Scientific Journal of Impact Factor(SJIF): 3.134

e-ISSN(O): 2348-4470
p-ISSN(P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 2,Issue 6, June -2015

Edge Detection Using FPGA

Jay Katira [#1], Prof. Mayank Mahant[#2], Prof. Kavindra Jain[#3], Prof. Chintan Patel [$4]

[#] *Department of Electronics & Communication, G. H. Patel College of Engineering & Technology (GCET),Vallabh VidyaNagar, Gujarat, India.*

[$] *Department of Electronics, BVM Engineering College, Vallabh Vidyanagar, Gujarat, India.*

Abstract- *In today's world of high technology, there is a greater need of conversion i.e. to convert the analog into digital. Since commence of digital scanners in the computer world, there has been a need to convert books/text into digital form which is viewable over the internet and/or on a computer. This is where the optical character recognition comes in picture. The goal of Optical Character Recognition (OCR) is to classify optical patterns (often contained in a digital image) corresponding to alphanumeric or other characters. The process of OCR involves several steps including segmentation, feature extraction, and classification. Character recognition with FPGA makes a system that can work more efficient and also in suitable cost. FPGA is a special circuit so the implementation of recognition blocks can be easy. Use of FPGA will help us to facilitate the needed usage of recognition. By this scenario we can make system for particular applications like reader for blind person, OCR for Forensic labs, Historical damaged books or monomials can be extracted. The main purpose of this Dissertation is to make a system which is efficient, cheap, easy to use and less complex.*

Keywords *-Edge detection, Sobel Edge Detection, MATLAB, FPGA, Model Sim, VHDL.*

I. INTRODUCTION

As in current time, we are surrounding with high technology. In this environment we have higher demand of conversion i.e. to convert the analog into digital. Since the digital scanners are invented in this computerized world, we need to convert books/ text into digitized form. This form can be seen over internet or computer. This is where the optical character recognition comes in picture. Optical Character Recognition (OCR) is the mechanical or electronic translation of images of handwritten, typewritten or printed text (usually captured by a scanner) into a machine-editable text. It is often used to convert paper books and documents into electronic files. When one scans a paper page into a computer, it produces just an image file, a photo of the page. The computer cannot understand the letters on the page, so you cannot search forwords or edit it and have the words re-wrap as you type, or change the font, as in a word processor. You would use OCR Software to convert it into a text or word processor file so that you could do those things. The result is much more flexible and compact than the original page photo. OCR can be little complicated. There may be some affection like Noise, Distortion, Skewing, etc. to fail or to get wrong output from the OCR algorithm for recognition. This is why OCR is generally statistical in nature. This means that different OCR algorithms are available for different scenario. Generally no OCR algorithm is fully suited to every possible scenario. As we have, for type-written books, a simpler OCR algorithm may suite for best performance. In other hand, the recognition process of digit we can use a much more complex OCR algorithm. This complex algorithm would be required to recognize digits which contain noise or distortion. For some forensic labs which analyse handwriting on, perhaps, broken scraps of paper, a much more complex OCR algorithm would be required.The goal of Optical Character Recognition (OCR) is to classify optical patterns (often contained in a digital image) corresponding to alphanumeric or other characters. The process of OCR involves several steps including segmentation, feature extraction, and classification. Each of these steps is a field unto itself, and is described briefly here in the context of a MATLAB implementation of OCROptical Character Recognition (OCR) is a type of document image analysis where a scanned digital image that contains either machine printed or handwritten script is input into an OCR software engine and translating it into an editable machine readable digital text format (like ASCII text).OCR works by first pre-processing the digital page image into its smallest component parts with layout analysis to find text blocks, sentence/line blocks, word blocks and character blocks. Other features such as lines, graphics, photographs etc. are recognized and discarded. The character blocks are then further broken down into components parts, pattern recognized and compared to the OCR engines large dictionary of characters from various fonts and languages. Once a likely match is made then this is recorded and a set of characters in the word blocks are recognized until all likely characters have been found for the word block.

