

# FPGA-based Implementation of Color Image Processing Techniques

F. S. Desai, R. R. Bhat, H. N. Shah, S. S. Rathod and P. V. Kasambe

**Abstract**—This paper describes an image processing system developed to work on an FPGA (Field Programmable Gate Array). FPGAs have the advantages of speed and reconfigurability over DSP, specifically required for video processing applications. The research work done on color image processing techniques using FPGA is limited. The aim of this research is to simulate and implement color image processing techniques namely pseudocolor processing, smoothing filter, median filtering and edge detection masks on FPGA. This simulation is carried out by establishing a link between MATLAB and a Hardware Description Language (HDL). Comparison of HDL simulation results show similarity with those obtained from standardized MATLAB functions.

**Index Terms**—Color Images, FPGA, VLSI, Image Processing.

## I. INTRODUCTION

IN the recent times, Field Programmable Gate Arrays have become a viable target for the implementation of various algorithms related to color video image processing applications. The unique architecture of FPGA supports many applications concerning high speed processing. With increase in image sizes, the use of software for image processing are being replaced by real time systems required for high speed image and video processing. Specialized image processing programs cannot adequately process very large amounts of streaming data on PCs since they have been designed for general purpose applications. In order to achieve further optimization, hardware devices need to be used. Increase in FPGA speeds and capacities have followed or exceeded Moore's law for the last several years.

---

Faheem Desai is with Department of Electronics Engineering at Sardar Patel Institute of Technology, Mumbai 400058 INDIA.  
(e-mail:faheemd2001@yahoo.com).

Rohit Bhat is with Department of Electronics Engineering at Sardar Patel Institute of Technology, Mumbai 400058 INDIA.  
(e-mail:bhatrohit2000@yahoo.com).

Harsh Shah is with Department of Electronics Engineering at Sardar Patel Institute of Technology, Mumbai 400058 INDIA.  
(e-mail:hardison87@yahoo.co.in).

S. S. Rathod is with Department of Electronics Engineering at Sardar Patel Institute of Technology, Mumbai 400058 INDIA.  
(e-mail:rathod\_spce@yahoo.com).

P.V. Kasambe is with Department of Electronics Engineering at Sardar Patel Institute of Technology, Mumbai 400058 INDIA.  
(e-mail:p\_vasantrao@rediffmail.com).

FPGAs do not have inbuilt cache mechanisms. They are usually connected to a number of local memories so as to maximize throughput. However, the available space in FPGA imposes a restriction on the size of a block of code which is to be executed in a single clock cycle. As long as the code fits on the chip, the performance of the FPGA is largely based on Memory Traffic and Clock Frequency. Results have shown that complex tasks have resulted in very large speedups as high as 800 by utilizing the parallelism within FPGAs [3]. Our goal is to familiarize application programmers with the state of the art in compiling programs using hardware description language to FPGA and to show how FPGAs implement a wide range of color image processing applications.

## II. METHODOLOGY

The FPGA based designing approach is via two HDLs – VHDL and Verilog. The limitation of HDL programming is the difficulty in implementing algorithmic expressions that takes place in applications programming. The Cameron project's SA-C (Single Assignment C language) and Celoxica's Handel C are two popularly used cross-compilers which map high-level algorithmic language on reconfigurable hardware [2]. However, VHDL was chosen in this paper due the flexibility and controllability it offers, over timing constraints and hardware.

This paper studies two methodologies to simulate a link between an Image Acquisition System and an FPGA device namely Direct Software Link and File I/O routines. MATLAB was chosen as the image end software due to its capability to work directly with matrices. The MATLAB environment is much suited for PC based image processing applications. MATLAB allows the designer to treat a color image as a 3-dimensional matrix of red, green and blue information of 8 bits each and develop optimized matrix operations. In this paper, MATLAB is used for rasterization of color image into pixels. These pixels are then transported to required HDL simulation software via the two discussed methodologies.

