FPGA Implementation of Image Enhancement using Verilog HDL

Prof. Narayan A. Badiger[1] , Miss.Jayamma Muragod[2] , Miss.Poornima Pattar[2] ,Miss. Priyanka Gundlur[2] , Miss.Savita Bhadri[2]

[1]*Assistant Professor, Dept. Electronics & Communication Engineering, S.G. Balekundri Institute of Technology, Belagavi, Karnataka, India.*

[2]*Undergraduate Students, Dept. Electronics & Communication Engineering, S.G. Balekundri Institute of Technology, Belagavi, Karnataka, India.*

---***---

Abstract - *Image enhancement is one of the challenging issues in low level image processing. This paper discusses basic image enhancement techniques with their implementation and results using a Hardware Description Language, Verilog. The use of HDLs to provide signal processing results is quite new technique replacing the classical simulations and operating a direct connection to hardware VLSI implementations. The purpose of this paper is designing, modeling, simulation and synthesis of four image enhancement techniques on FPGA. Image enhancement operations can be classified as a point processing techniques, in which operation is done on pixel level and Spatial Filtering Technique, in which operation is performed within neighborhood of a pixel.*

Key Words: **Verilog; FPGA, Spartan 6; Image Enhancement**

1. INTRODUCTION

Image enhancement is the procedure of improving the quality and the information content of original data before processing. Image enhancement widely used in computer graphics, and is the sub areas of image processing. Image enhancement refers to accentuation, or sharpening of image features such as edges, boundaries, or contrast to make a graphic display more useful for display and analysis.

The use of HDLs provides designers with the flexibility of simulating and evaluating the digital circuit performance with timing considerations. This flexibility can be extended into the field of digital signal processing. However, HDLs provide the main advantage of immediate FPGA based hardware implementation. Field Programmable Gate Arrays have traditionally been configured by hardware designers using specific so called Hardware Design Languages (HDLs). There are already few such languages available offering different levels of abstraction but the most important ones are Verilog HDL (Verilog) and Very High Speed Integrated Circuits (VHSIC) HDL (VHDL).

1.1 IMAGE ENHANCEMENT TECHNIQUES

Image enhancement aims at improving the interpretability or perception of information in image for human viewers, or providing 'better' input for other automated image processing techniques.

Image enhancement techniques can be classified into two categories:

1. Spatial domain technique:
Spatial domain technique refers to enhancement of image based on operations performed directly on pixels of the image.

Types of spatial domain techniques:
a. Range operations:
  1. Point operations.
  2. Neighborhood operations.
b. Domain operations.

2. Frequency domain technique:
It achieves enhancement through the use of mathematical transforms such as Fourier transforms and it is straightforward. The idea of blurring an image by reducing its high frequency components, or sharpening an image by increasing the magnitude of its high frequency components is intuitively easy to understand.

1.2. POINT OPERATION

Point operations are nothing but simplest image filters, where the new value of a pixel is only determine by the original value of that single pixel alone. As mentioned, point operations perform modification of pixel values without changing the size, geometry or local structure of the image. The pixel value is given by $a = I (u, v)$, which depends exclusively on the previous value $a = I (u, v)$ at the same position. This pixel value is independent from any other pixel value, including any of its neighbouring pixels. To map the original pixel

values to the new values a function *f (a)* is used, *a' ← f(a) I' (u, v) ← f (I (u, v))* for each image point with a spatial coordinate (u, v). The most commonly used point operations for image enhancement are:
A) Threshold Operation
B) Invert Operation
C) Brightness Operation
D) Contrast Operation

2. LITERATURE REVIEW

1."Image Enhancement Methods Approach Using Verilog Hardware Description Language"presented by I.Chiuchisan, M. Cerlinca, 11th International Conference on DEVELOPMENT AND APPLICATION SYSTEMS, Suceave, Romania, May 17-19, 2012: Given the importance of digital image processing based on hardware implementations in order to achieve higher performance, this paper discusses basic image enhancement techniques with their implementation and results using a hardware description language, Verilog. The use of HDLs to provide signal processing results is a quite new technique replacing the classical simulations and offering a direct connection to hardware VLSI implementations. This paper is providing an innovative method for simulation followed by immediate implementation possibility. The present HDL approach is applied to image processing and accordingly an overview of underlying principle and concepts, along with common algorithms usually used for image enhancement are described. The paper focuses on image enhancement in the spatial domain, with particular reference to point processing methods like: contrast manipulation, brightness manipulation, inverting images, threshold operation.

