IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 4, Issue 2, Ver. I (Mar-Apr. 2014), PP 49-53
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Object Counting Based On Image Processing: FPGA Approach

[1]Monali P. Patil, [2]Varsha R. Ratnaparkhe
Dept. of Electronics and Telecommunication Government Engineering College Aurangabad (M.S.), India

Abstract : *Image processing is used in very large and expanding areas covering applications in multimedia services, arts, medicine etc. For improving the performance of image processing systems the vital solution is implementation of image processing techniques in hardware. This paper presents the objects counting such as coins, with their implementation and simulation results using a hardware description language, VHDL.*
Keywords: *Digital image processing, Object Counting, FPGA, Hardware design languages, VHDL.*

I. INTRODUCTION

Now a days digital image processing is used in many applications such as multimedia services, arts, medicine, and automated industry. In many applications image processing is used for finding a group of pixels in an image that somehow belong together to finding out number of objects in image. For finding and counting objects in image some pre-processing is required such as enhancement of image so that object can be easily counted.

Objective of image enhancement is to process a given image so that the result is more suitable than the original image for a specific application. Image enhancement involves techniques to sharpen the image features such as edges, boundaries or contrast to make an image more useful for analysis.

Enhancement of images can be carried out in two different domains: spatial domain methods and spatial frequency domain methods. Spatial domain methods directly operate on the pixels and frequency domain methods operate on the mathematical transform of an image [1].

Some of the simplest, yet useful, image enhancement operations in the spatial domain are brightness manipulation, threshold operation, negative image transformation, contrast Stretching, histogram equalization, filtering. These are termed as point processing operations on image. Frequency domain methods include low pass filtering and high pass filtering.

For image processing applications portability is advantageous. For all applications it is not possible to utilize dedicated computer setup and image processing software like MATLAB. It is better to develop independent hardware design to give proper resource utilization and portability. The use of configurable hardware allows direct implementation of object counting algorithms with improved performances and reducing the computational and execution time.

There are many technologies are available for hardware implementation such as Application Specific Integrated Circuits (ASIC), Application-specific standard products (ASSPs), Digital Signal Processor (DSP) and Field Programmable Gate Arrays (FPGA's). ASIC design offers highest performance, but if any error occurs after fabrication then design cannot be changed. ASSPs are inflexible, expensive, and time consuming and DSPs are specialized microprocessors and they are very costly.

So a Field Programmable Gate Arrays (FPGA's) are better solutions which are able to solve this problem. FPGA consists programmable logic blocks and programmable interconnects that allow the same FPGA to be used in many different applications according to user's specification. FPGAs are reconfigurable devices able to support techniques such pipelining. FPGAs are flexible with a low development cost [2].

FPGAs are configured by using hardware design languages like Verilog HDL (Verilog) and Very High Speed Integrated Circuits (VHSIC) HDL (VHDL) [3].

In this proposed work algorithms are implemented using VHDL language because of some advantages of VHDL over Verilog. VHDL is strongly typed allowing finding bugs in a design in early verification process. Verilog is Weak typed language. VHDL is not case sensitive whereas Verilog is case sensitive language. User defined data types are supported in VHDL and not supported in Verilog [4].

The rest of the paper is organized as follows. Section II we discuss the methodology. Section III highlights implementation of proposed algorithm for counting objects such as number of coins. Section IV focused on experimental results. Finally, section V concludes the work.

II. METHODOLOGY

It is important that the object needs to be differentiated from the background. So some pre-processing is essential to count number of objects correctly .So mainly three enhancement techniques are used in this paper. 1) Brightness Manipulation, 2) Contrast Stretching, 3) Threshold Operation.

2.1) Brightness manipulation

Brightness manipulation is useful to enhance the brightness of image when image is captured in low light condition. After brightness manipulation it is easy to interpret the information in image easily.

Brightness manipulation operations are commonly used for increasing and decreasing Brightness. After brightness manipulation, dark image may become brighter or bright image may become dark .If we add the constant value in pixel value of image, then brightness manipulation operation increase brightness and similarly subtraction operator reduces the brightness as explained by following equation 1:

$$F(x,y) = \begin{cases} G(x,y) + t \ if \ G(x,y) + t \leq 255 \\ 255 \quad if \ G(x,y) + t > 255 \end{cases} \qquad (1)$$

Where $t \geq 0$

$G(x,y)$ – Gray Level for Input Pixel

$F(x,y)$ –Gray Level for Output Pixel [5]

2.2) Contrast Stretching

Contrast stretching is a technique in which some or all of the intensity value in the original image are stretched out to occupy a larger range of values to easily interpret the information in image [8]. It can be explain by following equation 2:

$$F(x,y) = \begin{cases} t_0 \ if \ G(x,y) < g_0 \\ G(x,y) \ if \ g_0 < G(x,y) < g_1 \\ t_1 if \ G(x,y) > g_1 \end{cases} \qquad (2)$$

Where $G(x,y)$ – Gray Level for Input Pixel

$F(x,y)$ –Gray Level for Output Pixel

t_0, t_1 - Constant values

g_0, g_1 – Pre-decided gray level which is to be highlighted from input image

Contrast stretching used to increase the brightness of object compare to background so that it can be easily interpreted. Contrast stretching is commonly used in application such as CT scans for enhancing the quality and contrast of medical images.

