International Journal of Electronic Engineering Research
ISSN 0975 - 6450 Volume 1 Number 3 (2009) pp. 279–285

http://www.ripublication.com/ijeer.htm

FPGA Implementation of High Speed Infrared Image Enhancement

[1]M. Chandrashekar, U. Naresh Kumar[2], K. Sudershan Reddy[3] and [4]K. Nagabhushan Raju

[1]Design and Engineerin Division, Bharat Dynamics Limited, Hyderabad, India
Email: mcsbdl@hotmail.com
[2,3]Research Centre Imarat, Hyderabad, India
[4]Dept. of Electronics, SK University, Ananthpur, A.P., INDIA

Abstract

This paper deals with Field Programmable Gate Array (FPGA) based hardware Implementation of Infrared Image (IRI) enhancement of thermo graphic images. The image enhancement capabilities and properties of the transform are analyzed. The transform is capable to perform both a nonlinear and a shape preserving stretch of the image histogram. FPGA Implemented results compared with Matlab Experiments and comparisons to histogram equalization are conducted.

Introduction

Enhancing digital image to extract true image is a desired goal in several applications. Such transformation is known as image enhancement. Performing the task automatically without human intervention is particularly hard in image processing. Different approaches and techniques have been suggested to solve this problem [1-5]. One well established method is the histogram equalization [3]. Histogram equalization automatically flattens and stretches the dynamic range of the histogram of the image. Hence, an enhancement of the contrast in the image is achieved.

The Successive Mean Quantization Transform (SMQT) has properties that reveal the underlying structure in data. The transform performs an automatic structural breakdown of information. This can be interpreted as a progressive focus on details in an image. These characteristics make the transform interesting for automatic enhancement of any image.

This paper deals with H/w implementation SMQT is applied for automatic image enhancement. An adjustment parameter is introduced to further control the enhancement. The image enhancement results are compared to histogram equalization.

Description of the SMQT

Let *x* be a pixel and the intensity of a pixel will be denoted V(*x*). The SMQT has only one parameter input, the level *L* (indirectly it will also have the number of pixels *D* as an important input. The output pixel set from the transform is denoted *M* (*x*). The transform of level *L* from *D(x)* to *M(x)* is denoted

$$\mathrm{SMQT}_L : D(x) \rightarrow M(x) \tag{1}$$

The SMQT_L function can be described by a binary tree where the vertices are Mean Quantization Units (MQUs). A MQU consists of three steps, a mean calculation, a quantization and a split of the input set. The first step of the MQU finds the mean value of the pixels, denoted V(*x*), second, mean quantization of pixel set. The third step splits the input set into two subsets

$D_0(x) = \{x \mid V(x) \leq V(x), Vx \in D\}$

$D_1(x) = \{x \mid V(x) > V(x), Vx\ D\}$

where $D_0(x)$ propagates left and $D_1(x)$ right in the binary tree, see Fig. 1.

U(x) can be interpreted as the structure of ***D(x)***.

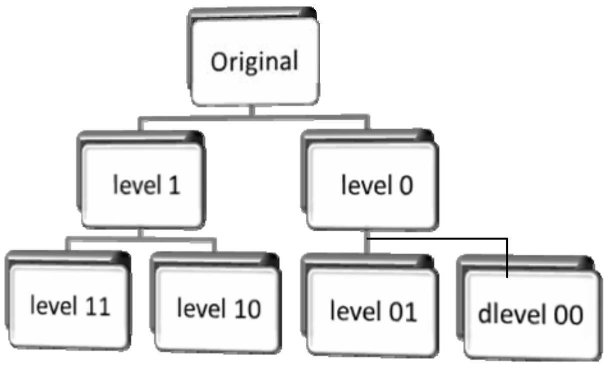

Figure 1: 2 level operation of one Mean Quantization Unit (MQU). Example.

The first level transform, SMQT_1, is based on the output from a single MQU, where ***U*** is the output set at the rootnode. The outputs in the binary tree need extended notation. Let the output set from one MQU in the tree be denoted ***U(l,n)*** where $l = 1,2,..., L$ is the current level and n = 1,2,..., $2^{(l-1)}$ is the output number for the MQU at level ***l***, Weighting of the values of the pixels in the ***U(l,n)*** sets are performed and the final SMQT_L is found by adding the results. The weighting is performed by $\boldsymbol{2^{L-l}}$ at each level ***l***.

