IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735
PP 07-12
www.iosrjournals.org

A Hardware Reconfigurable For Dtt Based On Image And Video Coding

B.Arun [1], K.Somu[2]

PG Student, VLSI Design, Department of ECE, Maha Barathi Engineering College, India[1].
Assistant Professor, Department of ECE, Maha Barathi Engineering College, India[2]

Abstract: *Advances in the areas of image coding have generated a growing interest in discrete transforms. The demand of high quality with a limited use of computational resources and improved cost benefits has lead to experimentation with transform coding methods. This project deals with Discrete Tchebichef Transform (DTT) which is a polynomial-based orthogonal transform. Image quality measures that span both the subjective and objective evaluation techniques are computed for the compressed images and the results are analyzed by taking the statistical properties of the images into account. The advantages of DTT are simplicity, good image compression potential and computational efficiency. The performance of the transform is compared with the existing transform like Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT). FPGA implementation using Virtex-6 kit is also conducted to prove the effectiveness of the transform.*

I. INTRODUCTION

In most of the existing frame recompression methods are designed independently of video encoder. Unfortunately, they did not intend to finely cooperate with encoder, but only focus on improving the compression ratio. Actually, many information can be used for frame recompression and make it work more efficiently with encoder [1]. The most significant part of multimedia systems is application involving image or video, which require computationally intensive data processing. Moreover, as the use of mobile device increases exponentially, there is a growing demand for multimedia application to run on these portable devices. A typical image/video transmission system is shown in the Figure 1.1.

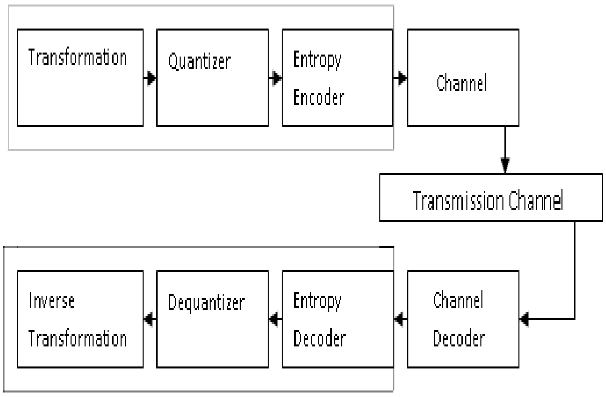


Figure1.1. Image/Video Transmission System

A. Discrete cosine transforms

Discrete cosine transform (DCT) is one of the major compression schemes owing to its near optimal performance and has energy compaction efficiency greater than any other transform. The principle advantage of image transformation is the removal of redundancy between neighboring pixels. This leads to uncorrelated transform coefficients which can be encoded independently. DCT has that de correlation property.The transformation algorithm needs to be of low complexity. Since the DCT is separable 2-D [2] can be obtained from two 1-D DCTs. The 2-D DCT equation is given by Equation (1)

$$C(u,v)=\alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{\pi(2x+1)u}{2N}\right]\cos\left[\frac{\pi(2y+1)v}{2N}\right], \qquad 1$$

[1]

For u,v= 0,1,2,…,N −1 .

The inverse transform is defined by Equation ()

$$f(x,y)=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\alpha(u)\alpha(v)C(u,v)\cos\left[\frac{\pi(2x+1)u}{2N}\right]\cos\left[\frac{\pi(2y+1)v}{2N}\right], \qquad 2$$

[2]

For x,y= 0,1,2,…,N −1. The 2-D basis functions can be generated by multiplying the horizontally oriented 1-D basis functions with vertically

B. Quantization

To discard an appropriate amount of information, the compressor divides each DCT [4] output value by a quantization coefficient and rounds the result to an integer. The larger the quantization coefficient, the more data is lost, because the actual DCT value is represented less and less accurately. Each of the 64 positions of the DCT output block has its own quantization coefficient, with the higher-order terms being quantized more heavily than the low-order terms i.e. the higher-order terms have larger quantization coefficients. The resulting coefficients contain a significant amount of redundant data. Huffman compression will losslessly remove the redundancies, resulting in smaller data. A typical quantization matrix, as specified in the original JPEG Standard, is as follows:

$$Q=\begin{bmatrix}16 & 11 & 10 & 16 & 24 & 40 & 51 & 61\\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55\\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56\\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62\\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77\\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92\\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101\\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99\end{bmatrix}.$$

C. Entropy encoding

Entropy coding is a form of lossless data compression. The steps involved in entropy encoding are arrangement of image components in a zigzag manner employing run length encoding (RLE) algorithm. The RLE groups similar frequencies together, inserting length coding zeros, and then using Huffman encoding on what is left. The arithmetic coding technique is mathematically superior to Huffman coding and the JPEG standard also allows its though it is not mandatory.