2.3) Threshold operation

Thresholding of image means converting gray level information of image to two-level information. In case the object in the foreground has quite different gray levels than the surrounding background, image thresholding is an effective tool for this separation, or segmentation. Threshold operation is defined using following equation 3 where a_{th} is the threshold value which is useful to separate the pixel values in two classes [2, 9]

$$F(x,y) = \begin{cases} G_0(x,y) \ for \ G(x,y) < a_{th} \\ G_1(x,y) \ for \ G(x,y) \geq a_{th} \end{cases} \qquad (3)$$

Where $G(x,y)$ – Gray Level for Input Pixel

$F(x,y)$ –Gray Level for Output Pixel

a_{th}- Threshold Value

By using equation 3, values of each pixel from the input image are replaced by the corresponding pixel in the destination image using $G_0\ (x,y)$=0 and $G_1\ (x,y)$=1.

2.4) Counting Algorithm

After the thresholding objects are get separated from the background. So it becomes easier to find group of pixels in an image that somehow belong together for finding out number of objects in image. . By

counting number of white pixels, numbers of objects are counted. Number of objects in image is shown by using LED's. Up to 256 objects can be shown by using LED's.

III. IMPLEMENTATION

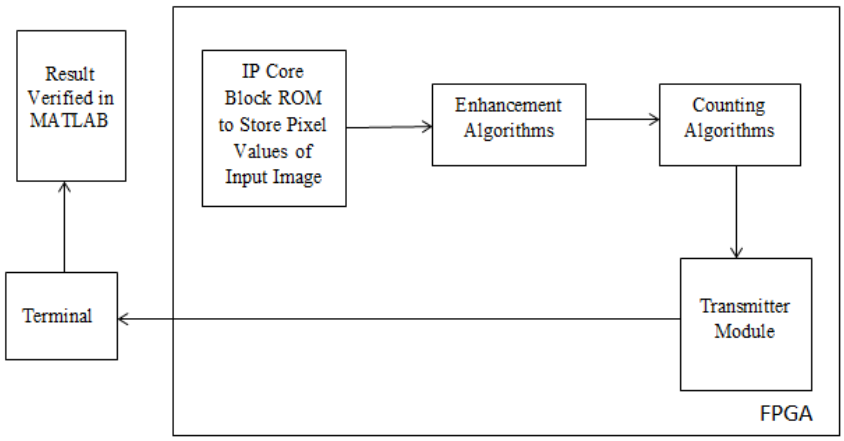

Fig.1: The Generalized Block Diagram of System

The generalized block diagram of a system to implement counting of objects algorithm is as shown in Fig.1. It consists of IP core ROM to store the input images, in this proposed work the size of input image is 128*128 so to store these many pixels the ROM size is also kept same as input image size. Enhancement algorithm block consist of algorithm of three enhancement techniques. Firstly, the input image gets enhanced by using brightness manipulation to increase the brightness of image as image is taken in low light then contrast stretching was done so that objects can be interpreted clearly and finally by using thresholding operation, separation the objects from background was carried out. By counting number of white pixels, numbers of objects are counted. Output of algorithm block is sent through hyper terminal which necessitates the use of transmitter block prior to hyper terminal. Finally output image is verified in MATLAB. The numbers of objects in image are indicated by using led on the FPGA.

3.1) IP Core (Block Memory Generator)

The IP Core (Intellectual property Core) refers to preconfigured logic functions that can be used in design. Xilinx provides a wide selection of IP that is optimized for Xilinx FPGAs. The CORE Generator System creates customized cores which delivers high levels of performance and area efficiency. In this work one IP Core is used, ROM IP Core to store the input image.

3.2) Transmitter Module In UART Protocol

Universal asynchronous receiver and transmitter (UART) is a circuit that sends parallel data through a serial line. The UART serial communication module consist of three sub-modules: the baud rate generator, receiver module and transmitter module, The baud rate generator is used to set the baud rate to control the UART receive and transmit; The UART receiver module is used to receive the serial signals at RXD, and convert them into parallel data; The UART transmit module converts the bytes into serial bits according to the basic frame format which is shown in Fig.2 and transmits those bits through TXD, basic frame format consist of start bit, data bits and stop bit. The transmission starts with a start bit, which is '0', followed by data bits and an optional parity bit, and ends with stop bits, which are '1'. The number of data bits can be 6, 7, or 8. The optional parity bit is used for error detection [10, 11].

