International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 3, Issue 1, January 2014

GLCM ARCHITECTURE FOR IMAGE EXTRACTION

M.Harsha vardhan[1] *Student (M.Tech), CS*
Department of ECE
Anil Neerukonda Institute of technology

S.Visweswara Rao[2] *Asst. Professor*
Department of ECE
Anil Neerukonda Institute of technology

Abstract

In this paper we are extracting the image features using different algorithms that are specified with architectural models with internal modules represented. The main objective involves calculating the different features for a given image. Digital image has several features where a feature is a characteristic that can capture a certain visual property of an image either globally for the whole image, or locally for objects or regions. A key function in different image applications is Feature extraction. There are different algorithms to extract texture features such as Structural, Statistical methods [3]. Feature extraction is a key function in various image processing applications. A feature is an image characteristic that can capture certain visual property of the image. Texture is an important feature of many image types, which is the pattern of information or arrangement of the structure found in a picture. Texture features are used in different applications such as image processing, remote sensing and content-based image retrieval. These features can be extracted in several ways. The most common way is using a Gray Level Co-occurrence Matrix (GLCM). GLCM contains the second-order statistical information of neighboring pixels of an image. Textural properties can be calculated from GLCM to understand the details about the image content. [6]. The Gray Level Co occurrence Matrix is a second order Statistical method. These features are implemented in VERILOG language. The tools used for implementation of the paper are XILINX ISE.

Index Terms: *Gray Level Co occurrence Matrix (GLCM), VLSI architecture, Block Matching Algorithm.*

1. INTRODUCTION

An image is defined as a two dimensional function, f(x; y), where x and y are spatial Coordinates and f (x; y) is a set of G grey-tones. When x, y and the grey-tones of f Have discrete quantities, the image is called a digital image.
The function f(x; y) is:

$$f(x;y)=\begin{bmatrix} f(0;0) & f(0;1) & \dots\dots\dots & f(0;Ny-1) \\ f(1;0) & f(1;1) & \dots\dots\dots & f(1;Ny-1) \\ \dots & \dots & \dots\dots & \dots \\ \dots & \dots & \dots\dots & \dots\dots\dots \\ f(Nx-1;0) & f(Nx-1;1) & \dots\dots & f(Nx-1;Ny-1) \end{bmatrix} \text{--eq 1}$$

A digital image has a finite number of elements; each of these elements has a particular value and location. These elements are called pixel or image elements.

In order to reduce the amount of data, an image is represented using a set of features. Feature extraction is a primitive type of pattern recognition and it is very important for pattern recognition. This step extracts some features such as Entropy, Angular Second Moment, Contrast, Maximum Absolute Deviation, and Mean. Features contain the relevant information of an image and will be used in image processing (e.g. searching, retrieval, storing).

Real-time image pattern recognition is a challenging task which involves image processing, feature extraction and pattern classification. It applies to a wide range of applications including multimedia, military and medical ones [4]. Its high computational requirements force systems to use very expensive clusters, custom VLSI designs or even both. These approaches suffer from various disadvantages, such as high cost and long development times.

Recent advances in fabrication technology allow the manufacturing of high density and high performance Field Programmable Gate Arrays (FPGAs) capable of performing many complex computations in parallel while hosted by conventional computer hardware. A variety of architecture designs capable of supporting real-time pattern recognition have been proposed in the recent literature, such as implementations of algorithms for image and video processing, classification and image feature extraction algorithms [6]. Most prominent approaches include the extraction of Gabor wavelet features for face/object recognition and the computation of mean and contrast Gray Level Co occurrence Matrix (GLCM) features.

In the second case the two features are approximated without computing GLCMs. The proposed architecture combines both software and hardware to raster

ISSN: 2278 – 909X

scan input images with sliding windows and produce feature vectors consisting of four GLCM features calculated for four directions.

2.1 Feature Extraction Algorithms:

The different methods for feature extraction are

1. Structural Method:

Structure represents a texture according to the local properties (micro-texture) and spatial organization (macro-texture) of local properties. The structural approaches provide a good symbolic description of the image, are useful for texture generation as well as texture analysis This method is not suitable for natural textures because of the variability both of micro texture and macro-texture and there is no clear distinction between them.