An **FPGA** is a device that contains a matrix of reconfigurable gate array logic circuitry. When a FPGA is configured, the internal circuitry is connected in a way that creates a hardware implementation of the software application. Unlike processors, FPGAs use dedicated hardware for processing logic and do not have an operating system. FPGAs are truly parallel in nature so different processing operations do not have to compete for the same resources. As a result, the performance of one part of the application is not affected when additional processing is added. Also, multiple control loops can run on a single FPGA device at different rates. FPGA-based control systems can enforce critical

interlock logic and can be designed to prevent I/O forcing by an operator. However, unlike hard-wired printed circuit board (PCB) designs which have fixed hardware resources, FPGA-based systems can literally rewire their internal circuitry to allow reconfiguration after the control system is deployed to the field. FPGA devices deliver the performance and reliability of dedicated hardware circuitry.

II. LITERATURE SURVEY

For the Character Recognition process, some different algorithms from Image Processing Part has been studied. From that we are performing Edge Detection process. Some limitations are found from the Literature. To overcome the limitations the Edge Detection algorithm is present. Here are some reviews which are extracted from Literature.

α-soft is the technic that uses database in place of ANN (Artificial Neural Network) because OCR should not be expansive and complex[1].Mainly they are working on ENGLISH language. It is an international language. It is spoken and used in all over world. They are implementing this α-soft for recognizing Business cards and documents.The Recursive Text Alignment Scheme (RETAS) technic is used for accuracy of OCR for recognition of books written in different languages[2]. In this they have used ground truth for evolution. We can built the ground truth manually. They have used two ways. One way leads to create ground truth is to type the data then print it and scan it. Scanned data will be given to OCR. Result will be compared with known ground truth. Another way is to create synthetic data and add the noise with the help of degradation models.

OCRopus is developed in IUPR Research Group and it is hosted by Google Code[3]. It consists main components.They have study a system and implemented a system that can recognize documents and mathematical formulas in it. It is made of formula structure analysis and character recognition steps. There is a need of improvement in processing for handling more number of documents. That's why they have implemented mathematical OCR with the help of OCRopus system.Some implementations even use State Vector Machines (SVM) along with wavelets as the input variables for the OCR process [4]. The above mentioned methods use extensive mathematical operations and involve several calculations to deduce the wavelets. These in turn use matrix multiplications, summations and the SVM essentially requires time to take different input-output pairs and calculate the relation between them. Approaches using artificial neural networks also use training mechanism for OCR. Image Pre-processing, there are also systems which do not need training and memory based training or recognition.

By reviewing these papers, OCR reveals that computational time is more. The decryption process takes a longer time for prediction. Until now FPGA based work along with OCR is not done. Line, width and height are to be predefined which remain a constraint with the recognition process. Various ANN models and required weights are used but use of FPGA would decrease both computational time as well as decryption would be fast. System would become much cheaper, easy to use and quite efficient.

III.SYSTEM DESIGN

In this paper Image Edge Detection Algorithm is implemented for detect the edges in Image. Character Recognition is a process of extracting the character from typed, scanned, fuzzy image. For this different steps are being performed so that correct character is recognized. Edge Detection process is one of the part of this process. Here, an algorithm for Edge Detection of a character or any image is implemented.