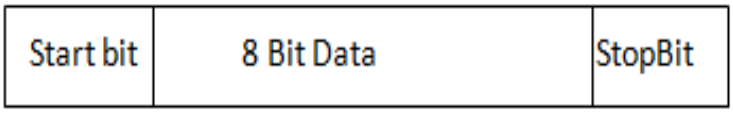

Fig.2: Basic Frame Format

In this proposed work only baud generator and transmit module is used to transfer the data from FPGA to PC. Any terminal or hyper terminal software can be used to receive data on PC through RS 232 port. Start bit and stop bit are attached to the 8 bit data and this 10 bit data is serial transmitted back to PC.

The complete RTL schematic of embedding the enhancement top module is presented in fig: 3 consist of mainly seven blocks, ROM IP block, brightness block, contrast block, thresholding block, counting block, LED block and transmitter block. Each block is designed separately using VHDL and all the seven blocks are brought in single top module to work together.

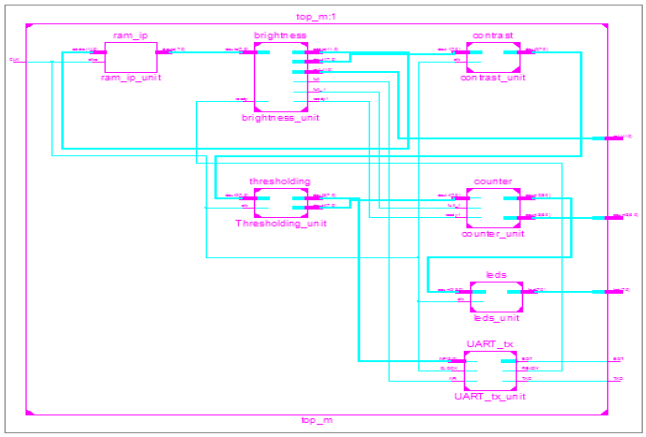

Fig.:3-RTL Schematic

IV. EXPERIMENTAL RESULTS

For hardware-based implementation of object counting algorithms, Spartan 3E family of FPGAs is used. The software development tool used for developing and verifying the design is the Xilinx's ISE 14.2-version. The image size considered for testing is a 128 x 128 pixel resolution gray-level image. The reason for considering a 128 x 128 resolution image is because of the limited memory size of FPGA.

Fig.:4 show the simulation result which is verified using ISE simulator. TXD signal is used to transmit the data serially including start bit, 8 bit data and stop bit, shown by green colure waveform when these 10 bits are transmitted successfully, end of transmission (EOT) signal become high. The final output after thresholding is shown by blue colure waveform, and number of object shown by black colure waveform which is 9. Fig.:5 shows the result verified in MATLAB.

The most basic element of the Spartan-3E device is a logic cell (LC), which contains a four-input LUT and a D FF. It also contains four types of Macro Blocks: Combinational Multiplier, Block RAM, Digital Clock Manager (DCM), and Input/Output block (IOB). So utilization of these devices are summarized in Table No.:1 as per table very less number of devices are utilized as the design is simple.