2. Statistical Method:

Statistical methods represent the texture indirectly according to the non-deterministic properties that manage the distributions and relationships between the gray levels of an image [2]. By computing local features at each point in the image, and deriving a set of the local features, statistical methods can be used to analyze the spatial distribution of gray values. Based on the number of pixels defining the local feature, statistical methods can be classified into first-order (one pixel), second-order (two pixels) and higher-order (three or more pixels) statistics.

The difference between these classes is that the first-order statistics estimate properties (e.g. average and variance) of individual pixel values by waiving the spatial interaction between image pixels, but in the second-order and higher-order statistics estimate properties of two or more pixel values occurring at specific locations relative to each other [5]. The most popular second-order statistical features for texture analysis are derived from the co-occurrence matrix.

Description of some of statistical methods is:

a. First order histogram:

An image is a function f(x; y) of two dimensions x and y, x = {0, 1 ...Nx-1} and y = {0, 1....Ny -1}. The f(x; y) can take discrete values i = {0, 1....Ng-1}, Ng is the total number of intensity levels (the number of pixels in the whole image) in the image [7]. A histogram is a diagram which shows how many pixels of an image have certain intensity. It has some advantages, one of them is that histograms of the image and its rotation image are the same, and another is that the size of storage place for histogram is lower than the storage size of the image

b. Second-order histogram:

In the second-order, the relationship between two pixels is considered. The co occurrence matrix is defined as a second-order histogram statistics and it is one of the best known texture analysis methods. This method is known Gray Level Co-occurrence Matrix (GLCM). It is usefulness in applications where the space distribution of gray levels is important (e.g. in radar signals), or in image analysis applications (e.g. biomedical), and also it is useful for remote sensing techniques that are an important in grasping damage information caused by earthquakes.

In the second-order, measures are used to consider the relationship between groups of two pixels (usually neighboring) in the image. It is assumed that an image is stored as a 2D array, f(x; y). The spatial domains of x and y are Lx = {1, 2....Nx} as a horizontal spatial domain, Ly = {1, 2...Ny} as a vertical spatial domain. The Lx * Ly is the set of individual pixels and the digital image I is a function that assigns a gray level value (brightness value) of G = {1, 2...Ng} to each pixels. The matrix defines the probability of joining two pixels Pd(i,j) that have values i and j, with distance d.

3. COOCCURENCE MATRIX

The co-occurrence matrix is a statistical model that is useful in a variety of image analysis applications, such as in biomedical, remote sensing, industrial defect detection systems, etc. FPGAs are reconfigurable hardware devices and have ability to execute many complex computations in parallel; these abilities enable a hardware system dedicated to performing fast co-occurrence matrix computations. The Very Large Scale Integration (VLSI) architectures could be considered as competitive options but they are not reconfigurable, and also have a high development cost and time consuming development process. The image feature extraction process involves raster scanning the image with windows (sub-images) of a given dimension and a given scanning step. This step corresponds to the offset between two consecutive sub-images [9]. We have considered all the directions (360 degrees), as well as a predefined distance of one pixel in the formation of the matrices.

An FPGA-based system for the computation of Haralick Features. With this method, some of the features are taken. In this entropy and Sum of absolute difference is calculated in the extraction of data. Some of the features are:

1. ENTROPY:

Entropy shows the amount of information of the image that is needed for the image compression. Entropy measures the loss of information or message in a transmitted signal and also measures the image information.

ISSN: 2278 – 909X

$$\text{Entropy} = -\sum_{i=0}^{Ng} \cdot \sum_{j=1}^{Ng} pij.\log(pij) \text{----eq 2}$$

2. SUM OF ABSOLUTE DIFFERENCE:

Sum of Absolute Difference is a widely used and extremely simple algorithm for finding the correlation between image blocks.