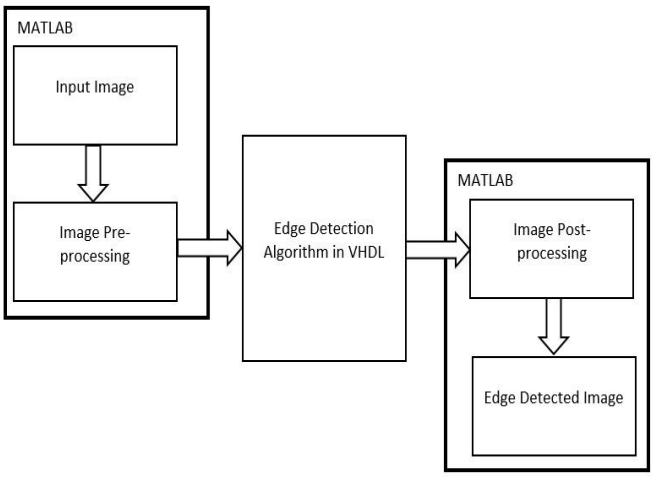


Figure 1: Block Diagram of Image Edge Detection

Block diagram shows the whole implementation. Which includes Input Image, Image Pre-processing, Edge Detection algorithm in VHDL, Image post-processing and Edge Detected image. Blocks implemented in MATLAB are: Input Image, Image Pre-processing, Image Post-processing and Edge Detected Image. Blocks Implemented in VHDL are: Edge Detection algorithm in VHDL. VHDL block is tested in Modelsim via Testbench.Description of each block in the diagram is as follow.

Input Image: This block includes that we can take any image. As we have characters in coloured form or black and white form, any person's Image, any object Image, etc. this will take the any kind of image. We have one condition that we have to take 640x480 resolution image.

Image Pre-processing: This block includes Pre-processing steps which are Read the image, convert to gray scale, transpose,
Convert 2D to 1D and write to File. This block will read the Image then convert it to Gray scale form. It will transpose the Image then convert this 2D image to 1D form. This 1D form data will be loaded to a file which is like a text file.

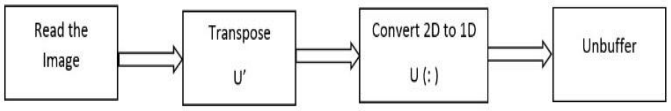


Figure 2: Image Pre-processing

Image Post-processing: This block is totally reverse process of Pre-processing. This will get the edge detected file or image. It includes read the file, convert 1D to 2D, Convert to text file, Convert to Image. This process will read the file then convert 1D data to 2D. These data will be written to a text file and then it will generate the matrix which will form the Image.

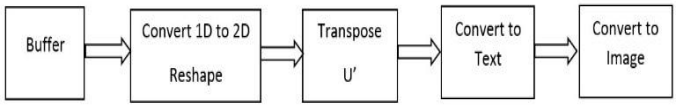


Figure 3: Image Post-processing

Edge Detected Image: This will show the Edge Detected image. Which is easily identify the original Image.

Edge Detection algorithm in VHDL: This block is having the VHDL code. Which can extract the edges of an Image. We have tested the whole code in ModelSim software via Testbench. This includes the Sobel Edge Detection algorithm. Which is correctly performing Edge Detection of any kind of Image.

Figure shows the Hierarchical flow of Edge Detection algorithm. It includes Top level entity which is having edge_sobel entity. Edge_sobel entity is having three cache memory, Horizontal filter, Vertical filter, Double FIFO Line buffer, FIFO Line Buffer and Convolution with mask and Gradient.

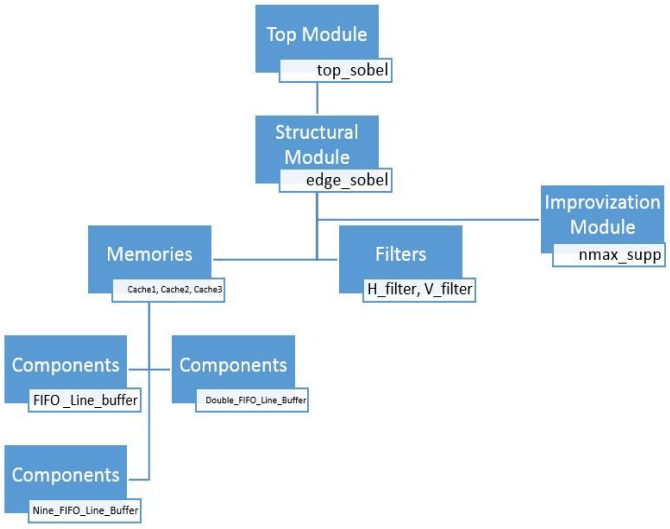


Figure 4: Hierarchy Model

Sobel algorithm:

