Image Compression using Fully Pipelined and Multiplierless 2D DCTI/DCT Architecture for JPGE Encoder

Shailesh Baberiya[1], Abhinav Shukla[2]

[1]Research Scholar, Department of Electronics & Communication Engineering, VIT, RKDF University Bhopal, India
[2]Professor, Department of Electronics & Communication Engineering, VIT, RKDF University Bhopal, India
[1]shaileshbaberiya1998@gmail.com, [2]abhinav.shukla@hotmail.com

* Corresponding Author: Shailesh Baberiya

Abstract: *Picture compression reduces the size of a design document in bytes without distorting the image's nature to an unacceptable level. The reduction in document size makes it possible to fit more photos into a given amount of space in a circle or memory. Additionally, it reduces the amount of time needed for photographs to be downloaded from websites or sent over the Internet. JPEG and JPEG 2000 are two crucial picture compression techniques. The 2D-DCT and 2D-IDCT have also been implemented using VLSI technology. This VLSI design is using low-quality DCT to implement JPEG image compression and picture compression. Any image can be thought of as a fuzzy network (connection) by normalising the estimates of its pixels, which is separated into smaller networks (perhaps square ones) referred to as fragments. Each square is associated with the discrete fuzzy change of a capacity in two factors, and it is gradually decompressed through the associated opposite fuzzy change. The image is recreated using the decompressed pieces, and the quality is determined by computing the PSNR (Peak Signal to Noise Ratio) and MSE (Mean Square Error) for the first image.*

Keywords: JPEG, JPEG 2000, 2D-DCT, 2D- IDCT, fuzzy transform, PSNR (Peak Signal to Noise Ratio), MSE (Mean Square Error).

I. Introduction

How does data compression work? One of the technologies that enabled the multimedia revolution is data compression, which is essential for the swift advancement of information technology. Data compression is the process of downsizing data files to improve storage and transmission efficiency. Digital data have emerged as a crucial source of information in the current communication systems world. Why is data compression necessary? The JPEG and MPEG compression standards for still images and videos, respectively, are Joint Photographic Experts Group and Moving Pictures Experts Group. In these standards, data compression algorithms are used to reduce the number of bits needed to represent an image or a video sequence. The process of delivering information in a tangible form is known as compression. Data compression treats information in digitised form, i.e., as binary integers represented by very large data units in the form of bytes of data. For instance, a single small 4" by 4" colour photograph that is scanned at 300 dots per inch (dpi) with 24 bits of actual colour per pixel results in a file that contains more than 4 megabytes of data. A minimum of three flexible discs are needed to store such a picture. This image takes longer than a minute to transmit over a standard transmission line (ISDN at 64 kbps). Large image documents continue to be a major bottleneck in a communicated environment because of this. Although increasing the bandwidth is a potential solution, the very high cost makes this considerably less appealing. As a result, compression is a necessary and essential method for creating image files that can be managed and sent.

II IMAGE COMPRESION AND RECONSTRUCTION

The three primary data redundancies for the picture compression standard are as follows: -

1. Spatial redundancy caused by the correlation of nearby pixels
2. Spectral redundancy caused by the colour components' relationship.
3. Psychovisual redundant processes brought on by features of the human visual system.

The spatial and extraterrestrial redundancies are made possible by the fact that certain extraterrestrial and spatial examples exist among the pixels and the shading segments, even though the psycho-visual repetition begins with the human eye's aversion to specific extraterrestrial frequencies. The goal here is to obtain a satisfactory printout of a computerised image while also safeguarding the essential data included in that Pacific informational collection.

1.1 IMAGE COMPRESSION METHODS :

Two types of image compression methods which is defined as follows:

- Lossy compression
- Lossless compression

1.1.1 Lossy Compression :-

A lossy compression strategy involves compressing information and then decompressing it to recover information that is sufficiently close to the original to be useful in some way. Lossy pressure is most frequently employed to pack visual and auditory information (sound, video, and still images), particularly in applications like streaming media and web communication. For informational documents such as bank records, content articles, and so forth, differentiated, lossless stress is necessary. It's frequently worthwhile to create an error-free document that may be used to create picked records for a variety of purposes; For instance, a multi-meg document can be used in its entirety to deliver a full-page advertisement in a glossy magazine, while a 10 kilobyte lossy duplicate can be made for a small image on a website page. For instance, a multi-meg document can be used in its entirety to provide a full-page advertisement in a colourful magazine, while a 10 kilobyte lossy duplicate can be made for a little image on a website page. The strategies for lossy compression include JPEG compression, Wavelet compression, and fractal compression.