It works by taking the absolute difference between each pixel in the original block and the corresponding pixel in the block being used for comparison.

$$\text{SAD} = 1/n. \sum_{i=1}^{n} |xi - m(x)| \text{--- eq 3}$$

Where n is the number of patterns in the matrix
Xi is data element
M(x) is the mean

Entropy is calculated for each pixel value. Sum Entropy is calculated as the summation for all the pixel values.

3.1 ARCHITECTURE DESCRIPTION

The proposed architecture consists of two stages, a pre-processing stage and the feature extraction block (Fig. 2). The first prepares input data to be processed by the feature extraction block while the second combines both software and hardware to calculate GLCM features. Most of the GLCM feature vectors are calculated in hardware.

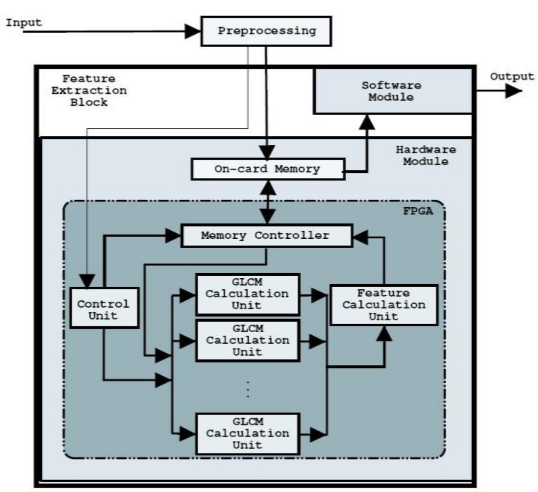

Fig-1 Overview of Architecture

3.1.1 PRE-PROCESSING:

The pre-processing handles the conversion of an image into an array A suitable for processing by the feature extraction block. Each element a = [a0a1a2a3a4] of A corresponds to each pixel. It is formed by five integers. The first (a0) is the gray level of the corresponding pixel, while the others (a1, a2, a3, a4) are the gray levels of its first neighbors for all the directions considered.

3.1.2 FEATURE EXTRACTION BLOCK:

The feature extraction block consists of hardware and a software module. The hardware module is implemented on a Xilinx FPGA using VERILOG.

The FPGA architecture consists of:

• A control unit that coordinates the functionality of the FPGA, by generating the signals that synchronize the other units.
• A memory controller that handles transactions from and to the on-card memory
• A feature calculation unit capable of reading GLCMs, calculating the feature vectors and storing them into the on card memory.

3.1.3 GLCM CALCULATION UNIT:

Gray Level Co-occurrence Matrix is a tabulation of how often different combinations of pixel brightness values occur in an image. GLCM contains the information about the positions of pixel having similar gray level values. GLCM calculation units receive pairs of gray level values as input. The GLCM calculation unit consists of the different combinations of gray values like a0b1, a2b3, a10b21 etc. This gives the deviation present in the image when compared with original image by predictive image.

The Gray-Level Co-occurrence Matrix (GLCM) considers the relationship between two neighboring pixels, the first pixel is known as a reference and the second is known as a neighbor pixel [8]. The GLCM is a square matrix with Ng dimension, where Ng equals the number of gray levels in the image. Each element of the matrix is the numbers of occurrence of the pair of pixel with value i and a pixels with value j. A co-occurrence matrix is a two dimensional array in which both rows and columns represent a set of possible image values.
For example consider the 4 by 5 matrix of an image I

1	1	5	6	8
2	3	5	7	1
4	5	7	1	2
8	5	1	2	5

Fig-2 Image Matrix

ISSN: 2278 – 909X

Equivalent GLCM matrix for the above image I is

1	2	0	0	1	0	0	0
0	0	1	0	1	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0
1	0	0	0	0	1	2	0
0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0

. Fig-3 GLCM Image Matrix

Element (1, 1) in the GLCM contains the value 1 because there is only one instance in the image where two, horizontally adjacent pixels have the values 1 and 1. Element (1, 2) in the GLCM contains the value 2 because there are two instances in the image where two, horizontally adjacent pixels have the values 1 and 2. Gray co matrix continues this processing to fill in all the values in the GLCM.