Today digital imaging devices typically use the range 0 ... 255, that is 8 bits is used. For automatic image enhancement of 8 bits images ***L*** is chosen to 8. Nevertheless, it could be convenient to control the amount of enhancement applied. Given the original pixel set *D(x)* and the SMQT_8 enhanced pixel set ***M(x).***

Infrared Image Enhancement

A straight forward way to enhance an image is to use the SMQT directly. The only parameter to adjust is the level

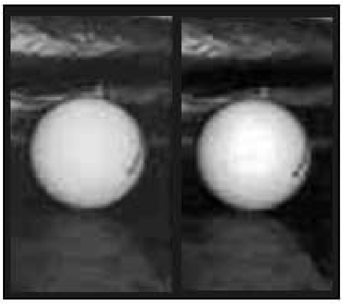

Figure 2: (Left) Original image. (Right) $SMQT_8$ enhanced image.

In the original image histogram it is possible to see that this image does not take advantage of the full dynamic range.

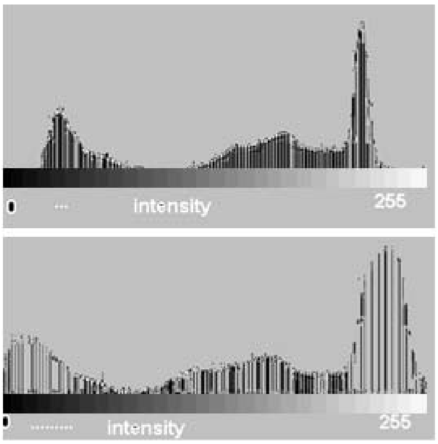

Figure 3: (UP) Original image histogram. (DOWN) $SMQT_8$ enhanced image histogram. **Intensity vs Percentage of Gray Values.**

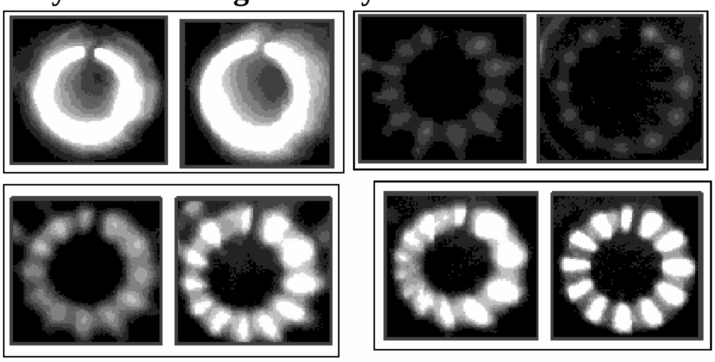

Figure 4: Image enhancement, original image and $SMQT_8$ of image.

Hardware Implementation Image Enhancement Successive Mean Quantization Transform (SMQT)

A. Design Assumptions

The design is based on the following assumptions

(1) Image enhancement using successive mean quantization transform (SMQT) core implementation on FPGA.
(2) The Image data assumed that image available in SRAM memory (FPGA – Block RAM) and image size is 64 x 64.
(3) The FPGA- Block RAM image data loaded by external processor and it generates start command to SMQT core.
(4) The enhanced image data should be available in memory and accessed by external Co-processor.
(5) External clock frequency 50 MHz
(6) The enhancement core implementation in two ways
    (a) Using Division (restoring) algorithm
    (b) Using multiplier logic.