1.1.2 Lossless Compression :-

The goal of lossless picture compression is to transmit a photo motion with the fewest amount of bits possible without losing any data, hence accelerating transmission and restricting stockpiling conditions. Lossless image compression is a type of image compression calculation that enables the creation of the correct, unique image from a captured image. The amount of bits communicating to the flash is often communicated as a standard number of bits per test for still images and a standard number of bits per second for video. Certain image file formats, most notably PNG, only employ lossless compression.

These three techniques—Huffman encoding, arithmetic coding, and run-length coding—are used in lossless compression.

1.2 Performance Measures of Image Compression:-

the three parameters below are used to gauge how well a data compression strategy works. These are:

I. Effectiveness of compression

II. Complexities

III. Measurement of distance for a loss of impression.

The compression ratio is used to assess the compression's effectiveness (CR). The CR can be defined as the ratio of the original data length (number of bits) to the size of the corresponding compressed data. The data operations comprise additions, subtractions, multiplicative and distributive operations, and shift operations. The complexity of a digital data compression algorithm is measured in encoding and decoding operations by the number of required data operations.

When using a lossy compression algorithm, the amount of information lost during the reconstruction of the original signal is measured by the distortion measurement. The performance of the lossy compression algorithms is also measured by two performance parameters, Mean Square Error (MSE) and Signal to Noise Ratio (SNR). For 1D data, these measurement criteria are frequently utilised. For image data (2D compression), a parameter known as the peak signal to noise ratio replaces the SNR (PSNR). I-D data uses the same measurement parameters as well. The metric known as the peak signal to noise ratio is used to replace the SNR parameter for 2-D (Image Data) compression (PSNR). Measurement of distribution is also reflected in the percentage of energy that is retained in the compressed data.

III TRANSFOMING AGLORITHM :-

This study presents the discrete cosine transform method for image compression. Rthogonal approximation utilising a simple, low-complexity discrete cosine transform Also covered is the discrete cosine transform for compressing images. The well-known JPEG technique is used to compress images using the discrete cosine transform [3–4]. Another picture compression algorithm that uses the Rthogonal Approximation DCT is that provided by Renatto J. Cintra and Fabio M. Bayer [4].

3.1 IMAGE COMPRESSION USING DCT :-

The most well-liked and efficient technique for still image compression is JPEG. It serves as the global standard for persuasive communication. JPEG stands for Joint Photographic Experts Group, which is also the name of the committee that developed the JPEG standard [3–4]. In JPEG, you can compress an image into a stream of bytes and then decompress it to get the original image back.

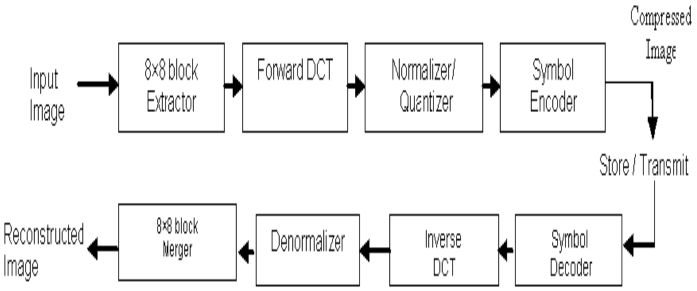

FIG. 3.1:- SCHEMATIC DIAGRAM OF JPEG COMPRESSION

JPEG may adjust the compression ratio and image quality based on user requirements. The file is divided into 8 X 8 blocks and the RGB (colour representation) made in the JPEG process is converted to YUV. Transform the pixel information using the discrete cosine transform from the spatial domain to the frequency domain. By dividing each coefficient by an integer value and rounding off to the nearest integer, quantize the resulting values.

3.2 Simulation Results for the JPEG Compression:-

For our effects, 4 test images are taken from the image database and resize them to a size of 256x256.

Fig 3.2 Test images (a) cameraman (b) lena (c) circuits (d) pappers.