This processing is divided into four modules. The first module is pre-processing block, second module is GLCM calculation unit, third module is Feature calculation unit and last module is Control Unit.

In the first module we are loading the image data values by giving some positions to the values and rounding the values in order to avoid the merging of data. In the Second module we are comparing the input image data values with the predictive values. In this image data, each and every pixel is compared with the other and for this comparison we are using fast full search algorithm. The size of the image is 128*128 pixels. After comparing we get the deviation which is nothing but the contrast.

In the Third module the input is contrast and we are getting the output that is how much amount of data is shifted from the original data which is nothing but the Entropy. In this we are using Block Matching Algorithm.

In the Fourth module the input is entropy and we are getting the output that is data is shifted back to the original data values.

MODULE - 1:

In this module the input image data is obtained from Pre-processing block. Pre-processor is a program that processes its input data to produce output which will be used as input to another program.

By accepting the input we are allocating the position and the corresponding rounded value i.e. (where the data location is there in that instant which is the expected location but not the exact location) for that particular pixel data. For storing the position and rounding values, we are using on-card memory. Rounded value gives the information about the neighboring pixels exact location value.

The input data is of 8 bit width and for each byte there will be one position and rounding data. For example if the 7^{th} bit is 1 then some position is assigned and the rounding data will be the remaining bits. If the 6^{th} bit is 1 then another position is assigned and so on. In this if Reset is 1 then the position and the rounding value will be zero.

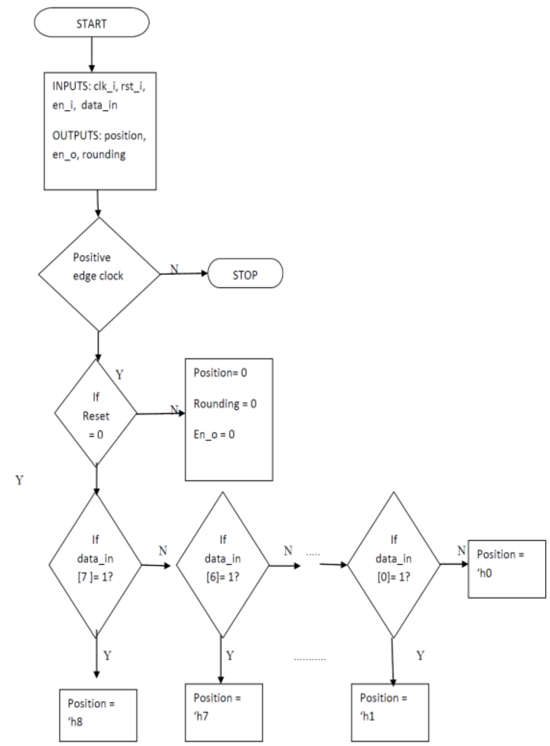

Fig-4 Module-1 Algorithm analysis

MODULE -2:

In this module we are comparing the input values with the predictive values. In this we are comparing every pixel values with the predictive pixel values which gives the deviation which is nothing but the Contrast. In this we are using Fast Full Search technique. In this we have taken two inputs. Among those inputs one of them is reference point and another one is predictive point .Comparison can be done for each and every bit. The result is nothing but the deviation point from the original point.

Fast Full Search: In this technique we propose a novel algorithm, referred to as Enhanced Bounded Correlation (EBC) that significantly reduces the number of computations required to carry out template matching based on Normalized Cross Correlation (NCC) and gives exactly the same result as the full search algorithm. The algorithm reduces the amount of computation of the Full search algorithm. We obtain faster elimination of continuous pairs

ISSN: 2278 – 909X

using this technique. This is a faster technique when compared to Full search algorithm.

In this module the inputs are the data-in which is the original / reference image data values and the other image is predictive image data values which are 8-bit data. For example 0 in the original data value is compared with the predictive image data of 1 and this operation is repeated until all the pixels are compared with each other. The output is the deviation or contrast.