### A. Direct Software Link

This link establishes a connection between Model-Sim and MATLAB through software protocols [2]. 8bit ports are defined both in MATLAB and Model-Sim. Initialization file for Model-Sim is defined to implement handshake signals for synchronous transfer. A GUI is built to feed in the input image file and designate the output VHDL file. The

communication “.m” file handles the actual transfer of 8bits per a common clock cycle. The R-Matrix, G-Matrix and B-Matrix can be sent in successive streams as well as in parallel form required for certain applications.

*B. File I/O Routines*

This link uses the file handling capability of EDA software to simulate the transfer of image data to an FPGA [1]. MATLAB is used to save the image matrix as binary text file having every word on a new line. The VHDL file read routine is then used to access the file serially and save it using the write routine after processing. MATLAB is then again used to access this file and save it as the transformed image file.

III. RESULTS AND DISCUSSIONS

Using the links discussed in section II, a transfer was established to Model-Sim and following color image processing applications were simulated and observed:

*A. Pseudocolor Image Processing*

This technique consists of assigning colors to gray values based on specified criterion. The principal use is for human visualization and interpretation of grayscale events in an image or sequence of images. An example of such a technique is intensity slicing [4].

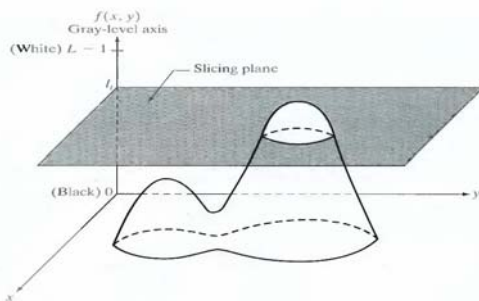


Fig. 1. Geometrical interpretation of intensity –slicing technique

Figure shows a plot of gray scale intensity versus spatial coordinates. The method consists of placing planes parallel to the coordinate plane of the image. Each plane then slices the function in the area of intersection. If a different color is assigned to each side of the plane, every pixel in a particular range of gray intensity will have a corresponding color value coded to it.

A biomedical application for this technique is the pseudocoloring of the Picker Thyroid Phantom test pattern which has been implemented in this simulation. The gray scale values of this test pattern image are sent to the VHDL routine. This routine divides the gray plane into 8 different intensities. These intensities are assigned 8 values with white indicating the highest intensity and black indicating lowest. The color pixel values are then obtained in MATLAB and saved as an image file. Fig. 4 shows RTL schematic after