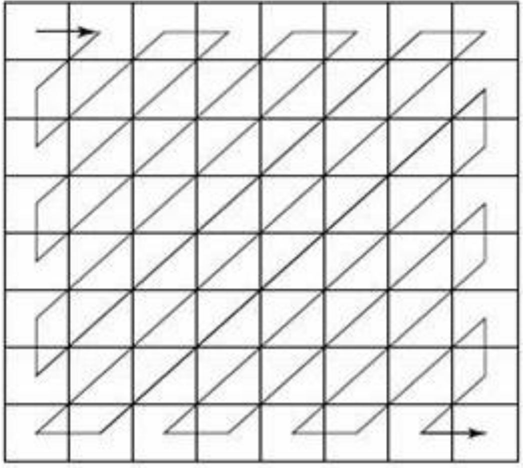


Figure 1.2 Zigzag ordering

II. RELATED WORK

A novel radix-4^2 algorithm with the low computational complexity of a radix-16algorithm but the lower hardware requirement of a radix-4 algorithm. The proposed pipeline radix-4^2 single delay feedback path (R4^2SDF) architecture adopts a multiplier less radix-4 butterfly structure, based on the specific linear mapping of common factor algorithm (CFA) [2], to support both 256-point fast Fourier transform/inverse fast Fourier transform (FFT/IFFT) and 8times8 2D discrete cosine transform (DCT) modes following with the high efficient feedback shift registers architecture. A 100-MHz two-dimensional discrete cosine transform (DCT) core processor applicable to the real-time processing of HDTV signals [4] is described. An excellent architecture utilizing a fast DCT algorithm and multiplier accumulators based on distributed arithmetic have contributed to reducing the hardware amount and to enhancing the speed performance. A layout scheme with a column-interleaved memory and a new ROM circuit are introduced for the efficient implementation of memory-based signal processing circuits. Furthermore, mean values of errors generated in the core were minimized to enhance the computational accuracy with the word-length constraints. Conventional distributed arithmetic (DA) [2] is popular in application-specific integrated circuit (ASIC) design [5], and it features on-chip ROM to achieve high speed and regularity. In this paper, a new DA architecture called NEDA is proposed, aimed at reducing the cost metrics of power and area while maintaining high speed and accuracy in digital signal processing (DSP) applications. Mathematical analysis proves that DA can implement inner product of vectors in the form of two's complement numbers using only additions, followed by a small number of shifts at the final stage. Comparative studies show that NEDA outperforms widely used approaches such as multiply/accumulate (MAC) and DA in many aspects. Being a high-speed architecture free of ROM, multiplication, and subtraction, NEDA can also expose the redundancy existing in the adder array consisting of entries of 0 and 1. A hardware compression scheme is introduced to generate a butterfly structure with minimum number of additions. NEDA-based architectures for 8 × 8 discrete cosine transform (DCT) core are presented as an example. Discrete Cosine Transform (DCT) plays an important role in image and video compression, but computing a two-dimensional (2D) DCT, a large number of multiplications and additions are required in a direct approach. Multiplications, which are the most time-consuming and expensive operations in simple processor, can be completely avoided in our proposed architecture for multiple channel real-time image compression. In this paper, a compressed distributed arithmetic architecture for 2D 8times8 DCT is presented, which offers high speed and small area. The basic architecture consists of a ID row DCT followed by a transpose register array and another ID column DCT, in which an 8-input ID DCT structure only requires 15 adders to build a compressed adder matrix and no ROM is needed.

III. DISCRETE TCHEBICHEF TRANSFORM

The discrete Tchebichef transform (DTT) is a powerful method for image de coding and data de correlation [1]. In recent years, signal processing literature has employed the DTT in several image processing problems, such as artifact measurement [2], blind integrity verification [3], and image compression [4]. In particular, the 8-point DTT has been considered in blind forensics for integrity check of medical images [3]. For image compression, the 8-point DTT is also capable of outperforming the 8-point discrete cosine transform (DCT) in terms of average bit-length in bit stream codification [4]. Such high arithmetic complexity may be a hindrance for the adoption of the DTT in contemporary devices that demand low-complexity circuitry and low power consumption.

Methodology used:

- Step1: To drive discrete Tchebichef polynomials matrix for forward and inverse transform and its multiplication free hardware implementation.
- Step2: To prove the energy compaction properties of DTT matrix & to prove its efficiency over DCT.
- Step3: To reduce the dynamic ranges of DTT matrix to get approximation to reduce the complexity.
- Step4: to prove the efficiency of approximated DTT by implement it in image processing applications.