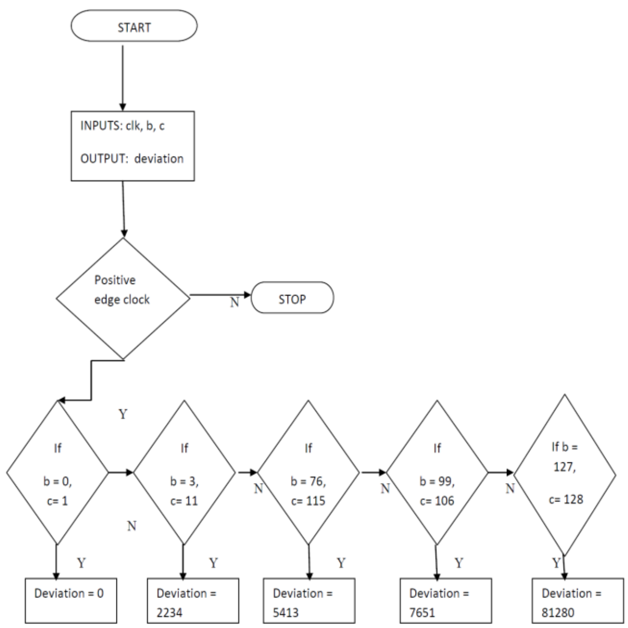

Fig-5 Module-2 algorithm analysis

MODULE -3:

In Module 2 we found out the amount of data that is shifted from the original image by comparing with reference image. That is called Deviation.

In Module 3 we are going to find out the entropy by giving the module-2 output i.e., deviation as input and also Clock will be given as inputs in this module. We will be getting the output which is nothing but the amount of part that the deviation point is outside of the comparison block. In this we are using Block Matching Algorithm.

Block Matching Algorithm: Block Matching Algorithm is a way of locating matching blocks in a sequence of digital video frames for the purposes of motion estimation. The purpose of a Block Matching Algorithm is to find out a matching block from a frame i in some other frame j which may appear before or after i. This can be used to discover temporal redundancy in the video sequence, increasing the effectiveness of inter-frame video compression. Block matching algorithms make use of criteria to determine whether a given block in frame j matches the search block in frame.

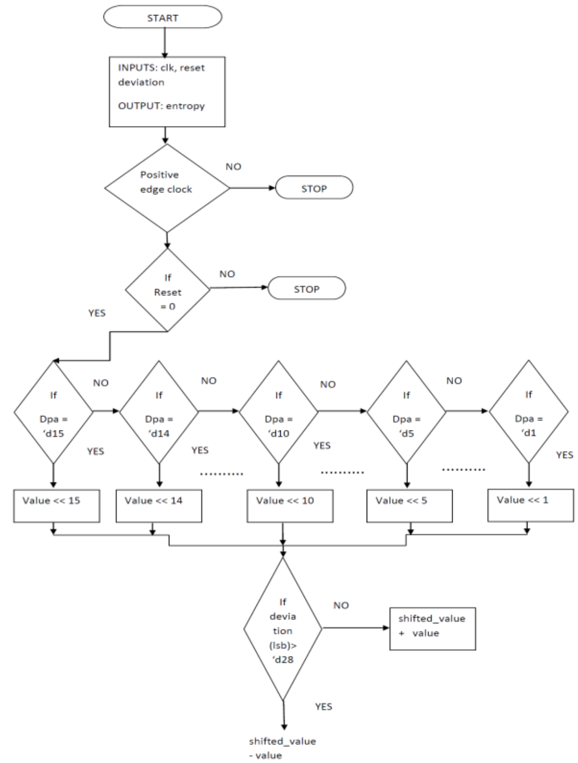

Fig-6 Module-3 algorithm analysis

MODULE -4:

In Module 3 we are found out the entropy which is nothing but error. This error needs to be corrected and this correction will be done in this module.

In Module 4 we are going to correct / fix the entropy by giving the module-3 output i.e., 6-bit entropy as input and also Clock, Reset will be given as inputs to this module. After the block to block matching procedure will be finished then we can get the exact location for that data, which is nothing but the output of this particular module.