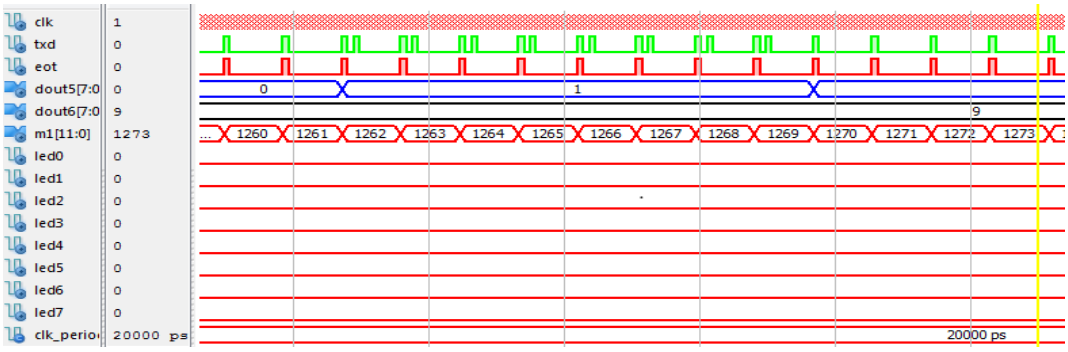

Fig.:4 Simulation Result

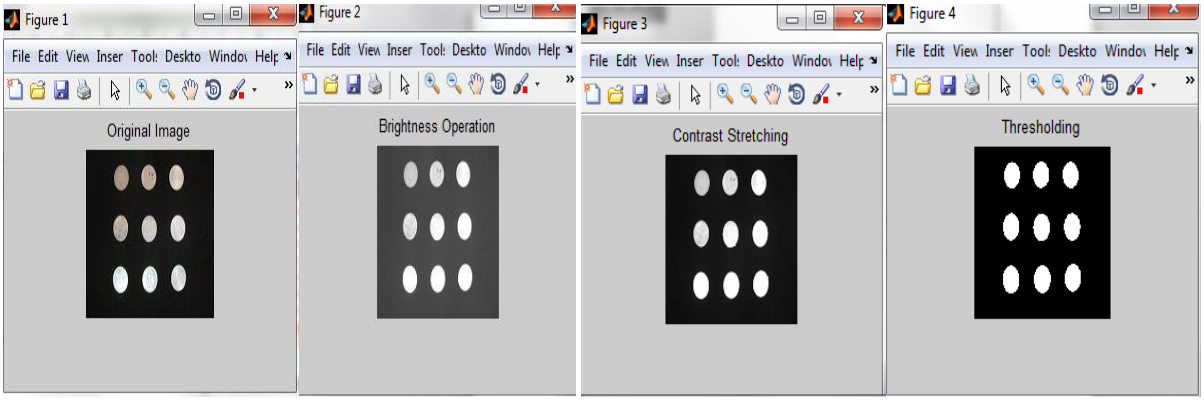

Fig:5 Result Verified in MATLAB

Device Utilization Summary			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Flip Flops	57	9,312	1%
Number of 4 input LUTs	19	9,312	1%
Number of occupied Slices	41	4,656	1%
Number of Slices containing only related logic	41	41	100%
Number of Slices containing unrelated logic	0	41	0%
Total Number of 4 input LUTs	47	9,312	1%
Number used as logic	19		
Number used as a route-thru	28		
Number of bonded IOBs	34	232	14%
Number of BUFGMUXs	2	24	8%
Average Fanout of Non-Clock Nets	2.08		

Table No.:1-Device Utilization Summary

V. CONCLUSION

The main goal of this proposed work is to implement simple and low cost portable system for counting the objects. As per the Table No.1the proposed design is simpler and memory efficient. In this paper hardware description language like VHDL is used which allows doing parameterizable designs and it gives the possibility of designing the circuit independent of the technologies used for its realization. The development of a software test bench in the same language lets us to check the designs in order to search the best solution for each application before doing the physical implementation. The simulation results were obtained are verified using MATLAB. This system has the advantages of being simple, flexible with reasonable development cost.

REFERENCES

[1] R C Gonzalez, R E Woods, "Digital Image Processing" 3rd Edition, Pearson Prentice Hall, 2004.
[2] Sparsh Mittal, Saket Gupta,and S. Dasgupta3 "FPGA: An Efficient And Promising Platform For Real-Time Image Processing Applications" Proceedings of the National Conference on Research and Development in Hardware & Systems (CSI-RDHS 2008) June 20-21, 2008, Kolkata, India.
[3] Iuliana CHIUCHISAN, Marius CERLINCA, Alin-Dan POTORAC, Adrian GRAUR "Image Enhancement Methods Approach using Verilog Hardware Description Language" 11th International Conference on Development And Application Systems, Suceava, Romania, May 17-19, 2012
[4] Stephen Bailey , "Comparison of VHDL, Verilog and SystemVerilog", Digital Simulation White Paper by Model Technology.
[5] S.Sowmya, Roy paily, "FPGA Implementation of Image Enhancement Algorithm," 978-1-4244-9799-7/111 IEEE 2011.
[6] M.Chandrashekar, U.Naresh Kumar, K.Sudarshan Reddy, and K.Nagabhushan Raju, "FPGA Implementation of High speed Infrared Image Enhancement," International Journal of Electronic Engineering Research vol.1, no. 3 2009, pp.279-285.
[7] Tarek.M.Bittibssi,Gouda Salama,Yehia.Z.Mehaseb, and Adel.E Henawy, "Image enhancement Algorithms using FPGA," International Journal of Computer Science and Communication Networks vol.2, no. 4, pp. 536-542.
[8] U. Bidarte, J. A. Ezquerra, A. Zuloaga, J. L. Martín "VHDL Modeling of an Adaptive Architecture for Real-Time Image Enhancement"
[9] Azeema Sultana, Dr. M. Meenakshi, "Design and Development of FPGA based Adaptive Thresholder for Image Processing Applications", 978-1-4244-9477-4/11 IEEE 2011.
[10] FANG Yi-yuan, CHEN Xue-jun, "Design and Simulation of UART Serial Communication Module Based on VHDL", 978-1-4244-9857-4/11 IEEE 2011.
[11] Pong P. Chu , "FPGA PROTOTYPING BY VHDL EXAMPLES", John Wiley & Sons, Inc.2008.
[12] Douglas Perry, "VHDL" 3rd Edition, Tata Mcgraw-Hill, 2001.