2."A Review on Image Enhancement using Hardware co-simulation for Biomedical Application" presented by Kalyani A. Dakre, International Journal of Advanced Research in Computer Engineering & Technology(IJARCET) Volume 3 Issue 12, December 2014:FPGAs are providing a platform for processing real time algorithms on application-specific hardware with substantially higher performance than programmable digital signal processors (DSPs). This project focus on implementation issues of image enhancement algorithms like brightness control, contrast stretching, negative transformation, thresholding, filtering techniques on FPGA that have become a competitive alternative for high performance digital signal processing applications. This project will use System Generator tool and modular construction methods to build a image algorithm platform in MATLAB. By a brief analysis about display image and resource consumption after achieving on Spartan-3E development board, we can see the image using System Generator for FPGA algorithm design superiority, have the vast application prospects.

3. "An Implementation of Image Enhancement on Real Time Configurable system using HDL" published by Mahavir Singh, Gitanjali Pandove, in International Journal of Advanced Research in Electronics and Communication engineering volume 7,issue 3, March 2018: Whenever the one form of image is converted to another form, the degradation occurs in the quality of the original image. Therefore to improve the visuality of the image, the image enhancements techniques are used for the improvements. These Techniques are the contrast stretching, brightness control, invert operation, threshold operation etc. The image enhancement can be done with the help of the both software implementation and hardware implementation. But as the hardware implementation has better performance than the software implementation therefore we use the configurable system that is in real time to enhance the image quality. Image processing with the help of hardware description languages is the technique with new approach in the area of digital system design using VLSI.

4."Image Enhancement on FPGA using Verilog" published by Dr. Sagar Patel1, Krinesh Patel2, Keval Patel3 and Chaitanya Patel4, in International Journal of Technical Innovation in Modern Engineering &Science (IJTIMES) Impact Factor: 5.22 (SJIF-2017), e-ISSN: 2455-2585 Volume 5, Issue 03, March-2019: Image magnification techniques are to improving the visual quality of Image. essential objective of image enhancement is to process an image in order that end result is greater suitable than original image for specific application. This paper shows constant equipment image enhancement methods utilizing field programmable gate array (FPGA). image enhancement algorithms are brightness addition, brightness subtraction, inverting and thresholding use in this paper. FPGA is highly alternative of fast performance of digital signal processing application. These algorithms are implemented in Verilog HDL using Xilinx ISE, MATLAB.

Summary: The purpose of Literature survey is to understand the existing technology which can be improvised according to the advanced technology. It helps in understanding the existing or implemented technology which has to be worked on.

3. OBJECTIVE

1. Image enhancement is to process an image in order that end result is greater suitable than original image for specific application, like weather forecasting.

2. Image enhancement aims at improving the interpretability or perception of information in image for human viewers, or providing better input for other automated image processing techniques.

3. Image enhancement techniques help in improving the visibility of any portion or feature of the image suppressing the information in other portion or features.

4. It involve different processes including image quality enhancement and object detection.

5. We implemented four basic operations of Image Enhancement i.e. Threshold, Contrast, Brightness, Invert.

4. METHODOLOGY

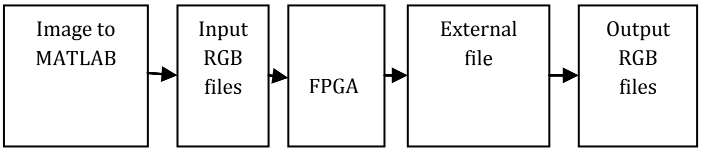


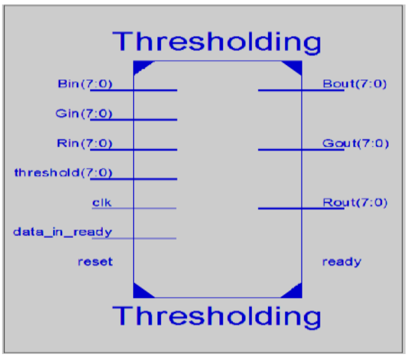
Fig -1: Block Diagram of the System

The Verilog language has the power to browse or write files from a storage setting. This feature create it potential to significantly style the test benches to browse the test information from device, generate the stimulant signals to the Verilog check module and write back the results to the device. We need image but image file will be in .bmp file extension so this file is not readable to Xilinx Software. Moreover, here image convert to hex file using MATLAB.