If the reset is 1 then the output is 0. In this 6-bit entropy is considered and we will be getting 64 combinations. If the input 6-bit entropy is 0 then the output fixation will be zero. If the input 6 bit entropy is 15 then the output will be 4548 and this procedure is repeated up to 64 combinations.

ISSN: 2278 – 909X

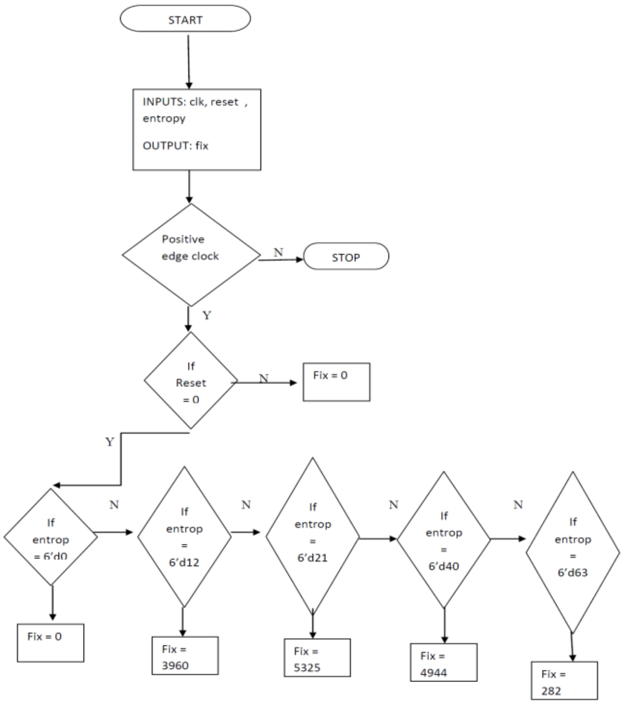

Fig-7 Module-4 algorithm analysis

4. SIMULATION RESULTS

This plot explains the module 1 used for the data taken with clock which represent the output data

.

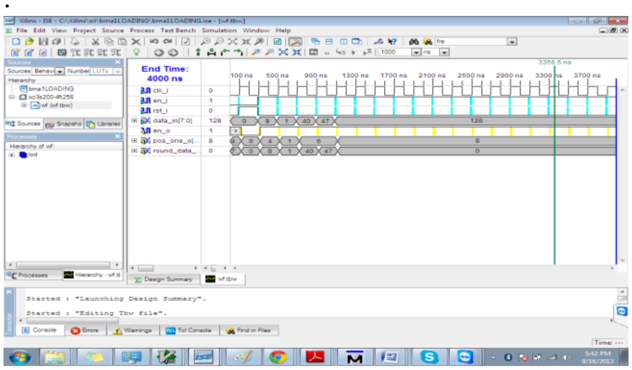

Fig-8 Module 1 data output

This plot explains about the module 2 plot proposed architecture in graphical representation.

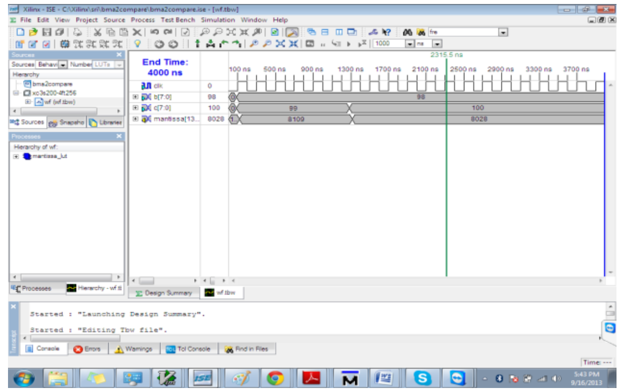

Fig-9 Simulation result of module-2.

This plot explains about the module 3 plot of proposed architecture in graphical representation.