3.3 Orthogonal Approximation Discrete Cosine Transform:

For the 8-point discrete sine transform (DCT), only zeros and ones are present in the transposition matrix; multiplications and bit shift operations are absent. As a design criterion, precise spectral behaviour in relation to the DCT was adopted. The suggested algorithm outperforms the signed discrete cosine transform. It could also execute state-of-the-art algorithms in both low and high image compression scenarios, displaying comparable computational complexity at the same time.

It is sought in adjustment matrix A to orthogonalize B_0. In fact, the orthogonalization matrix is provided by:

$$\mathbf{A} = \sqrt{(\mathbf{B_0}.\ \mathbf{B_{0T}})}\ \mathbf{-1}$$

where the matrix square root is taken in the principal sense. This computation furnishes the diagonal matrix

A= diag $(1/2\sqrt{2},1/\sqrt{6},1/2,1/\sqrt{6},1/2\sqrt{2},1/\sqrt{6},1/2,1/\sqrt{6})$

$$\mathbf{S} = \mathrm{diag}(1/2\sqrt{2}, 1/\sqrt{6}, 1/2, 1/\sqrt{6}, 1/2\sqrt{2}, 1/\sqrt{6}, 1/2, 1/\sqrt{6})$$

Therefore, the DCT matrix can be more adequately approximated by the following proposed matrix:.

$$\widehat{\boldsymbol{B}}_{\text{orth}} = \mathrm{A.}\ \mathbf{B0}$$

Matrix $\mathbf{B_{0rth}}$ possesses useful properties:

i) it is orthogonal;

ii) it inherits the low computational complexity of $\widehat{\boldsymbol{B}}_{\text{orth}}$; and

iii) the orthogonalization matrix **A** is diagonal.

The image or any two dimensional signal can be transformed to the with the help of B̂ orth the another domain

$$f\,(\boldsymbol{u},\boldsymbol{v})\ \widehat{\boldsymbol{B}}_{\text{orth}} * \boldsymbol{f}(\boldsymbol{x},\boldsymbol{y}) * \widehat{\boldsymbol{B}}_{\text{orth}} \quad \ldots\ldots\ldots.(3.2)$$

For the purpose of image compression, we have used the OA – DCT in the similar way as the DCT was used. The remaining methodology adopted is same for the OA-DCT that used for JPEG.

3.4 A New Modal for Image Compression Using DCT VLSI Architecture :-

This work proposes a novel VLSI architecture for image compression modelling using a discrete cosine transform. Verilog HDL may also be used to implement a VLSI design. The proposed hardware architecture was created using an RTL compiler and mapped using 180 nm standard cells. Modelsim is used to conduct the simulation. The creation occurs as a result of looking into MATLAB and Verilog HDL. Using the RTL compiler from CDENCE, point by point investigation for power and region was completed. In this procedure, the input image is divided into 8 × 8 non-overlapping blocks and entered into the baseline encoder. The power consumption is lower, at 1.027mW at the very least.

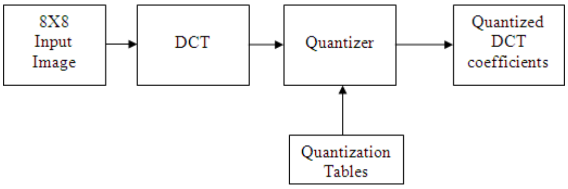

Fig. 3.3 :- Block diagram of DCT Process

DCT transforms the pixel data into a block of spatial frequencies, which yields the DCT coefficients. Most of the block energy will be stored in the lower spatial frequencies as a result of the DCT output since pixels within the 8 x 8 neighbourhood typically have minimal variations in grey ranges.

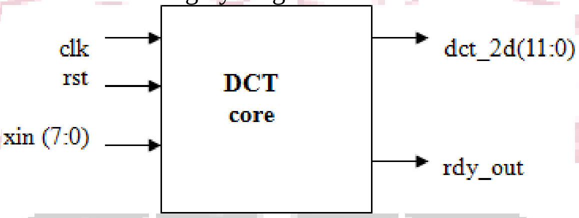

Fig. 3.4 : Top level schematic for DCT core

The controller/subtractor is connected to CLK. The adder/subtrator module alternately selects addition and subtraction for each clk. The toggle switch handles this selection. The multiplier's other input is connected to stored values in registers that function as memory, and the add-output sub's is fed into it. The outputs of the four multipliers are added in the final order at each CLK. The second-DCT coefficient values sent outside the request where the data sources were read-in are the results from the snake in the second phase. At the point where the 2d DCT calculation is over the red out flag, the yield is being excessively demonstrated.

