*SindhUniv. Res. Jour. (Sci. Ser.) Vol. 48 (2) 435-440 (2016)*

**SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)**

# Model Based FPGA Design of Histogram Equalization

F. MEMON, F. JAMEEL*, M. ARIF**, F. A. MEMON***

Department of Electronic Engineering, Mehran University of Engineering & Technology, Jamshoro, Sindh, Pakistan

Received 8th October2015 and Revised 16th March 2016

**Abstract:** Histogram equalization is a preliminary process for image processing and enhancement, with a key focus on hardware implementations, real-time applications and fast performance. In this paper, a novel approach using Model Based Design for hardware implementation of histogram equalization algorithm into Field Programmable Gate Array (FPGA) is presented. The proposed methodology assists to develop and prototype the design in a comparatively short time by removing time consuming and tedious work due to manual coding. With the Model Based Design, MATLAB model of histogram equalization is translated into FPGA hardware implementation via HDL Coder. The design is implemented on Xilinx Virtex5 FPGA device and can work with an estimated frequency of 183.733MHz by occupying less than 5% of available hardware with total power consumption of 0.367W. Experimental results suggest that the proposed design methodology provides an equalized image comparable in accuracy to the full precision MATLAB's output, and generates output image of size 494 x 335 pixels in 0.9057msec vs 10.845sec in MATLAB, that is much faster than real time video rate.

**Keywords:** Image Enhancement, Histogram Equalization, HDL Coder, FPGA.

## 1. INTRODUCTION

Image enhancement techniques have become basic requirement for present day applications that require images as their principle source of interpretation and analysis. There are various ways to enhance the contrast of an image. One of effective and well established method is the histogram equalization. A histogram is bar graph representation of the frequency of occurrence of brightness values in an image. Histogram equalization creates an image with more equally distributed brightness levels for achieving higher contrast. The application of histogram equalization has been found in different fields including medical imaging (Sathyam-urthy, *et al.,* 2013, Selvaraj, 2013) radiology (Salcic, 1999), and speech recognition (Torre, *et al.,* 2005) etc.

In literature, most of the image enhancement techniques are found to be implemented using MATLAB and C/C++. It is the easier way and enough for certain cases, but this limits in a severe way the achievable performance and efficiency of the design. A specialized hardware design, however, can optimize much more performance and gets better results with a fraction of the processing power.Reconfigurable hardware in the form of FPGA is considered as a practical way of achieving high performance for computationally intensive image processing algorithms (Drapper*et al.,*2003). FPGAs are configured using Hardware Description Languages (HDL) Verilog and Very High Speed Integrated Circuits (VHSIC) Hardware Description Language (VHDL) (Johnston, *et al.,* 2004). Typical HDL design method employs hand-written HDL which is tedious and time consuming (MathWork's, 2014).

In order to shorten the design time and enables to optimize design to meet predefined performance criteria, this research deals with Model Based FPGA implementation of a histogram equalization algorithm using HDL Coder tool.

The HDL Coder tool (MathWork's, 2014) enables the designer to focus on the design in system level, instead of struggling in the details of programming. However there is still a long learning curve to be familiar with these new tools. MATLAB/Simulink is an environment for model-based design, which covers almost all industrial and scientific areas. It is worthwhile to bridge over MATLAB/Simulink and FPGA design. In this paper, the Model Based Histogram Equalization system is designed and implemented using FPGA and validate its practical simulation using ModelSim.

## 2. RELATED WORK

In literature, different approaches and techniques have been studied on image enhancement with an FPGA implementation perspective. A brief review of related work is given below.

(Tsutsui, *et al.,* 2010)develop FPGA based real-time video image enhancement system.The proposed system uses variational model of the retinex theory and can process color picture of size 1900 × 1200 at 60 frames per sec. (Alsuwailem, *et al.,* 2010) present a nonconventional schemes for real time histogram equalization using FPGAs.They develop the efficient architectures for histogram equalization, but the resultant images obtained by their approach are generally not adequate.