A) DCT process

In the DCT process input image is divided into nonoverlapping blocks of 8 x 8 pixels, and input to the baseline encoder. The pixel values are converted from unsigned integer format to signed integer format, and DCT computation is performed on each block. DCT transforms the pixel data into a block of spatial frequencies that are called the DCT coefficients. Since the pixels in the 8 x 8 neighborhood typically have small variations in gray levels, the output of the DCT will result in most of the block energy being stored in the lower spatial frequencies. On the other hand, the higher frequencies will have values equal to or close to zero and hence, can be ignored during encoding without significantly affecting the image quality.

B) Low Complexity 2-D DCT Using 1-D DCT

The 2-D DCT is computationally intensive and as such there is a great demand for high speed, high throughput and short latency computing architectures. Due to the high computation requirements,2-D DCT processor design has been concentrated on small non overlapping blocks(typically 8x8 or 16x16).Many 2-D

DCT algorithms have been proposed to achieve reduction of the computationally complexity and thus increase the operational speed and throughput.

The various algorithm and architecture for the 2-D DCT can be divided into two categories i) row column decomposition methods and ii.) Non row column decomposition methods. For a given 2D spatial data sequence {Xij; i, j = 0, 1, … , N-1}, the 2D DCT data sequence ,Ypq where p, q = 0, 1, … , N-1 is defined by Equation (3)

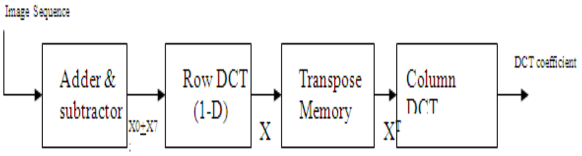


Figure1.3. Row Column Decomposed 2-D DCT Architecture

$$Y_{pq} = E_p E_q \frac{2}{N} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X_{ij} \cos\left[\frac{(2i+1)p\pi}{2N}\right] \cos\left[\frac{(2j+1)q\pi}{2N}\right] \quad [3]$$

where Ep,Eq is given by Equation(3)

$$E_x = \begin{cases} 1/\sqrt{2}, & x = 0 \\ 1, & x \neq 0 \end{cases}$$

Using the separability property of DCT the 2-D can be performed by two separate 1-D process.The 1-D DCT is given by equation (5)

$$Zk = \frac{c(k)}{2} \sum_{i=0}^{N-1} xi \cos \frac{(2i+1)k\pi}{2N} \quad [4]$$

where k=0,1,2,3--------N-1 and c(k) is given by equation (4)

$$c(k) = \begin{cases} 1/2 & k=0 \\ 1 & \text{otherwise} \end{cases}$$

The general block diagram for a 2-D DCT(8x8) architecture using row column decomposition method is shown in Figure 3.1. The architecture presented below uses row column decomposition method for 2-D DCT computation.

$$Y = C.X^T \qquad (5)$$

where X is the row matrix, C is the matrix of DCT Coefficients, and X^T is the Transpose of X
The even 1-D IDCT matrix is given by

$$\begin{bmatrix} x0+x7 \\ x1+x6 \\ x2+x5 \\ x3+x4 \end{bmatrix} = \begin{bmatrix} 0.7033 & 0.9271 & 0.7112 & 0.3835 \\ 0.7120 & 0.3835 & -0.7112 & -0.9271 \\ 0.7104 & -0.3835 & -0.7112 & 0.9271 \\ 0.7070 & -0.9271 & 0.7112 & -0.3835 \end{bmatrix} \begin{bmatrix} z0 \\ z2 \\ z4 \\ z6 \end{bmatrix} \qquad (14$$

The real valued inverse cosine basis are represented in 7-bit binary format and pre scaled by a factor of 2^7. The spatial data sequences obtained are integers and should be post scaled by the same factor. As in forward DCT, the multiplication is transformed into shifts and adds and pre-computing is also done for achieving hardware reduction.

C) 1-D DCT Architecture Using Pre computing Units

The 1-D DCT given by equation (5) can be split into two matrixes, the odd and the even.

The odd 1-D DCT can be expressed as

$$\begin{bmatrix}z_1\\z_3\\z_5\\z_7\end{bmatrix} = (x_0-x_7)\begin{bmatrix}a\\c\\e\\g\end{bmatrix}+(x_1-x_6)\begin{bmatrix}c\\-g\\-a\\-e\end{bmatrix} \qquad [6]$$

$$+(x_2-x_5)\begin{bmatrix}e\\-a\\g\\c\end{bmatrix}+(x_3-x_4)\begin{bmatrix}g\\-e\\c\\-a\end{bmatrix}.$$

The even 1-D DCT can be expressed as

$$\begin{bmatrix}z_0\\z_2\\z_4\\z_6\end{bmatrix} = (x_0+x_7)\begin{bmatrix}d\\b\\d\\f\end{bmatrix}+(x_1+x_6)\begin{bmatrix}d\\f\\-d\\-b\end{bmatrix} \qquad [7]$$

$$+(x_2+x_5)\begin{bmatrix}d\\-f\\-d\\b\end{bmatrix}+(x_3+x_4)\begin{bmatrix}d\\-b\\d\\-f\end{bmatrix}$$

where ck = cos kπ/16 , a = c1, b = c2, c = c3, d = c4, e = c5,
f = c6, g = c7 are the cosine basis.