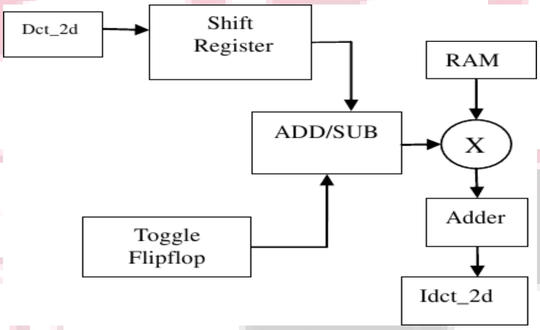

Fig 3.5 :- IDCT Architecture

The 1D-IDCT is implemented on the input DCT values, as shown in Fig. 3.3. This produces a purple output known as the intermediate value in an RM.

The final 2D-IDCT output IDCT 2D is delivered by performing a second 1D-IDCT operation on this stocrimson value. The 2D-IDCT output is eight bits wide, whereas the inputs are 12 bits wide. The input signals are taken one pixel at a time in the range of x00 to x07, x10 to x07, and so on up to x77 in the first 1D section. The 8 bit shift register receives such inputs as inputs. The eight bit shift register outputs are reported every eighth clock cycle. We shall be able to sign in or register in 8 pixels (one row) at a time thanks to this.

TABLE I: Characteristic of DCT and IDCT

Feature	DCT	IDCT
No. of cells	6773	8571
Block Size	8X8	8X8
Latency	92 cycles	85cycles
Power	1.0271 mW	1.261 mW
Area	0.6281mm^2	0.7652mm^2

TABLE II: Performance Comparison

DCT Architecture	Power in mW	Area in mm^2
Proposed DCT core	1.0271	0.6281
DCT Architecture	29.78	0.343
DCT core	29.92	0.569

IV DESIGN SIMULATION :-

Fuzzy transforms developed into something incredibly versatile and powerful. The inverse F-transform function has adequate filtering capabilities that can be used to remove noise from images or from any other type of data. For the compression of data, F-transforms can be applied. We'll also demonstrate how this methodology may be applied to data compression and decompression.

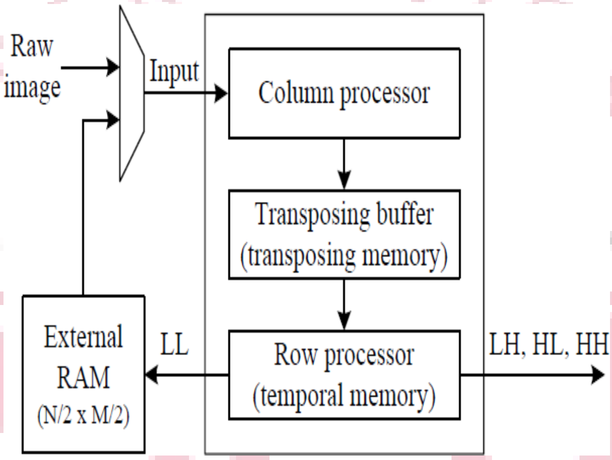
Raw image
Input
Column processor
Transposing buffer (transposing memory)
External RAM (N/2 x M/2)
LL
Row processor (temporal memory)
LH, HL, HH

Fig 4.1: The overall 2-D DCT architecture

The foreward and inverse 2-D DCT transformations are carried out by the specified architecture in a column-row fashion. The overall 2-D DCT architecture is depicted in Fig. 4.1 and consists of the column processor, the transposing buffer, and the row processor.

The given architecture simultaneously executes the column processor, the transposing buffer, and the read processor in order to carry out the one-stage decomposition for a NxM image, where N and M stand for the image's top and bottom. Additionally, the row processor must implement a row-smart transform if sufficient column-processed data have been obtained in order to reduce the internal memory size. The LL band output coefficients are then stored for the following stage decomposition using the MN/4 size external RM.

V SIMULATION RESULTS :-

The fig. 5.1 shows the schematic symbol of column processor in XILINX. This section deals with the simulation of column processor in XILINX ISE simulator.