++Corresponding author: farida.memon@faculty.muet.edu.pk
*Department of Computer Science SZABIST, Karachi
**Department of Biomedical Engineering, Mehran University of Engineering and Technology, Jamshoro, Sindh, Pakistan
***Department of Computer System Engineering, Quaid-e-Awam University of Engineering Science and Technology, Nawabshah, Sindh, Pakistan

(Xiying *et al.,*1998) present a real-time histogram equalization implementation with high processing speed for calculation completion and generation of a lookup table result. In this work, authors only study small image sizes and do not consider potential modifications to the architecture, required for large images which may necessitate long processing times and cannot be placed on-chip for convenience. (Sanny *et al.,* 2014 ) develop an energy-efficient histogram equalization architecture and propose a memory activation schedule to minimize energy consumption. For larger image sizes, they design an efficient buffering and power-down scheme to reduce external Dynamic Random Access Memory (DRAM) power computation. Pipelining and data hazard prevention are employed to achieve a realistic frame rate of 30+ frames per second.(Hanumantharaju *et al.,* 2013) present the development of a new algorithm for Gaussian based color image enhancement system. The proposed algorithm not only provides better dynamic range compression and color rendition effect but also achieves color constancy in an image. The design exploits high degrees of pipelining and parallel processing to achieve real time performance.

(Chandrashekar *et al.,* 2009) presentthe FPGA implementation for automatic enhancement of infrared images using Successive Mean Quantization Transform (SMQT). The SMQT performs a nonlinear stretch and is found to preserve the basic shape of the histogram. (Sowmya, 2011) address the FPGA implementation of different image enhancement techniques namely histogram equalization, brightness control and contrast stretching. The algorithms are implemented onxc2vp30 target FPGA.The hardware implementation results suggest the improvement in speed performance for enhancement techniques.(Sachdeva, 2010) present the FPGA based real time histogram equalization to enhance the vision quality of image. The proposed system makes use of counters and switching decoder,implemented on Altera's FPGA Stratix II family chip for computing both histogram statistics and equalization. (Wang, *et al.*,2006) develop a FPGA based self-adaptive enhancement system for infrared images using plateau histogram equalization. The implementation of proposed system is done using EPIK100QC208 FPGA and is capable of contrast enhancement of infrared images greatly.

Although the design space of the histogram equalization architecture is extensively explored, all of the abovementioned works have used conventional approaches for hardware implementation. The digital designs on FPGAs are implemented by using hardware description languages, which require prior knowledge and experience in the tools such as VHDL and Verilog. Present work addresses an optimal FPGA design of histogram equalization based on Model Based Design. In this research, it has been shown that the Model Based FPGA system using HDL Coder can be efficiently designed using Xilinx Vertix5 FPGA device and VHDL together.

## 3. HISTOGRAM EQUALIZATION

Among various image enhancement techniques, histogram equalization is well known for its simplicity and low cost (Kaur, *et al.,* 2013, Yeganeh, *et al.,* 2008) Histogram equalization adjusts the intensity values of the image so that the resultant histogram approximately matches the uniform histogram. Through this adjustment, the intensities can be better spreadout on the histogram and lower local contrast areas of image are allowed to acquire a higher contrast. The method is appropriate for the images having both bright or dark backgrounds and foregrounds.

In histogram equalizationusing transform function T, input pixel intensity x istransformed to new intensity value x′. The transform function T, is the product of a cumulative histogram and a scale factor. The scale factor is required to fit the new intensity value within the range of the intensity values, for example, 0 to 255 in case of gray level image.

$$\mathbf{x'} = \mathbf{T(x)} = \sum_{\mathbf{i=0}}^{\mathbf{x}} \mathbf{n_i} \cdot \frac{\mathbf{max.intensity}}{\mathbf{N}} \qquad (1)$$

where$n_i$ is the number of pixels at intensity i, N is the total number of the pixel in image. The functional diagram of the proposed histogram equalization algorithm is shown in **(Fig-1)**.

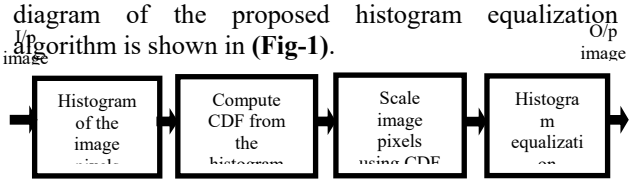



**Fig. 1: Functional block diagram of proposed method**

We divideproposed histogram equalization algorithm into three steps which are shown in **(Fig.1)**. Step1 takes the input image and computes a histogram of the image pixels. Step 2 calculates the Cumulative Distribution Function (CDF) based on the created histogram. Finally, Step 3 scales the image pixels using the CDF and creates the equalized image. This is the final output image.