SMQT core with Co-Processor

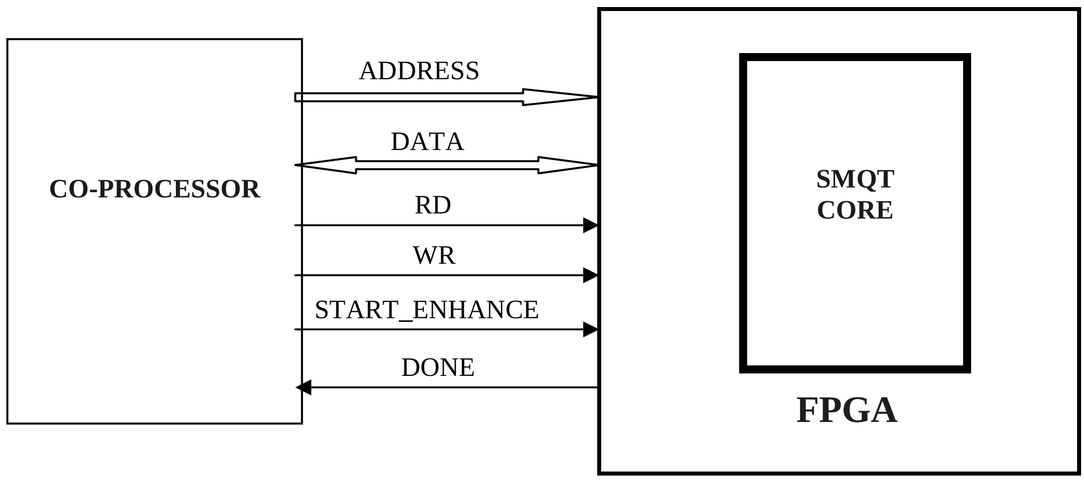

Figure 5: Interface of SMQT core with Co-processor.

Realization of Hard Core SMQT

The image data come from external world to FPGA and resides in Block memory of FPGA .The memory data is taken and calculated mean for each iteration. This SMQT algorithm has 8 levels for gray scale image. In each level it has 2 power (L-1) iterations and for each iteration mean is calculated and quantized. The Fig1.2 shows realization of SMQT core inside FPGA.

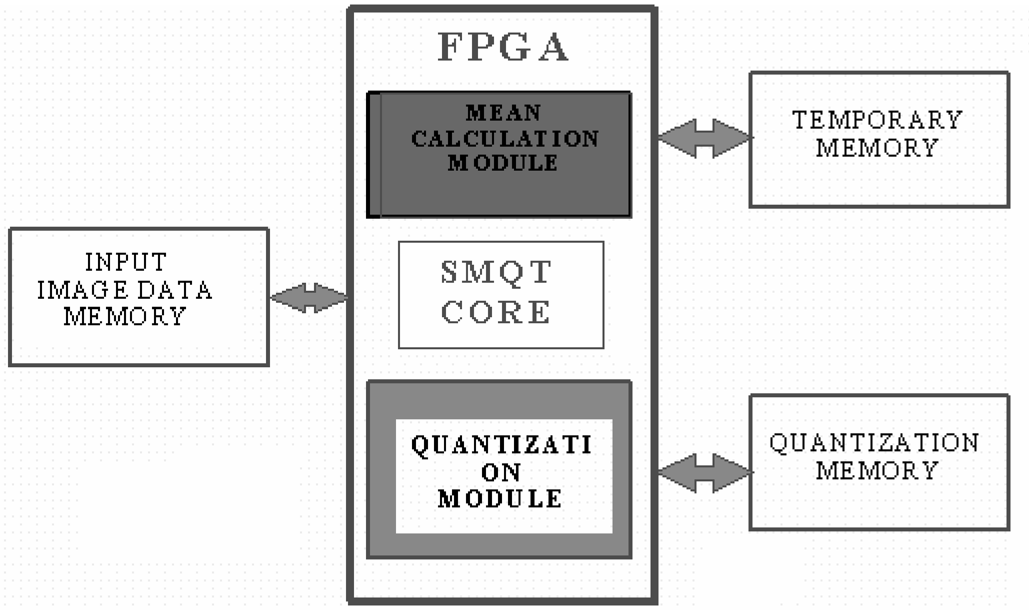

Figure 6: Block diagram of SMQT Hard core.

The input image memory of FPGA (4096x8) image data loaded by external microprocessor. Here, image data considered as size of 64 x 64.Once image data loaded into memory then FPGA waits for start enhancement command from microprocessor. In FPGA, temporary memory contents are also filled with zeros during initialization process. Whenever microprocessor generates start command to FPGA then mean calculation module reads temp memory contents as well as input image memory contents and compares both pixels and populates two sets for each iteration. In each iteration mean is calculated and registered in FPGA.Once mean is calculated then quantization for corresponding set pixels are updated in temp memory. Here no. of iterations depends on processing level. In the level1 only one iteration and similarly for level8 total 128 iterations takes place. The temporary memory contents are updated for each iteration based on mean value. In each level iterations completes then temporary memory contents are copied into quantization memory.

Mean calculation module

The mean is calculated from input image memory and temporary memory contents. The Fig1.3 shows mean calculation realization in hardware.

