International Journal of Electronic and Electrical Engineering.
ISSN 0974-2174 Volume 3, Number 3 (2010), pp. 139--145

http://www.irphouse.com

VLSI Implementation of Image Processing Algorithms on FPGA

Varsha S. Surwase and S.N. Pawar

Faculty, V.P.C.O.E. Baramati-413133, Maharashtra, India
Faculty, J.N.E.C. Aurangabad, India
E-mail: varsha.surwase@gmail.com, nil_pawar@yahoo.com

Abstract

This paper describes implementation of image processing algorithms on FPGA.

Currently, implementation of image processing algorithms using software approach is slower due to limited speed of the processor. So a dedicated processor for implementation of image processing algorithms is required which is possible with VLSI technology. FPGA's have the advantages of speed and re-configurability which is required for image processing applications. This paper deals with image enhancement algorithm. This paper presents three techniques of image enhancement algorithms of Negative transformation, Thresholding & Contrast stretching.

The aim of this paper is to simulate and implement these algorithms using Matlab and VHDL. The device selected here for implementation is Sparten 3E500K from Xilinx.

Keywords: FPGA,VLSI,Enhancement.

Introduction

Recently, Field Programmable Gate Array (FPGA) technology has become a viable target for the implementation of algorithms suited to video image processing applications. The unique architecture of the FPGA has allowed the technology to be used in many such applications encompassing all aspects of video image processing.[1]

The aim of this paper is actual implementation of proposed image enhancement algorithms on target FPGA hardware. This is accomplished by composing algorithms in VHDL language and synthesizing the algorithms for FPGAs.

Block Diagram

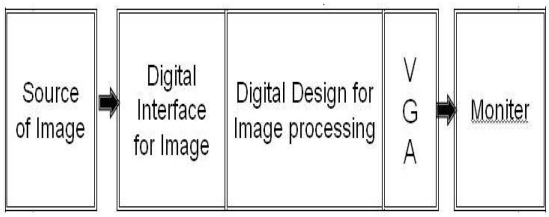


Figure 1: Block Diagram of system.

As shown in block diagram of Fig 1, PC is the source of image which will send image to digital hardware (FPGA).The image is sent to digital hardware using Transport utility of digilent board. The gray values of image received from PC are processed using digital hardware designed on FPGA.

Digital hardware consists of digital design for various spatial domain image enhancement techniques such as negative image, thresholding, contrast stretching, etc. The gray values are modified as per selected image enhancement technique. The processed image is then given to VGA controller. [2] VGA controller will display the enhanced image on monitor .VGA controllers are designed in FPGA architecture itself. The VGA monitor is controlled by five signals: red, green, blue, horizontal synchronization, and vertical synchronization. The digital design operates at 25 MHz clock frequency which is able to process 640×480 grayscale image.

FPGA for Image Processing

Application Specific Integrated Circuits (ASICs) represent a technology in which engineers create a fixed hardware design using a variety of tools. Once a design has been programmed onto an ASIC, it cannot be changed. Since these chips represent true, custom hardware, highly optimized, parallel algorithms are possible. However, except in high-volume commercial applications, ASICs are often considered too costly for many designs. In addition, if an error exists in the hardware design and is not discovered before product shipment, it cannot be corrected without a very costly product recall. [3]

Field Programmable Gate Arrays (FPGAs) represent reconfigurable computing technology, which is in some ways ideally suited for video processing. Reconfigurable computers are processors which can be programmed with a design, and then reprogrammed (or reconfigured) with virtually limitless designs as the designer's needs change. FPGAs generally consist of a system of logic blocks (usually look up tables and flip-flops) and some amount of Random Access Memory (RAM), all wired together using a vast array of interconnects. All of the logic in an FPGA can be rewired, or reconfigured, with a different design as often as the designer likes. This type of architecture allows a large variety of logic designs dependent on the processor's resources, which can be interchanged for a new design as soon as the device can be reprogrammed.