## 4. MODEL BASED DESIGN OF HISTOGRAM EQUALIZATION

Currently, Model Based Design is a familiar hardware approach with a broad collection of specialized software tools such as MATLAB/Simulink (Smith, *et al.*,2007). This design methodology presents a common platform for the integration of various stages of the development process (Grover, 2014). Moreover, Model Based Design automates HDL code generation and verification with MATLAB/Simulink and optimizes

the models to achieve speed, area and power constraints. Therefore, Model Based Design effectively works as a tool for rapid prototyping, system validation and testing. For the rapid design of histogram equalization, HDL Coder in the MATLAB/Simulink that supports automatic VHDL code generation for FPGA implementation is used. In addition, HDL Verifier tool integrated within HDL Coder verifies the design and continuously tests the designthrough co-simulation using ModelSim software. Thus, Model Based Design flow using the HDL Coderallows to pass directly from the histogram equalization algorithm to an FPGA implementation as illustrated in **(Fig.2)**.

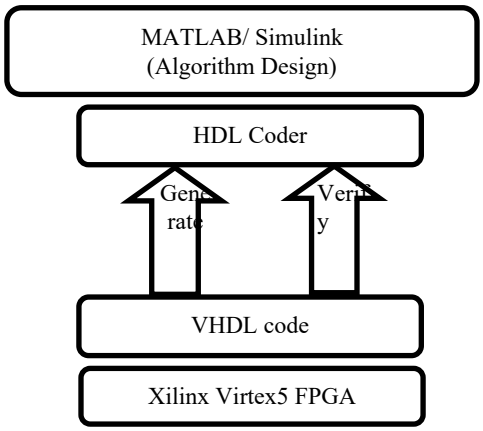



**Fig. 2: Model Based Design of histogram equalization using HDL Coder**

In implementing the MATLAB model of histogram equalization for FPGA hardware, the MATLAB algorithm of design is modified to meet the requirements of the HDL Coder. Initially, histogram equalization model is first designed and verified in MATLAB using floating-point data representation. However, to reduce hardware resources, conversion into a fixed-point data is done during the process of VHDL code generation. Moreover, a word-length and fraction-length optimization of the fixed-point data is one of critical phase of algorithm implementation on a FPGA. To achieve fixed-point model of histogram equalization which corresponds to floating-point, WL (Word Length) and FL (Fraction Length) values of 16 and 8 are chosen from fixed point optimization process using HDL Workflow Advisor of the HDL Coder.

Once the conversion process is completed and MATLAB code of design is successfully converted into hardware design, generated VHDL code of histogram equalization is verified through co-simulation using ModelSim 10.3d software. Further design is processed in Xilinx ISE 14.2 Design Suite for synthesis and implementation on Xilinx Virtex5 xc5vlx50t-ff1136 FPGA device.

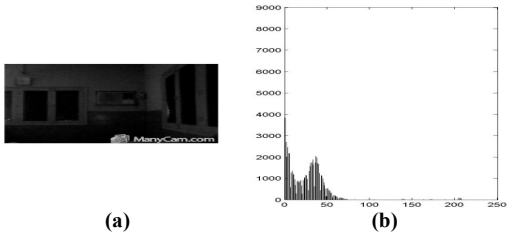
## 5. RESULTS AND DISCUSSIONS

Elaborate experiments were conducted on over a dozen varieties of images and consistently good results have been obtained for the proposed algorithm. As examples, three poor quality 8-bit gray scale images of *house* of size 494×335 pixels, *lena*and *cameraman* of sizes 256×256 pixels respectively have been enhanced using the proposed hardware approach.