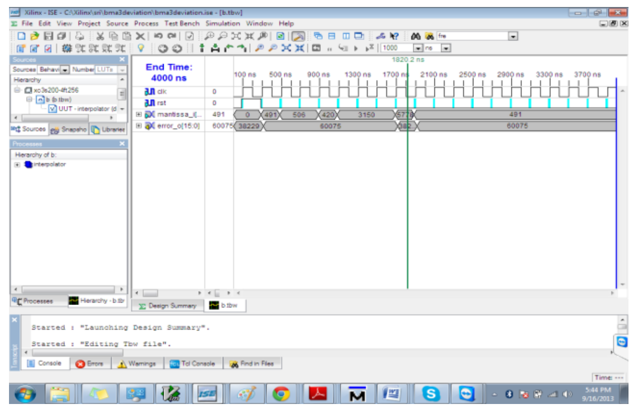

Fig-10 Simulation result of module 3

This plot explains about the module 4 plot of proposed architecture in graphical representation.

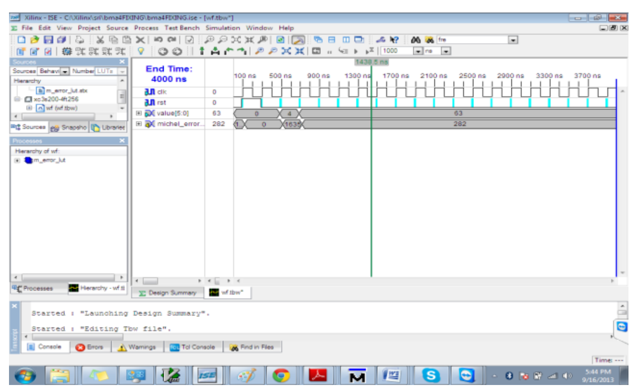

Fig-11 Simulation result of module -4

This plot explains the RTL schematic of the module 1 which gives the internal flip-flops and gates.

ISSN: 2278 – 909X

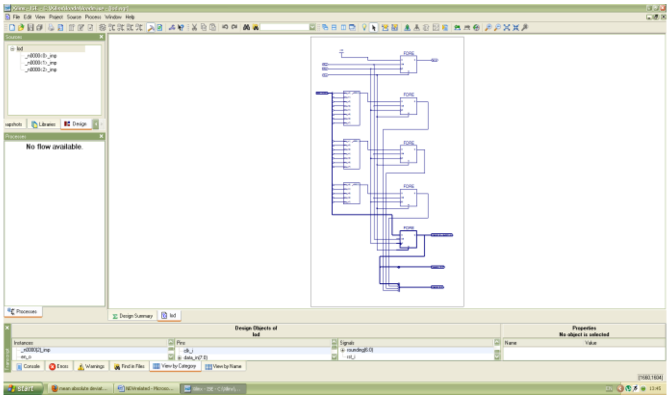

Fig-12 RTL schematic of module-1

This plot explains the RTL schematic of the module 2 which gives the internal flip-flops and gates.

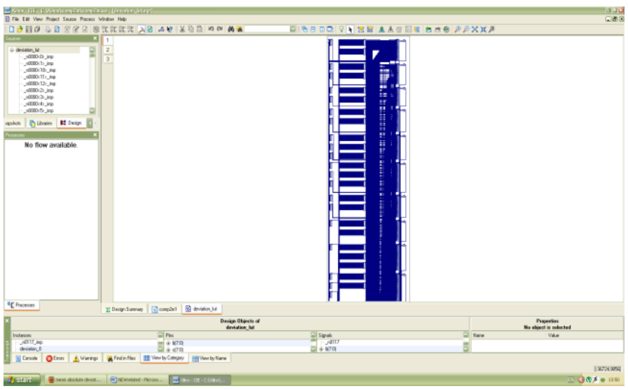

Fig-13 RTL schematic of module-2

This plot explains the RTL schematic of the module 3 which gives the internal flip-flops and gates.