The project is implemented on FPGA using Verilog, which is Hardware Description Language. The code written in Verilog describes the behavior of the desired hardware. The Hardware Description Language, Verilog HDL was developed to carry out readings and writings of files with ASCII characters and it does not allow to process bitmap or jpeg files. For that reason, it is necessary to represent binary information with ASCII characters in the hex format. Hex characters are quickly and easily converted to binary format by Verilog HDL. In order to resolve this problem it was defined a new image format to be used with the test bench described. The hex-file contain only information about RGB vector for each pixel of the input image and does not contain information about image dimensions or similar. The data from hex-file was applied as stimulus to the point operations blocks described in Verilog language. The result was obtained in another external file and we create an application described in Visual Studio to show the modified output image and to compare with the original input image.

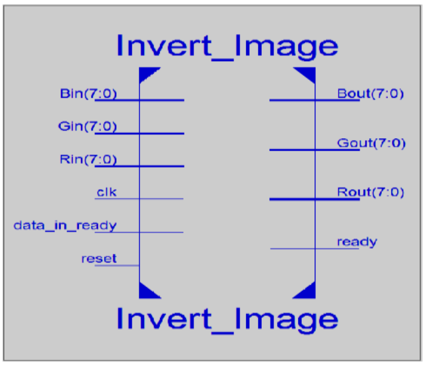
In this project we use four basic operations of image enhancement threshold, contrast, brightness, invert to manipulate the RGB values of every pixel of the image to improve the human interpretation of image.

4.1 Threshold operation: Thresholding operations are particularly interesting for segmentation in the process of isolating an object of interest from its background. Thresholding an image means transforming all pixels in two values only. Set the pixel above a threshold value to 255 and below it to 0.



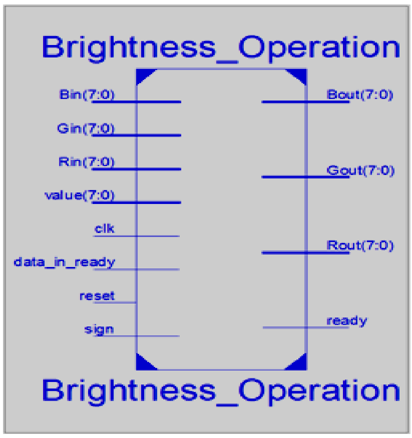

```
if (value > threshold) begin
        Rout = 255;
        Gout = 255;
        Bout = 255;
end
if (value < threshold) begin
        Rout = 0;
        Gout = 0;
        Bout = 0;
end
```

4.2 Invert operation: Inverting an intensity image is a simple point operation that reverses the ordering of pixel values (by multiplying with−1) and adds a constant value to map the result to the admissible range again.RGB pixel values must be equalized and it is done by taking the average of the three color components.




```
value2 = (Rin + Gin + Bin)/2;
value4 = (Rin + Gin + Bin)/4;
value  = (value2 + value4)/2;
     Rout = 255 - value;
     Gout = 255 - value;
     Bout = 255 - value;
```