The proposed FPGA implementation of histogram equalization algorithm is initiated in MATLAB R2013a first in order to confirm the correct working of the algorithm. We used an 8-bit grayscale image of *house* of size 494×335 as a test image. **(Fig.3aand b)**displays the original test image and corresponding histogram of the image. The original image has to be passed through the histogram equalization for enhancing the image in terms of brightness and contrast. **(Fig.4aandb)** illustrates the histogram equalized image along with histogram of equalized image, which is enhanced image in terms of improved pixels values compared to the previous blur original image. Running on i7-4500U CPU @ 1.80GHZ, MATLAB needs 10.845sec to get the equalized image.

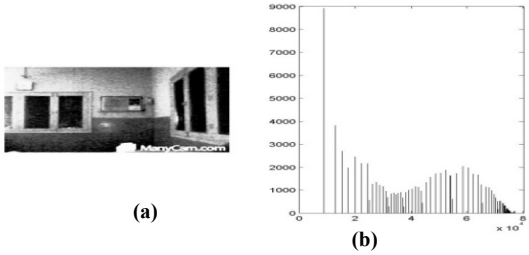
Subsequently, the MATLAB algorithm of design is converted to VHDL using HDL Coder so that it may be implemented on FPGA. To validate the hardware design, the system simulation is performed using Model Sim 10.3d software.**(Fig. 5a, b)** represents the VHDL simulation of design. Simulation results demonstrate that the equalized image produced the VHDL corresponds to the full precision MATLAB's output.

Simultaneously image enhancement is performed for *lena* and *cameraman* test images using proposed approach and corresponding results are displayed through **(Fig. 6 (aandb) to Fig. 11 (a andb))** and similar observations can be made for both.



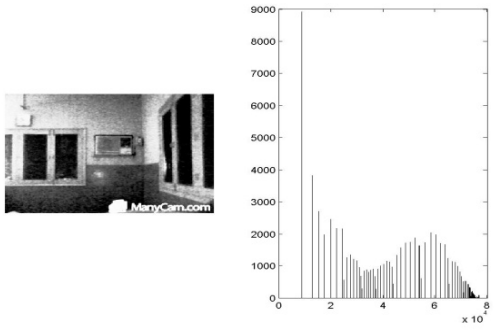

**(a)** **(b)**

**Fig. 3: (a) original house image; (b) original house histogram**



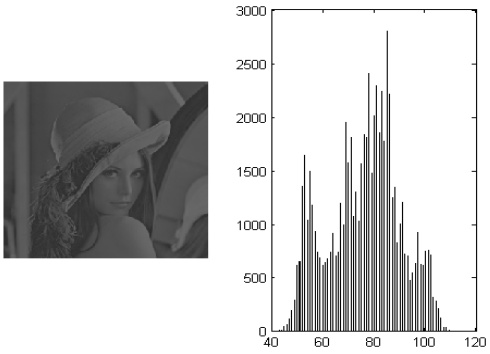

**(a)** **(b)**

**Fig. 4: MATLAB simulation (a) equalized house image (b) equalized house histogram**



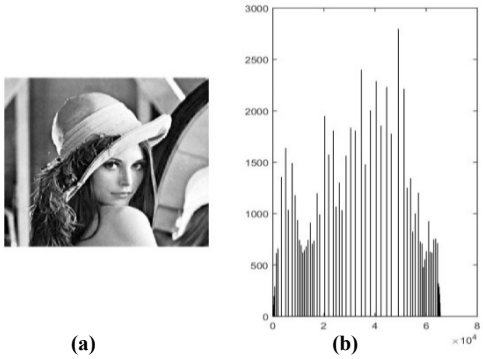
(a) (b)

**Fig. 5: VHDL simulation (a) equalized house image (b) equalized house histogram**



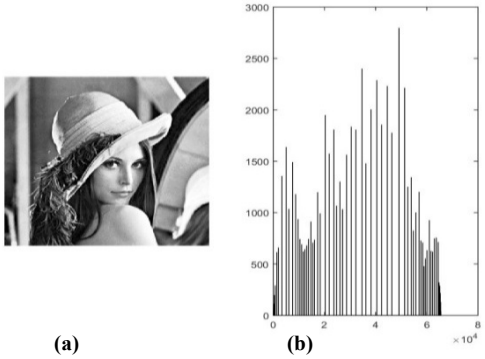
(a) (b)