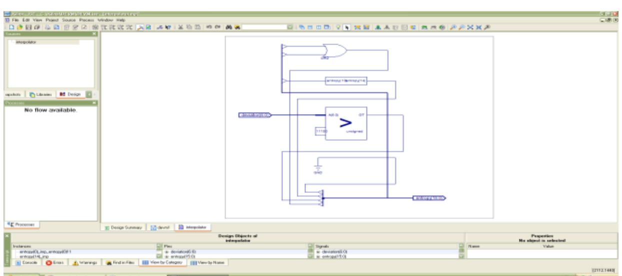

Fig-14 RTL schematic of module-3

This plot explains the RTL schematic of the module 4 which gives the internal flip-flops and gates.

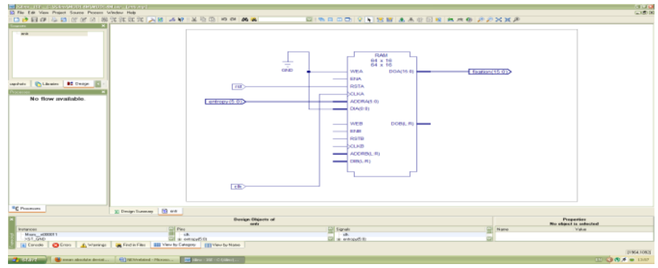

Fig-15 RTL schematic of module-4

6. CONCLUSION

We proposed and implemented a novel FPGA-based architecture for real-time extraction of four GLCM features. A 128*128 size gray level image is taken for feature extraction. The Digital image features were extracted using statistical second order method. In this thesis Gray Level Matrix is considered to extract features, which is a statistical method based on the gray level value of pixels. This method was proposed by Haralick. These features are further used in Image segmentation, which are used to identify the regions which are similar and different of sub images. Using this statistical method we obtained the image extraction in gray-scale level analysis which can be further extended to different algorithms represented theoretically in feature extraction of images.

References

[1] R.E. Haralick, K. Shanmugam, I. Dinstein, *Textural Features for Image Classification*, IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-3, No. 6, Nov 1973
[2] J.L. Hennesy & D.A. Patterson, *Computer Architecture,A Quantitative Approach*, Morgan Kaufmann, May 2002
[3] D.E. Maroulis, D.K. Iakovidis, S.A. Karkanis, D.A. Karras, *CoLD: a versatile detection system for colorectal lesions in endoscopy video frames*, Computer Methods and Programs in Biomedicine, vol 70, no. 2, pp. 99-186, Elsevier Science, 2003.
[4] W. Luk, P. Andreou, A. Derbyshire, F. Dupont-De-Dinechin, J. Rice, N. Shirazi and D. Siganos, *Field-Programmable Logic: From FPGAs to Computing Paradigm*, Springer-Verlag, Berlin, 1998.
[5] M. Nibouche, A. Bouridane, D. Crookes, O. Nibouche, *An FPGA-based wavelet transforms coprocessor*, in Proc. IEEE Int. Conf. Image Processing, pp. 194-197, vol.3, 2003.
[6] H. Hikawa, *Implementation of Simplified Multilayer Neural Network with On-chip Learning*, Proc. of the IEEE International Conference on Neural Networks (Part 4), Vol. 4, 1999, pp 1633-1637.
[7] T. Nakano, T. Morie, and A. Iwata, *A Face/Object Recognition System Using FPGA Implementation of Coarse*

ISSN: 2278 – 909X

Region Segmentation, SICE Annual Conference 2003, pp. 1418-1423, Fukui, Aug. 4-6, 2003.

[8] K. Heikkinen and P. Vuorimaa, *Computation of Two Texture Features in Hardware*, Proceedings of the 10th International Conference on Image Analysis and Processing, Venice, Italy, pages 125-129, September 27-29, 1999.

[9] K.C. Chang, *Digital Design and Modeling with VHDL and Synthesis*, IEEE Computer Society Press - Wiley, 1997*Int. Symp. Integr. Circuits*, 2009, pp. 364–367.

BIOGRAPHIES

M.Harsha Vardhan pursuing M.Tech in specialization of communication systems at ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY in Department of Electronics and Communication Engineering.

S.Visweswara Rao working as an Assistant Professor in ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY in Department of Electronics and communication Engineering.

ISSN: 2278 – 909X