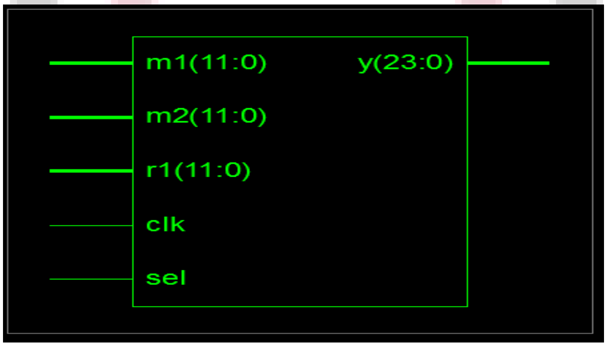

Figure 5.1: Schematic Symbol of Column Processor in Xilinx

The fig 5.2 shows the Test Bench Waveform of 12 Bit Column Processor.

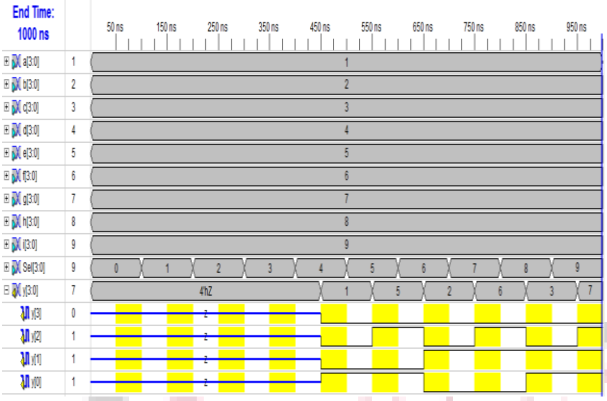

FIGURE 5.2 : **Test Bench Waveform of 12 Bit Column Processor**

5.1 Simulation Results of 16 Input Transposing Buffers:-

This section deals with the simulation of Transposing Buffer in XILINX ISE simulator. The fig. 5.3 shows the schematic symbol of Transponding Buffer in XILINX.

The fig 5.4 shows the Test Bench Waveform of Transposing buffer in XILINX.

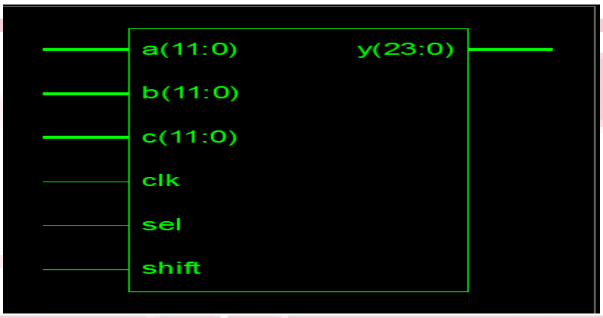

Fig.5.3 : Technology Schematic of Transposing Buffer in Xilinx

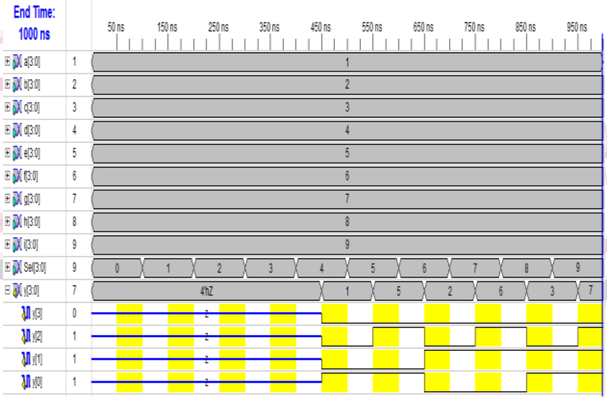

Figure 5.4 : Test Bench Waveform of Transposing Buffer in Xilinx

5.2 Simulation Results of 12 Bit Row Processor:-

In this section we discuse 24 Bit 2X1 Demultiplexer in XILINX ISE simulator. The fig. 5.5 shows schematic symbol of 12 Bit Row processor in XILINX Simulator.