**Fig. 6: (a) original lena image; (b) original lena histogram**



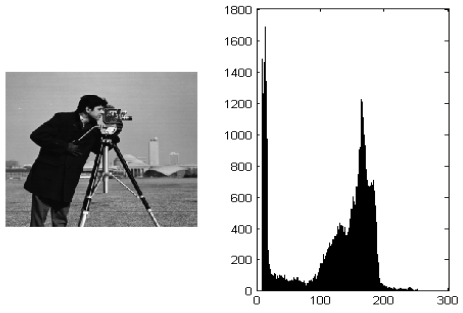
(a) (b)

**Fig. 7: MATLAB simulation (a) equalized lenaimage (b) equalized lena histogram**



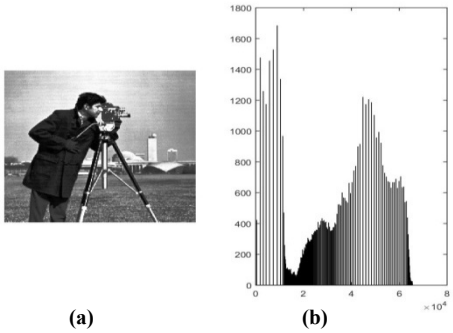
(a) (b)

**Fig. 8: VHDL simulation (a) equalized lenaimage (b) equalized lena histogram**



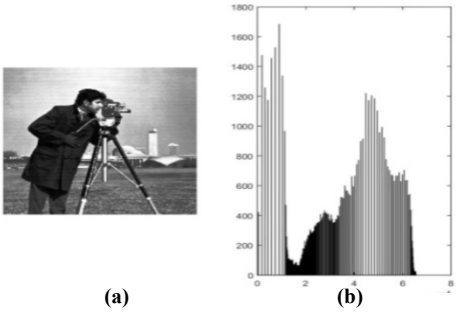
(a) (b)

**Fig. 9: (a) original cameraman image; (b) original cameraman histogram**



(a) (b)

**Fig. 10: MATLAB simulation (a) equalized cameraman image (b) equalized cameraman histogram**



(a) (b)

**Fig. 11: VHDL simulation (a) equalized cameraman image (b) equalized cameraman histogram**

### 5.1 Comparison of VHDL and MATLAB algorithms

To analysis the results of the VHDL simulation of the histogram equalization algorithm, a comparison between hardware (e.g. VHDL) algorithm and software (e.g. MATLAB) algorithm is performed. This is exciting because it allows to verify a hardware algorithm's accuracy and decides whether or not to implement design tradeoffs based on output validity. In **(Fig.4 a and b)** and **(Fig. 5 a and b)** MATLAB and VHDL simulations of a histogram equalization algorithm for a *house* test image are presented respectively. The corresponding error plot between MATLAB and VHDL algorithm for equalized image is given in **(Fig.12)**. Whereas the black resultant image of error plot represents the zero intensity value for every pixel in the image. Thus, from the plot, it is obvious that the two algorithms are identical.

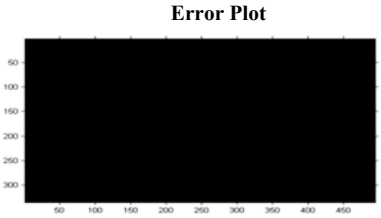



**Fig. 12: VHDL and MATLAB Comparison**

## 5.2 Implementation Results

The proposed design of histogram equalization algorithm for *house* test image is implemented on Virtex5 xc5vlx50t-ff1136FPGA target device using Xilinx ISE Design Suite 14.2. The logic resourcesutilized by the design with timing performance are shown in **(Table 1 and 2)** respectively. Table 1 represents the total number of slices and look up tables used in this design, which indicates total area occupied in the target device. From the Table 1 and 2, it is found that the proposed design is working with an estimated speed of 183.733MHzMHz by utilizing only 195 slices. Proposed model is using 307 look up tables, 493 slices registersand1 global clock buffer to implement theentire logic which is less than 5% of available hardware on FPGA. Hardware consumption in any deign determines its cost.Therefore, the cost of proposed design is decreaseddue to lesser hardware utilization. Hence, the suggesteddesign methodology improves efficiency in area and provides good choice in terms of low cost hardware.