Most edge detection methods work on the assumption that the edge occurs where there is a discontinuity in the intensity function or a very steep intensity gradient in the image. Using this assumption, if one take the derivative of the intensity value across the image and find points where the derivative is maximum, then the edge could be located. The gradient is a vector, whose components measure how rapid pixel value are changing with distance in the x and y direction. Thus, the components of the gradient may be found using the following approximation

$$\frac{\partial f(x,y)}{\partial x} = \Delta x = \frac{f(x+\Delta x,y)-f(x,y)}{\partial x};$$

$$\frac{\partial f(x,y)}{\partial x} = \Delta y = \frac{f(x,y+\Delta y)-f(x,y)}{\partial y};$$

Where ∂x and ∂y measure distance along the x and y directions respectively. In discrete images, one can consider ∂x and ∂y in terms of numbers of pixel between two points. ∂x = ∂y =1 (pixel spacing) is the point at which pixel coordinates are (i, j), thus,

$$\Delta x = f(i+1, j) - f(i, j);$$
$$\Delta y = f(i, j+1) - f(i, j);$$

In order to detect the presence of a gradient discontinuity, one could calculate the change in the gradient at (i, j) , .This can be done by finding the following magnitude measure

$$M = \Delta x^2 + \Delta^2 y;$$

and the gradient direction θ is given by

$$\theta = \tan^{-1}\frac{\Delta y}{\Delta x};$$

IV. SIMULATION RESULTS

Original Image is given to the MATLA to obtained edge of the image to the simulation level. In the Figure given below shows the original image. In ModelSim input file is taken as image.Sobel Edge Detection is applied to get edge of image. The edge detection and another out file generation is shown in fig which are in the form of waveform. This file is read and convert to the Image. Which is shown in the figure.

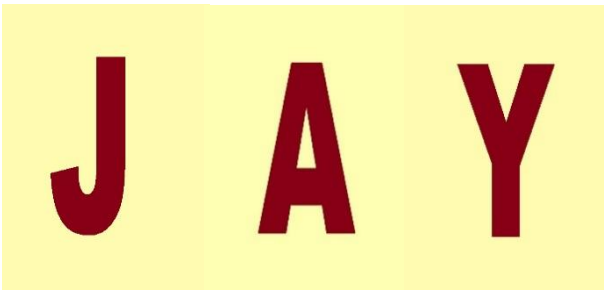
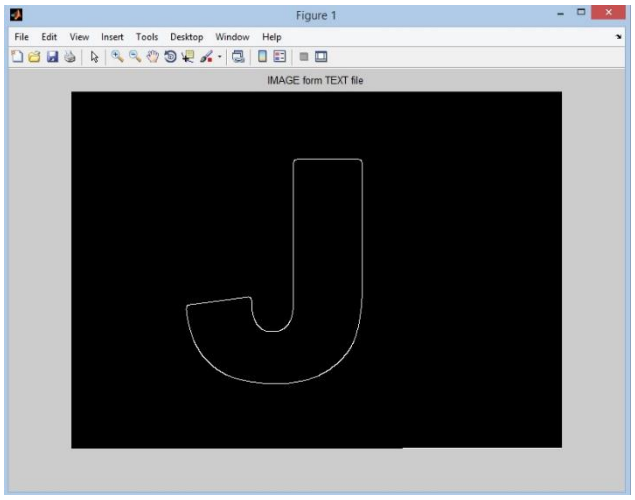
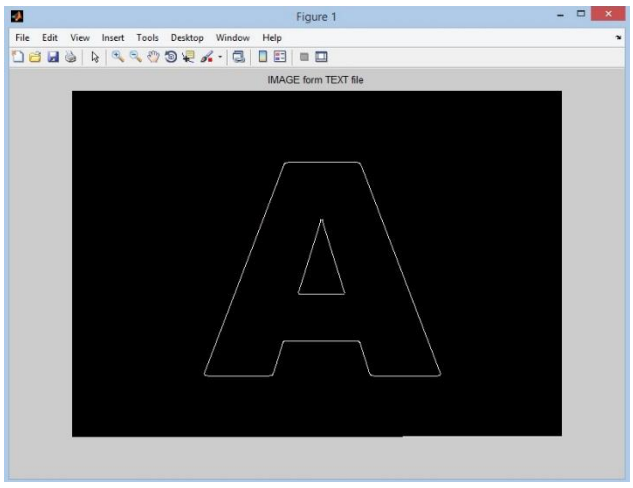