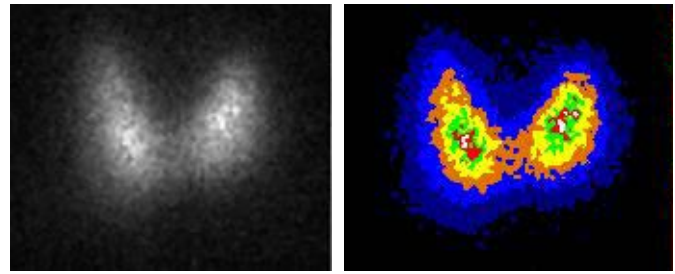


Fig. 2. Results from HDL simulation of intensity slicing

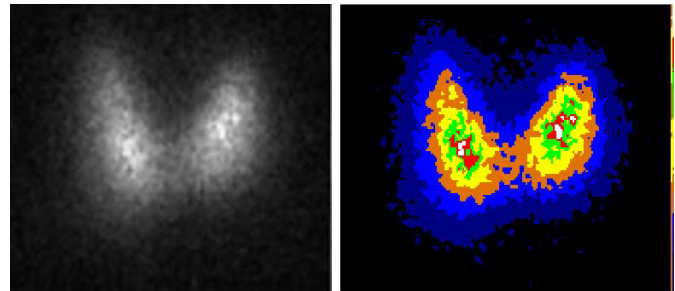


Fig. 3. Results from MATLAB simulation of intensity slicing

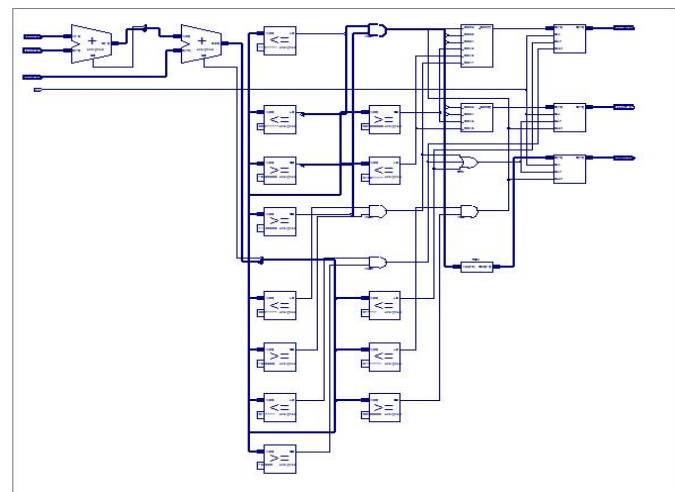


Fig. 4. RTL schematic after synthesis of pseudocoloring on Xilinx ISE

*B. Median Filtering*

Median filter is an order statistics filter which is spatial in nature whose response depends on ordering the pixels contained in the image area covered by the filter [4]. The process of median filtering is useful for restoration of images corrupted by salt and pepper noise and also to bring about smoothing of other noise that may not be impulsive. Median Filtering is more effective than convolution when the goal is to simultaneously reduce noise and preserve edges. Median filtering is achieved using a windowing operator.

This pixel window may be of any size or shape. The efficiency of noise filtering increases as the size of window increases. This paper employs a 3x3 pixel window as it is large enough to give accurate results. At the same time it is small enough and can be efficiently implemented on FPGA

without memory restrictions. The entire 3x3 area of the window is used to calculate the output value and the origin pixel is replaced by this value. In median filtering, the median of the 9 values within the window is found out and applied to the origin pixel. This window is scanned over each and every pixel of the image until all the pixels have been 'filtered'.

The simulation in this paper uses the color photograph corrupted by salt and pepper noise of 0.02dB. The rasterized pixel values within the window operator are passed onto VHDL routine. The routine uses bubble sort technique to arrange the 9 pixels in ascending order. The median value, obtained from the middlemost element is the output pixel. In this manner all the pixels scanned and filtered by moving the window with each scan. The output image thus obtained becomes devoid of noise.



Fig. 5. Results from HDL simulation of median filtering.



Fig. 6. Results from MATLAB simulation of median filtering.

*C. Smoothing Linear Filters*

Smoothing filters are mainly used for blurring and noise reduction [4]. The output of a smoothing, linear spatial filter is the average of the pixels contained in the neighborhood of the filter mask. This process results in an image with reduced sharp transitions in color intensity levels.

As explained in the previous sections, the windowed pixels are transferred to the VHDL routine. The routine finds the average of the 9 input pixel values in the window by multiplying every pixel by 0.1111 and adding all the results. This average value then replaces the center pixel given at the output port of the routine.



Fig. 7. Results from HDL simulation of averaging filter.



Fig. 8. Results from MATLAB simulation of averaging filter

*D. Edge Detection*

Edge detection is a common approach to detect meaningful discontinuities in gray level which can be extended to color intensity levels [4]. In an image, for a point to be classified as an edge point, its 2dimensional first order derivative should be greater than a specified threshold. A set of such points that are connected according to a predefined criterion is defined as an edge. First order derivatives of a digital image are based on various approximations of the 2D gradient. This gradient vector is formed by differentiation of the image information in the two spatial directions. This gradient points in the direction of maximum rate of change at a particular point. In case of Sobel Edge Detection technique, the derivative in the X-direction is approximated to difference between the third and first row and the derivative in the Y-direction is approximated to difference between the first and third column.

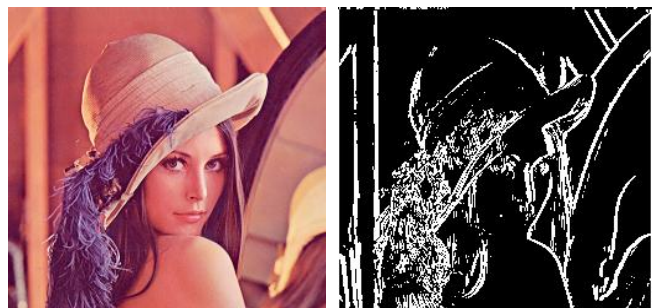


Fig. 9. Results from HDL simulation of Sobel edge detection mask





Fig. 10. Results from MATLAB simulation of Sobel edge detection technique

In this simulation, a similar 3-by-3 window is implemented and a Sobel mask is applied to the input pixel values obtained by the link. The horizontal and vertical derivatives are obtained and the gradient vector is calculated. By comparing the magnitude of this gradient vector with a particular threshold value, the output pixel is determined.