**Table 1: Resource Utilization**

| Logic utilization | Used | Available | Utilization |
|---|---|---|---|
| Number of Slice Registers | 493 | 28,800 | 1% |
| Number of Slice LUTs | 307 | 28,800 | 1% |
| Number of occupied Slices | 195 | 7,200 | 2% |
| Number of BUFGs | 1 | 32 | 3% |

**Table 2: Timing Summary**

| Parameter | Value |
|---|---|
| Minimum Period | 5.443ns |
| Maximum Frequency | 183.733MHz |

The power analysis of the implemented design of histogram equalization is performed using the Xilinx XPower Analyzer (XPA) tool. The total power consumption of design is relatively low to 0.367W at 25.6 degree C as shown in **(Table 3)**.

**Table 3: Power Consumption**

| Parameter | Value |
|---|---|
| Total Power | 0.367 (W) |
| Junction Temperature | 25.6 (degree C) |

**(Table 4)** shows the comparison of implementation results of the proposed work with that of existing works in the literature. Although it is difficult to make comparison since the approach used for the algorithm is changed and also hardware implementation technique and the device on which implementation is done is different. It can be noted that the proposed histogram equalization algorithm requires less area and time for equalizing the input image and hence is suitable for real time applications.

**Table: 4 Comparison of Implementation Results of the Proposed work with the Existing Works**

| Works | (M. Chandrashekar *et. al* 2009) | (Wang Bing-jian *et. al* 2006) | Proposed |
|---|---|---|---|
| Implementation | Infrared Image Enhancement using Successive Mean Quantization Transform | Plateau Histogram Equalization | Model Based Design of Histogram Equalization |
| Device | ACTEL FPGA - APA 600 | EPIKl00QC208-Altera Acexlk | Xilinx Virtex5 xc5vlx50t |
| Image size (pixel) | 64× 64 | 128×128 | 494 × 335 |
| Area | I/O cells 34 of 454(7%) Block RAMs 48 of 56(86%) Core Cells 2123 of 21504 ( 10% ) | I/O cells 25 of 160(40%) 32 × l RAMs 256 | I/O cells 82 0f 480 (17%) Block RAMs 16 0f 60 (26%) Slice Registers 493 of 28,800 (1%) Occupied Slices 195 of 7200 (2%) |
| Time Period (ms) | * | 1.018 | 0.9057 |
| Power (W ) | * | * | 0.367 |

*Data not available.

## 6. CONCLUSION

In this paper, FPGA-based hardware implementation flow for the histogram equalization is successfully done on the xc5vlx50t-1ff1136 Xilinx Virtex5 FPGA using the HDL Coder. Functionality of implemented design is verified through co-simulation using ModelSim 10.3d software. Simulation results indicate that VHDL simulations match with the MATLAB simulations and confirm the efficiency of presented methodology. The design implemented on target device can work with maximum frequency of 183.733MHz and utilizes less than 5% of available hardware resources to produce equalized image of a size 494 × 335 pixels in 0.9057msec allowing real time processing. Power analysis is also done using Xilinx XPA tool.It is demonstrated that by adopting Model Based approach the modeling and VHDL generation time of histogram equalization is reduced. Moreover, generated VHDL code can be easily verified and mapped into FPGA allowing the rapid prototyping of design.

## 7. ACKNOWLEDGEMENT

The authors wish to acknowledge Mehran University of Engineering and Technology, Jamshoro, Pakistan, for providing the technical facilities to carry out this research work.

**REFERENCES:**

Alsuwailem, A., and S. Alshebeili, (2005) "A New Approach for Real-time Histogram Equalization using FPGA", IEEE Proceedings of International Symposium on Intelligent Signal Processing and Communication Systems, 397-400, December 13-16, 2005 Hong Kong.

Chandrashekar, M., U. Naresh, K. Sudarshan, and K. Nagabhushan, (2009) "FPGA Implementation of High speed Infrared Image Enhancement," International Journal of Electronic Engineering Research, vol.1, no. 3, 279-285.

Drapper, B. A., J. R. Beveridge, A. P. W. Bohm, and C. Ross, (2003) "Accelerated Image Processing on FPGAs", IEEE Transactions on Image Processing, vol.12, no.12, 1543- 1551.