Figure 5:Original Images





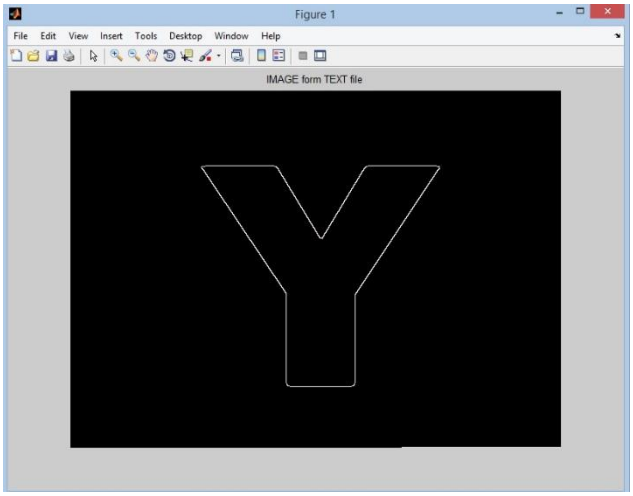


Figure 6: MATLAB Simulation

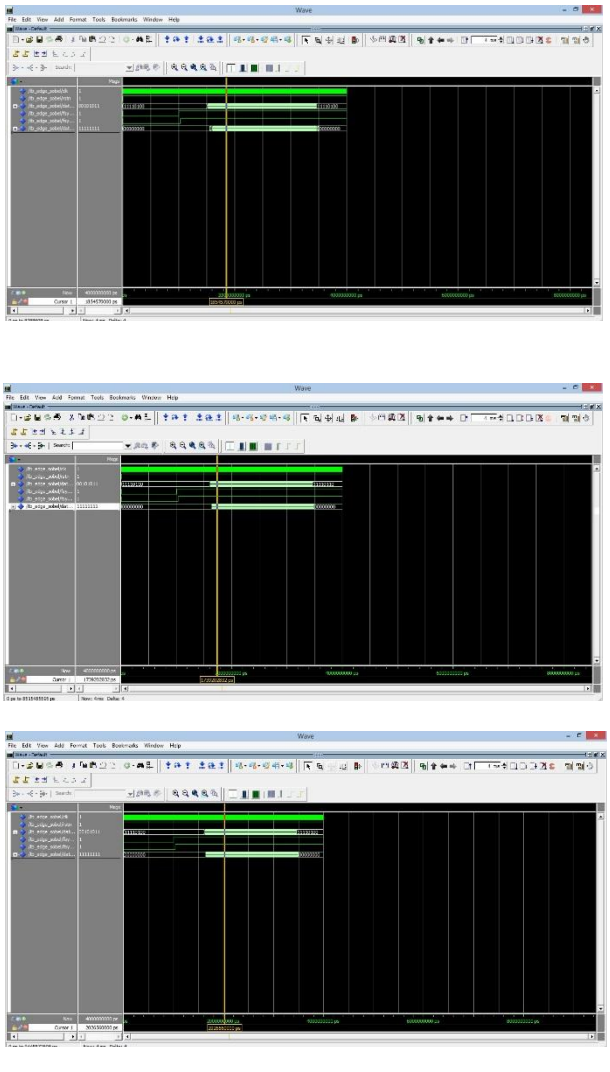


Figure 7: Model Sim Simulation

V.CONCLUSION

The implementation of Sobel Edge Detection Algorithm on FPGA is described, and it is done successfully. The software implementation of the system is working properly. The MATLAB models are used for the compatibility of implementing the image processing system on FPGA. Because it has direct functions for the image processing. Resources used are also very less for implementation. The result of Sobel Edge Detection algorithm detects more edges whit less missing edges. It is not noise sensitive. It is useful in those digital image applications where only shape and size of image is required. The Hardware implementation is remaining. The proper connection coding is required for that. We are going to implement the Hardware part in future.

VI. REFERENCES

[1]. Junaid Tariq, Umar Nauman, Muhammad Umair Naru, "α-Soft: An English Language OCR" , 2010 Second International Conference on Computer Engineering and Applications, 978-0-7695-3982-9/10 2010 IEEE, DOI 10.1109/ICCEA.2010.112
[2]. Ismet Zeki Yalniz, R. Manmatha, "A Fast Alignment Scheme for Automatic OCR Evaluation of Books" , 2011 International Conference on Document Analysis and Recognition, 1520-5363/11 2011 IEEE , DOI 10.1109/ICDAR.2011.157
[3]. Shinpei Yamazaki, Fumihiro Furukori, Qinzheng Zhao, Keiichiro Shirai and Masayuki Okamoto, "Embedding a Mathematical OCR Module into OCRopus" , 2011 International Conference on Document Analysis and Recognition, 1520-5363/11 2011 IEEE, DOI 10.1109/ICDAR.2011.180