*E. MSE and PSNR values*

TABLE I  
COMPARISON OF MSE AND PSNR VALUES

NAME	MSE	PSNR
Pseudocoloring	1.3366	46.8708
Median Filtering	0.0258	64.0095
Smoothing Linear Filters	0.0012	77.2647
Sobel Edge Detection	0.9613	48.3021

III. CONCLUSION AND FUTURE WORK

This paper has demonstrated the simulation of four algorithms for color image processing on FPGA. A comparison of results obtained from VHDL and MATLAB has been made by evaluating Mean Square Error and Peak Signal to Noise Ratio. Future work would be to extend the capability of this system to process video data in a continuous stream obtained frame by frame. Also a hardware implementation of the above simulated routines would be the next course of action.

IV. ACKNOWLEDGMENT

The authors gratefully acknowledge the facilities provided in VLSI Laboratory of Electronics Engineering Department, Sardar Patel Institute of Technology, Mumbai.

V. REFERENCES

- [1] Anthony Edward Nelson, "Implementation of Image Processing Algorithms on FPGA Hardware," May 2000, Vanderbilt University, Nashville, TN.
- [2] J. Hammes, A. P. W. Bohm, C. Ross, M. Chawathe, B. Drapper, W. Najjar, "High Performance Image Processing on FPGAs," Feb. 2004, Computer Science Department, Colorado State University, Ft. Collins, CO, USA.
- [3] Bruce A. Draper, J. Ross Beveridge, A. P. Willem Bohm, Charles Ross and Monica Chawathe, 'Accelerated Image Processing on FPGAs,' IEEE Trans. Image Processing, vol.12 pp. 1543-1551, Dec. 2003
- [4] Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing," 2002, Pearson Education, Inc.

VI. BIOGRAPHIES



**Faheem Desai** (N'2007) was born in Mumbai in India on 1987. He is graduate student of Sardar Patel College of Engineering, Mumbai. He was the executive member of the Electronics Students Association and the chief editor of the annual departmental magazine. He is also a member of IEEE. His area of interest includes VLSI design, image processing and embedded systems.



**Rohit Bhat** (N'2007) was born in Mumbai in India on 1986. He is graduate student of Sardar Patel College of Engineering, Mumbai. He is a member of the Electronics Students Association. He is also a member of IEEE. His area of interest includes VLSI design, image processing and embedded systems.



**Harsh Shah** (N'2007) was born in Mumbai in India on 1987. He is graduate student of Sardar Patel College of Engineering, Mumbai. He is a member of the Electronics Students Association. He is also a member of IEEE. His area of interest includes VLSI design, image processing and embedded systems.



**S. S. Rathod** (M'2007) was born in Amravati in India on Feb 28, 1975. He graduated from the College of Engineering Badnera, and completed his post graduation in the field of Electronics Engineering from V.J.T.I., Mumbai-India. Currently he is pursuing his doctoral research at Indian Institute of Technology, Roorkee-India.

His employment experience includes eight years as an educationalist. He has published more than 20 papers in various national and international conferences. His special fields of interest include VLSI Design, process and device modeling. His name is listed in the science category of Marques Who's Who USA. He received outstanding achieving ward by the Energy Society, India. He is member of IEEE, ISTE and ISNT.



**P. V. Kasambe** (M'2007) was born in Yavatmal in India on Jun 26, 1976. He graduated from the College of Engineering Badnera, and completed his post graduation in the field of Electronics Engineering from SPCE(UA), Mumbai-India.

His employment experience includes eight years as an educationalist. His special fields of interest include VLSI Design, process control and instrumentation. He is member of IEEE, ISTE, ISA and ISNT.