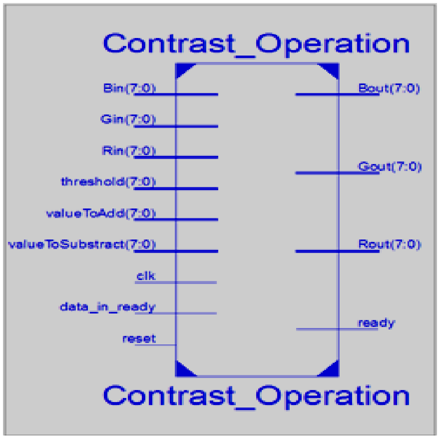
4.3 Brightness operation: A dark region in an image may become brighter after the point operation and the common used point operation are increasing and decreasing of brightness. If an operator takes each pixel value and adds a constant number to it, then this point operation increases the brightness of the image and similar subtraction operation reduces the brightness.




```
if(sign == 1) begin
        tempR = Rin + value;
        if (tempR > 256)
                Rout = 255;
        else
                Rout = Rin + value;
        tempG = Gin + value;
        if (tempG > 256)
                Gout = 255;
        else
                Gout = Gin + value;
        tempB = Bin + value;
        if (tempB > 256)
                Bout = 255;
        else
                Bout = Bin + value;
end
```

```
if(sign == 0) begin
        tempR = Rin - value;
        if (tempR[8] == 1)
                Rout = 0;
        else
                Rout = Rin - value;
        tempG = Gin - value;
        if (tempG[8] == 1)
                Gout = 0;
        else
                Gout = Gin - value;
        tempB = Bin - value;
        if (tempB_b[8] == 1)
                Bout = 0;
        else
                Bout = Bin - value;
        end
```

4.4 Contrast operation: Expanding the contrast range by assigning the darkest pixel value to black, the brightest value to white, and each of the others to linearly interpolated shades of gray makes good use of the display and enhances the visibility of features in the image.



```
if (value > threshold) begin
        tempR = Rin + valueToAdd;
        if (tempR > 256)
                Rout = 255;
        else
                Rout = Rin + valueToAdd;
        tempG = Gin + valueToAdd;
        if (tempG > 256)
                Gout = 255;
        else
                Gout = Gin + valueToAdd;
        tempB = Bin + valueToAdd;
        if (tempB > 256)
                Bout = 255;
        else
                Bout = Bin + valueToAdd;
end
if (value < threshold) begin
        tempR = Rin - valueToSubstract;
        if(tempR[8] == 1)
                Rout = 0;
        else
                Rout = Rin - valueToSubtract;
        tempG = Gin - valueToSubstract;
        if(tempG[8] == 1)
                Gout = 0;
        else
                Gout = Gin - valueToSubtract;
        tempB = Bin - valueToSubstract;
        if(tempB[8] == 1)
                Bout = 0;
        else
                Bout = Bin - valueToSubtract;
end
```

5. ADVANTAGES

1. No processing or fixing chemicals are needed to take and process digital images.
2. Unrecognizable features can be made prominent.
3. Images can be smoothened.
4. It allows robots to have vision.
5. It allows industries to remove defective products from the production line.

6. RESULTS

The simulation results obtained after applying the operations described using Verilog HDL to an input image are shown here.

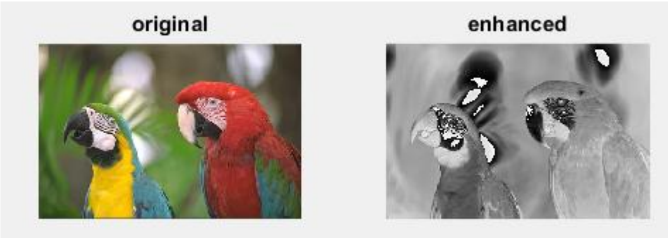


Fig - 2: Verilog result for Invert Operation

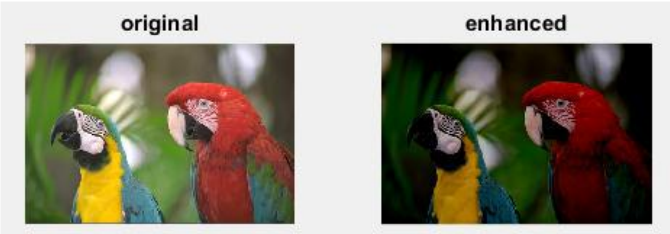


Fig - 3: Verilog result for Contrast Operation using threshold = 90, valueToAdd = 10 and valueToSubtract = 15