[4]. Sushruth Shastry, Gunasheela G, Thejus Dutt, Vinay D S and Sudhir Rao Rupanagudi, ""i"-A novel algorithm for Optical Character Recognition (OCR)" , 978-1-4673-5090-7/13/ 2013 IEEE.
[5]. Ms. P.H. Pawar, Prof. R. P. Patil, "FPGA Implementation of Canny Edge Detection Algorithm", IJECS Volume 3 Issue 10 October, 2014 Page No.8704-8709.
[6]. K. Anil Kumar and M. Vijay Kumar, "Implementation of Image Processing Lab Using Xilinx System Generator" 10.14738/aivp.25.471, September 2014.
[7]. Rakhi P. Ghugardare, Sandip P. Narote, "Implementing OCR on FPGA", IET-UK International Conference on Information and Communication Technology in Electrical Sciences (ICTES 2007), Dr. MG.R. University, Chennai, Tamil Nadu, India. Dec. 20-22, 2007. Pp.805-810.
[8]. Obili Ramesh, P. V. Krishna Mohan Gupta, B. Sreenivasu, "A Real Time Hardware and Software Co-Simulation of Edge Detection for Image Processing System", International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 2 Issue 8, August – 2013.
[9]. Ravi.s, Abdul Rahim.B, Fahimuddin shaik, "FPGA Based Design and Implementation of Image Edge Detection Using Xilinx System Generator", International Journal of Engineering Trends and Technology (IJETT) – Volume 4 Issue 10 - Oct 2013 ISSN: 2231-5381.
[10]. DSP System Generator User guide release 12.1.Xilinx System Generator User's Guide,www.xilinx.com
[11]. "Digital Image Processing", Rafael C. Gonzalez, Richard E. Woods, Second Edison, Pearson publication.
[12]. CMOS Digital Integrated Circuits- Sung Mo Kang and
[13]. Yusuf Leblebici, Third Edison, Tata MacGraw-Hill
[14]. "A VHDL Primer", J Bhasker, Third Edison, Pearson Education 2003.