Today, FPGAs can be developed to implement parallel design methodology, which is not possible in dedicated DSP designs. ASIC design methods can be used for FPGA design, allowing the designer to implement designs at gate level. However, usually engineers use a hardware language such as VHDL or Verilog, which allows for a design methodology similar to software design. This software view of hardware design allows for a lower overall support cost and design abstraction.

Digital Design of Enhancement

Negative Transformation

In this technique dark image pixels are converted into bright and vice Versa. Suppose input image is 8-bit then each pixel gray level value is subtracted from 255 to get its negative image.[4] This can be implemented in hardware by using one 8 bit subtractor which will perform the required subtraction shown in fig.2

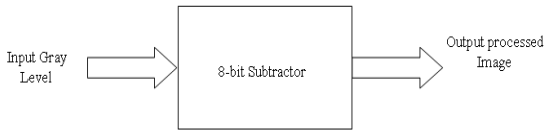


Figure 2: Negative transformation.

Thersholding –

In this technique each gray level is compared with specific threshold which is set by the user. [4]If incoming pixel gray level is less than threshold, then at the output zero value is passed else 255 i.e. highest value is passed. Shown in fig.3

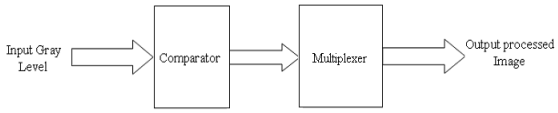


Figure 3: Thersholding.

Contrast Stretching-

Contrast stretching is implemented with two threshold values. These values are taken from user. Incoming pixel value is compared with these thresholds and contrast stretching is carried out.[4] If input is less than threshold, then it is stretched to dark by subtracting some constant and if it is greater than threshold then it is stretched towards bright.

VGA signal generation

The term *Video Graphics Array (VGA)* refers either to an analog computer display standard. VGA was the first graphical standard that the majority of manufacturers

conformed to, making it the lowest common denominator that all PC graphics hardware supports before a device-specific driver is loaded into the computer. [2]

The simple VGA controller is designed in FPGA itself. This VGA controller generates three colors signals red, green, blue and two control signals Hsync and Vsync. Analog levels between 0v (completely dark) and 0.7V (maximum bright) on the colors signals tells the monitor what intensities of these primary colures to combine. Each analog colors input can be set to one of four levels by using two digital outputs. So a pixel can have one of 64 different colors and only 4 grey shades.

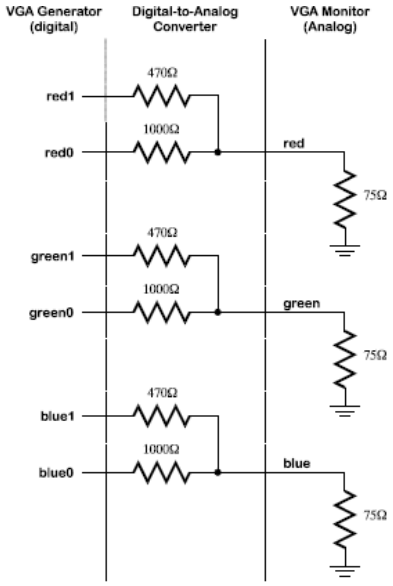


Figure 4: Digital to analog VGA monitor interface.

VGA clock frequency and resolution

The VGA controller operates at clock frequency of 25 MHz. The actual pixels are sent to monitor within 25.17μs window. With 25 MHz clock we get the horizontal resolution of 640 pixels and vertical resolution of 480 pixels.[2]

Results

Simulation result

Fig 5 shows the simulation result for negative transformation. Here input pixel values are inverted i.e. they are subtracted from 255 .

Input pixel values : 80 66 8 0

Output pixel values: 175 189 247 255

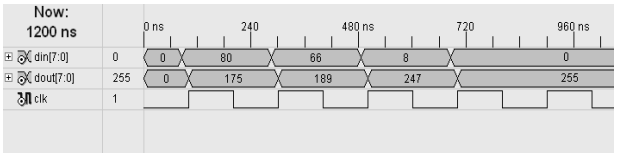


Figure 5: Result of VHDL simulation for Negative transformation.