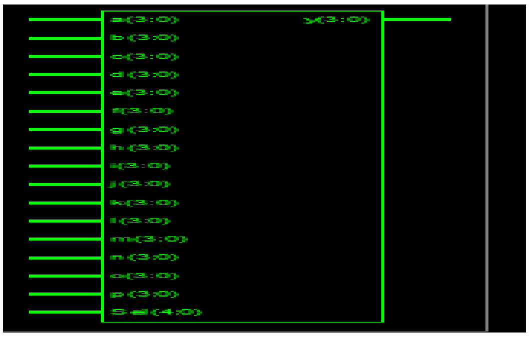

Figure 5.5:- Schematic Symbol of 12 Bit Row Processor in Xilinx

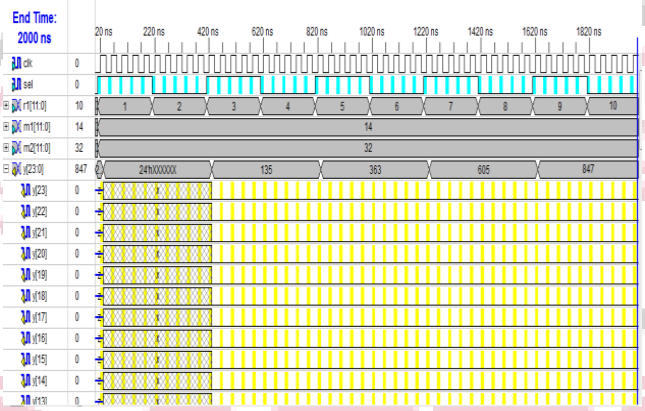

Fig 5.6 :- Test bench waveform of 12 Bit Row Processor in Xilinx

VI CONCLUSION:-

The primary objective of those tactics is to make the DCT method less complex than it is currently. A "Discrete Csine Transform" method is used, for example, in the first step of compression. The second use the "DCT approximation" methodology for compression. The third expresses itself via a "Low Complexity 88 Transform" method.

We offer a novel technique for image compression. This method's first objective is to make things less complicated and, secondly, to make the impression approach quickly with a credible image impression. This new approach makes the compression and decompression process more efficient. Without compression, both the transmission time and the amount of storage needed could be high, but the inclusion of compression and reconstruction minimises both of these factors. The VLSI architecture for the 2DDCT and 2DIDCT, a very effective approach for image compression, has also been detailed in this chapter. The described VLSI architecture can also be used to perform the image compression.

REFERENCE:

[1] Wei Zhang, Member, IEEE, Zhe Jiang, Zhiyu Gao, and Yanyan Liu, "An Efficient VLSI Architecture for Lifting-Based Discrete Wavelet Transform" IEEE Transactions On Circuits And Systems—Ii: Express Briefs, Vol. 59, No. 3, March 2015.

[2] Chih-Hsien Hsia, Member, IEEE, Jen-Shiun Chiang, Member, IEEE, and Jing-Ming Guo, Senior Member, IEEE "Memory-Efficient Hardware Architecture of 2-D Dual-Mode Lifting- Based Discrete Wavelet Transform" IEEE Transactions On Circuits And Systems For Video Technology, Vol. 23, No. 4, April 2014.

[3] Jinook Song, Student Member, IEEE, and In- Cheol Park, Senior Member, IEEE, "Pipelined Discrete Wavelet Transform Architecture Scanning Dual Lines" IEEE Transactions On Circuits And Systems—Ii: Express Briefs, Vol. 56, No. 12, December 2013.

[4] Yeong-Kang Lai, Member, IEEE, Lien-Fei Chen, Student Member, IEEE, and Yui-Chih Shih, "A High-Performance and Memory- Efficient VLSI Architecture with Parallel Scanning Method for 2-D Lifting Based Discrete Wavelet Transform" IEEE Transactions on Consumer Electronics, Vol. 55, No. 2, MAY 2009 Contributed Paper.

[5] Basant Kumar Mohanty, Senior Member, IEEE, and Pramod Kumar Meher, Senior Member, IEEE,"Memory-Efficient High-Speed Convolution-based Generic Structure for Multilevel 2-D DWT" IEEE Transactions On Circuits And Systems For Video Technology, Vol. 23, No. 2, February 2013.

[6] Yusong Hu, Student Member, IEEE, And ChingChuen Jong, Member, IEEE, "A Memory- Efficient High-Throughput Architecture For Lifting-Based Multi-Level 2-D Dwt" IEEE Transactions On Signal Processing, Vol. 61, No. 20, October 15, 2013.

[7] UshaBhanu.N and Dr.A. Chilambuchelvan, "A Detailed Survey on VLSI Architectures for Lifting based DWT for efficient hardware implementation" International Journal of VLSI design & Communication Systems (VLSICS) Vol.3, No.2, April 2012.