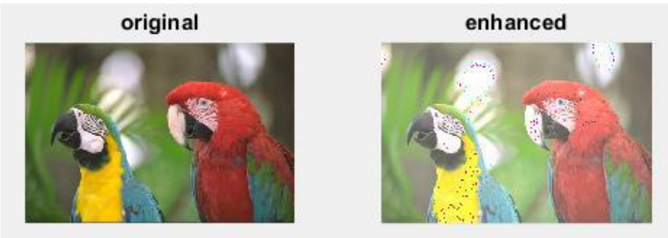


Fig - 4: Verilog result for Brightness Operation using sign = 0 and value = 60

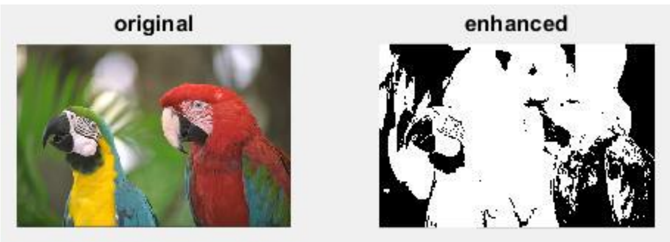


Fig - 5: Black & White result using threshold =120

7. CONCLUSION AND FUTURE SCOPE:

Image Processor has been designed and implemented using Verilog HDL and simulated using ISim from Xilinx ISE Design Suite 14.3 and synthesized using Xilinx XST. With lot of image processing creeping into the PDA and other hand held equipment, there is an urge for providing a specialized Image Processor. This work gives an insight into the most generalized architecture that can be customized for other image processing applications too. Though, the most fundamental point operations on the image are discussed, the idea may be carried forward for designing filtering applications also. The major challenge in this work is to choose a proper FPGA for prototyping, since the memory buffer needs enormous memory, the crucial aspect is to choose such an FPGA which has enough RAM, FIFO resources.

REFERENCES:

[1] John C. Russ - "Image Processing Handbook (sixth edition)", CRC Press, pp. 270-331, 2011.

[2] Raman Maini, H. Aggarwal - "A Comprehensive Review of Image enhancement Techniques", Journal of Computing, vol. 2, issue 3, ISSN 2151-9617, pp. 269-300, 2010.

[3] Wilhelm Burger, Mark J. Burge - "Principles of Digital Image Processing – Fundamental Techniques", Undergraduate Topics in Computer Science, DOI 10.1007/978-1-84800-196_4, Springer- Verlag London Limited, 2009.

[4] A. Zuloaga, J.L. Martin, U. Bidarte, J.A. Ezquerra - "VHDL test bench for digital image processing systems using a new image format", ECSI,2007 (http://mx.reocities.com/CapeCanaveral/8482/).

[5] "image Enhancement on FPGA using Verilog" published by Dr. Sagar Patel, Krinesh Patel, Keval Patel and Chaitanya Patel, in International Journal of Technical Innovation in Modern Engineering &Science (IJTIMES) Impact Factor: 5.22 (SJIF-2017), e-ISSN: 2455-2585 Volume 5, Issue 03, March-2019.

[6] "FPGA Implementation of Image Enhancement Techniques" presented by Karan Kumar, Aditya Jain, and Atul Kumar Srivastava, The International Society for Optical Engineering · Vol. 7502 750208-1, JUNE 2009.

[7] "Image Enhancement Methods Approach Using Verilog Hardware Description Language" presented by I. Chiuchisan, M. Cerlinca, 11th International Conference on DEVELOPMENT AND APPLICATION SYSTEMS, Suceave, Romania, May 17-19, 2012.

[8] "High-Level Design for Image Processing on FPGA using Xilinx AccelDSP" presented by Yahia, Laboratory of Electronics and Microelectronics(E) Faculty of Sciences, 2014.

[9] "A Review on Image Enhancement using Hardware co-simulation for Biomedical Application" presented by Kalyani A. Dakre, International Journal of Advanced Research in Computer Engineering & Technology(IJARCET) Volume 3 Issue 12, December 2014.

[10] DhirajR. Gawhane, Karri Babu Ravi Teja, AbhilashS. Warrier, AkshayS. Belvadi "Design and Implementation of a Digital Image Processor for Image Enhancement Techniques using Verilog Hardware Description Language" Dhiraj R Gawhane et al , Int.J.Computer Technology & Applications,Vol 5 (1),87-91,2014.