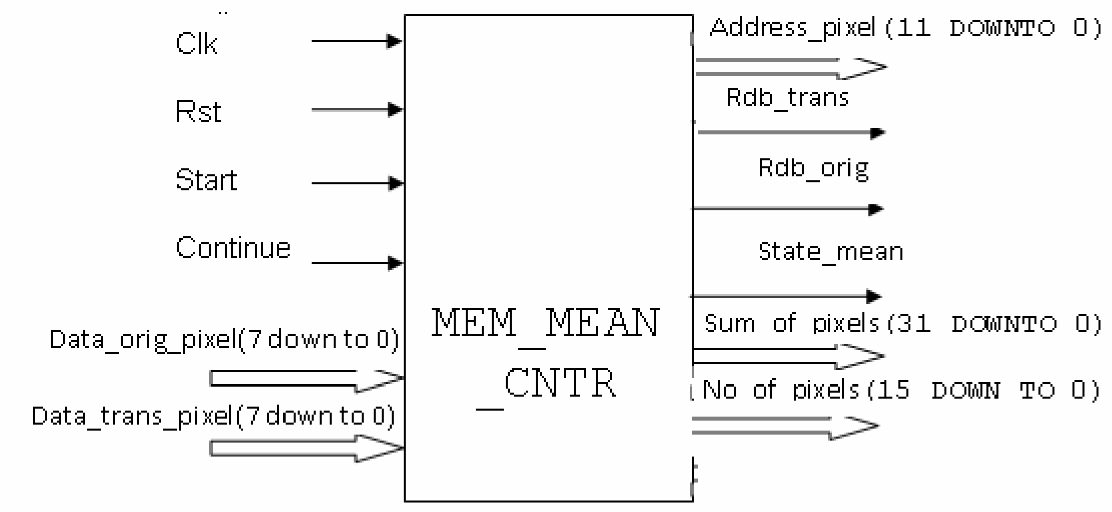

Figure 7: realization of mean calculation module.

Synthesis and Performance Estimation

The SMQT core implemented using VHDL and synthesized for ACTEL FPGA - APA 600.The core includes BRAM's for storing as well as quantization of Image data.

Core Cells	**:**	**2123**	**of**	**21504**	**(10%)**
IO Cells	**:**	**34**	**of**	**454**	**(7%)**
Block Rams	**:**	**48**	**of**	**56**	**(86%)**

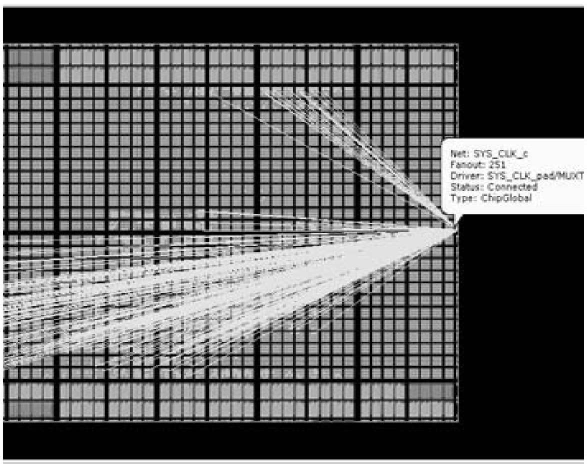

Figure 8: Pre-Routing of clock distribution over FPGA.

Mean calculation module
Mean calculation with multiplication for further processing
Start processing of SMQT core
Image write into FPGA by co-processor

Conclusions

FPGA implementation of the SMQT has been applied and analyzed for automatic enhancement of infrared images. Properties of the SMQT on images by means of histogram change have been investigated. The SMQT was found to retain the basic shape of the histogram and performs a nonlinear stretch. Hence, the SMQT is found to perform a balanced and natural enhancement of images. A comparison with histogram equalization has been performed, which showed the advantage of the SMQT based enhancement.

References

[1] C. Munteanu and A. Rosa, “Towards automatic image enhancement using genetic algorithms,” *Proceedings of the 2000 Congress on Evolutionary Computation*, vol. 2, pp. 1535–1542, July 2000.

[2] C. Munteanu and A. Rosa, “Gray-scale image enhancement as an automatic process driven by evolution,” *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 34, pp. 1292–1298, April 2004.

[3] William K. Pratt, *Digital Image Processing*, John Wiley & Sons, 3rd edition, 2001.

[4] Anil K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, 1989, ISBN 0-13-336165-9.

[5] Z. Rahman, D.J. Jobson, and G.A. Woodell, “Multi-scale retinex for color image