Input image Output image

Figure 6: Result on monitor Negative transformation.

Simulation result

Fig 6 shows the simulation result for thresholding. Here input pixel values are mapped to either zero or 255 by comparing them with threshold as described earlier. Here threshold is set to 64.

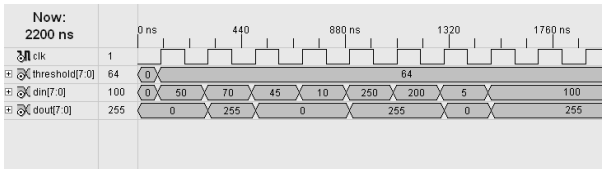


Figure 7: Thresholding simulation result.

Input image Output image

Figure 8: Result on monitor Thresholding.

Fig 8 shows the simulation result for Contrast stretching .Here input pixel values are stretched towards either zero or 255 by comparing them with two thresholds or they are kept as it are if they fall between these two thresholds. Here thresholds are set to 64 and 150.

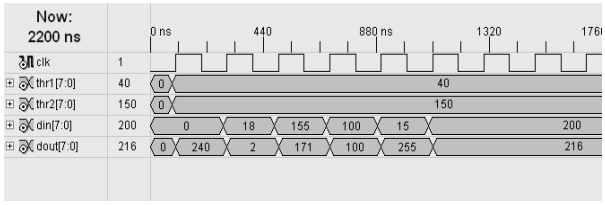
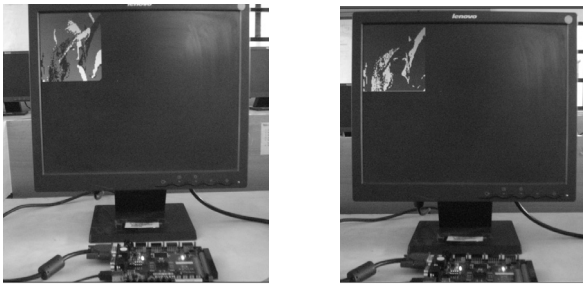


Figure 9: Contrast stretching simulation result.



Input image Output image

Figure 10: Result on monitor contrast stretch.

Conclusion and Future work

By using this paper, high speed image enhancement applications can be implemented on digital image & we can implement this image on FPGA devices.

Producing digital images with good brightness/contrast and detail is a strong requirement in several areas like vision, remote sensing, biomedical image analysis, fault detection. Producing visually natural images or transforming the image such as to enhance the visual information within, is a primary requirement for almost all vision and image processing tasks.

Any medical imaging application where contrast enhancement and sharpening is needed. The potential areas of applications include digital X-ray, digital mammography, CT scans, MRI etc.

The same image enhancement techniques can be applied on RGB color images instead of gray scale images for Manual and automatic color correction.

Thus we have seen some spatial domain image enhancement techniques. These techniques are also called point processing techniques as they are operated on pixels directly. In this project, we have taken these image enhancement techniques for VLSI implementation on FPGA hardware.

References

[1] Peter A Ruetz and Robert W Brodersen," *Architectures and design techniques for real time image processing IC's"* ,fellow IEEE,Electronics research library, University of California.

[2] Anthony E. Nelson, "*Implementation of image processing algorithms on FPGA hardware*", Graduate school of Vanderbilt University, May 2000.

[3] Yiran Li "*FPGA Implementation for Image Processing Algorithms*" , EEL 6562 Course Project Report, December 2006

[4] R.C.Gonzalez and R.E.Woods, "*Digital Image Processing*" Reading MA: Addison – wesely Publication, 1992.

[5] Mathivanan,"Microprocessor, Pc Hardware and Interfacing." PHI.

[6] Sparten3E FPGA data sheet.

[7] Diligent Nexys2 Board Reference Manual.