilinx IDE. There are complex algorithms for further analysis in the future. At the same time, we need to build a more complex image algorithm model, and further applied to video processing, multimedia processing and other areas.

## REFERENCES

1. S. Varadarajan, C. Chakrabarti, L. J. Karam, and J. M. Bauza, "A Distributed psycho-visually motivated canny edge detector," *IEEE ICASSP*, pp. 822-825, Mar. 2010.
2. L. Torres, M. Robert, E. Bourennane, and M. Paindavoine, "Implementation of a recursive real time edge detector using Retiming techniques," *VLSI*, pp. 811-816, Aug. 1995.
3. Qian Xu, Chaitali Chakrabarti and Lina J. Karam, "A Distributed Canny Edge Detector and Its Implementation On FPGA," Tempe, AZ.
4. D. V. Rao and M. Venkatesan, "An efficient reconfigurable Architecture and implementation of edge detection algorithm Using Handle-C," *ITCC*, vol. 2, pp. 843-847, Apr. 2004.
5. Shengxiao Niu, Jingjing Yang, Sheng Wang, Gengsheng Chen, "Improvement and Parallel Implementation of Canny Edge Detection Algorithm Based on GPU,"
6. W. He and K. Yuan, "An improved Canny edge detector and its Realization on FPGA," *WCICA*, pp. 6561-6564, Jun. 2008.
7. J. Canny, "A computational approach to edge detection," *IEEE Trans. PAMI*, vol. 8, no. 6, pp. 679-698, Nov. 1986.
8. Sudeep K. Cand Dr. Jharna Majumdar "A Novel Architecture for Real Time Implementation of Edge Detectors on FPGA," *IJCSI International Journal of Computer Science Issues*, Vol. 8, Issue 1, January 2011 193 ISSN (Online): 1694-0814
9. Ravi.S Abdul Rahim, Fahimuddin Shaik "FPGA Based Design and Implementation of Image Edge Detection Using Xilinx System Generator," *IJETT* Volume 4 Issue 10 - Oct 2013
10. Jinu Prabhakar. K. P, Anitha Mary "FPGA Based Lane Deviation System Using System Generator," *IJARCSSE - Volume 3, Issue 2, February 2013*
11. Rajesh Mehra, Rupinder Verma "Area Efficient FPGA Implementation of Sobel Edge Detector for Image Processing Applications," *IJCA - Volume 56 No.16, October 2012*
12. Veni. S. K. A. Narayanankutty Mohammad Raffi "Hardware Implementation of Edge detection on Hexagonal Sampled Image Grids," *IJCA - Volume 24 No.2, June 2011*
13. Mandeep Kaur, Kulbir Singh "Implementation and performance analysis of Video Edge Detection system on Multiprocessor Platform," *IJACSA - Vol. 2, No.5, 2011*
14. N. Prathyusha and Prof. A. Balaji Nehru "A High Speed ASIC Design for Sobel Edge Detection using FPGA," *IJNTEC - Vol.1, Issue. 4, Nov. 2013*
15. Dr. Abdulsattar M. Khidhir, Nawal Younis Abdullah "FPGA Based Edge Detection Using Modified Sobel Filter," *IJRDE - Vol.2: Issue.1, June-July 2013*