Grover, N., M. and K. Soni, (2014) "Simulation and Optimization of VHDL code for FPGA-Based Design using Simulink", International Journal of Information Engineering and Electronic Business, vol. 3, 22-27.

Hanumantharaju M. C., M. R. avishankar, and D. R. Rameshbabu, (2013). "Design of Novel Algorithm and Architecture for Gaussian Based Color Image Enhancement System for Real Time Applications", Proceedings of 3rdSpringer International Conference on Advances in Computing, Communication, and Control, 595-608, January 18-19, Mumbai, India.

Johnston, C. T., K. T. Gribbon, and D. G. Bailey, (2004) "Implementing Image Processing Algorithms on FPGAs", Proceedings of the Eleventh Electronics New Zealand Conference, Palmerston North, New Zealand, 118-123.

Kaur, M., K. Jain, and V. Lather, (2013) "Study of Image Enhancement Techniques: A Review", International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, no. 4, 846-848.

Sachdeva, N., and T. Sachdeva, (2010) "An FPGA based Real-time Histogram Equalization Circuit for Image Enhancement", International Journal of Electronics and Communication Technology, vol.1, no. 1, 63-67.

Salcic, Z., and J. Sivaswamy, (1999) "IMECO: A Reconfigurable FPGA-based Image Enhancement Co-Processor Framework," Real-Time Imaging, vol.5, no. 6, 385-395.

Sanny, A., Y.H.E. Yang, and V. K. Prasanna, (2014)" Energy-efficient histogram equalization on FPGA", Proceedings of IEEE Conference on High Performance Extreme Computing (HPEC), 1-6, 9-11 , Waltham, MA, September 2014.

Sathyamurthy, S., and C. Lakshmi, (2013) "Adaptive Histogram Equalization for Detecting Cancer in Digital Mammogram", International Journal of Science and Research (IJSR), vol. 2, no. 5.2.

Selvaraj, D., and R. Dhanasekaran, (2013) "MRI Brain Tumour Detection By Histogram And Segmentation By Modified GVF Model", International Journal of Electronics and Communication Engineering & Technology (IJECET), vol.4, no.1, 55-68.

Simulink HDL Coder™ user's guide, (2014) The MathWorks ,Inc. http://cn.mathworks.com/help/pdf_doc/hdlcoder/hdlcoder_ug.pdf.

Smith, P., S. Prabhu, and J. Friedman, (2007) "Best Practices for Establishing a Model-Based Design Culture", Mathworks Inc., SAE Technical Paper 2007-01-0777, 2007, doi:10.4271/2007-01-0777.

Sowmya, S., and R. Paily, (2011) "FPGA Implementation of Image Enhancement Algorithm", IEEE International Conference on Communications and Signal Processing, 584-588,10-12 Calicut.

Torre, A., A. M. Peinado, J. C. Segura, and J. L. Perez-Cordoba, (2005) "Histogram equalization of speech Representation for robust speech recognition". IEEE Transaction on Speech and Audio Processing, vol.13, no.3, 355-366.

Tsutsui, H., H. Nakamura, R. H. Hashimoto, Okuhata, and T. Onoye, (2010) "An FPGA Implementation of Real-time retinex video image enhancement", IEEE Transaction on World Automation Congress, 1-6,19-23 September, 2010,Kobe.

Wang, B., S. Liu, Q. Li, and H. Zhou, (2006) "A real time contrast enhancement algorithm for infrared images based on plateau histogram", Infrared Physics and Technology, Elsevier, vol. 48, no. 1, 77- 82.

Xiying Li, GuoQiang Ni, Yanmei Cui, TianPu, and YanliZhong., (1998) "Realtime image histogram equalization using FPGA ", Proceedings of SPIE 3561 Conference on Electronic Imaging and Multimedia Systems II, 293-299, August 19, 1998, China.

Yeganeh, H., A. Ziaei, and A. Rezaie, (2008) "A Novel Approach for Contrast Enhancement Based on Histogram Equalization", Proceedings of the International Conference on Computer and Communication Engineering, 256-260, 13-15 May 2008, Kuala Lumpur.