The external memory organization is shown in the right part of Fig. 3. There are two memory banks for reference pixel storage: PR bank and TR bank, and one memory bank for address table (TLB) storage. Two types of addressing method are used in our design: TLB addressing and fixed addressing. For fixed length data, such as TR and TLB data, it can be addressed by their base address and block index. For PR data, the data size is variable. To continuously load and store data, PR data is byte aligned, and the TLB is used to record start address and length of each 8 × 8 block.
shifting operations.

D) DTT Approximation and Fast Algorithm

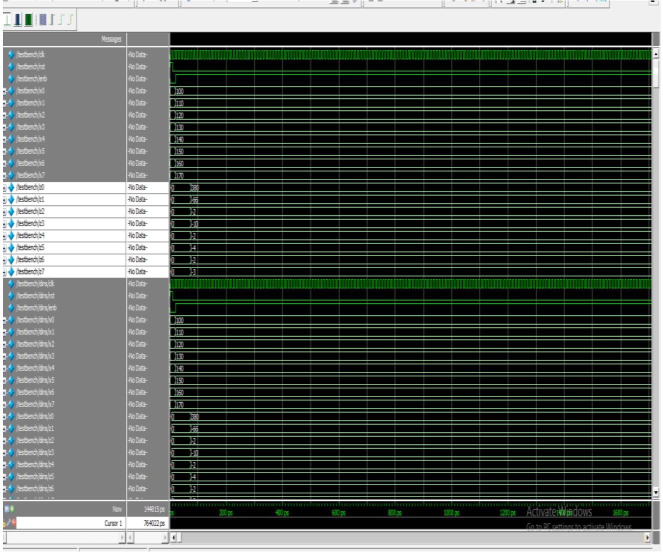
Here we proposed new approach to obtain an 8-point DTT approximation. The scale-and-round approach is particularly effective when discrete trigonometric transforms are considered. This is because the entries of such transformation matrices have smaller dynamic ranges when compared to the DTT. In contrast, the DTT entries have values with a dynamic range roughly seven times larger than the DCT, for example. Thus the approximation error implied by the round function is less evenly distributed in non-trigonometric transform matrices, such as the DTT. we introduce a parametric family of approximate DTT matrices , which are given by:

$$\mathbf{T}(\alpha) = \text{round}\left(\alpha \cdot \mathbf{T} \cdot \mathbf{D}_0\right),$$

We aim at identifying a particular optimal parameter T such that results in a matrix satisfying the following constraints: (i)T0 entries must be defined over T and(ii) T* must possess low arithmetic complexity.

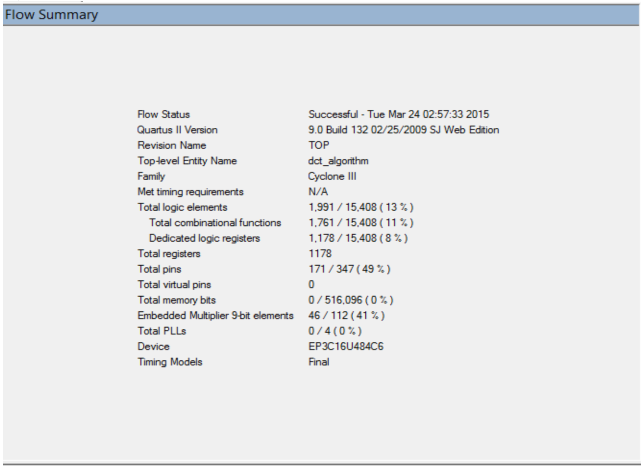
IV. SOFTWARE IMPLEMENTATION RESULTS

A) Modelsim output



```
Messages
/testbench/clk          -No Data-
/testbench/rst          -No Data-
/testbench/enb          -No Data-
/testbench/x0           -No Data-   100
/testbench/x1           -No Data-   110
/testbench/x2           -No Data-   120
/testbench/x3           -No Data-   130
/testbench/x4           -No Data-   140
/testbench/x5           -No Data-   150
/testbench/x6           -No Data-   160
/testbench/x7           -No Data-   170
/testbench/z0           -No Data-   0   380
/testbench/z1           -No Data-   0   -66
/testbench/z2           -No Data-   0   -2
/testbench/z3           -No Data-   0   -10
/testbench/z4           -No Data-   0   -2
/testbench/z5           -No Data-   0   -4
/testbench/z6           -No Data-   0   -2
/testbench/z7           -No Data-   0   -3
/testbench/dins/clk     -No Data-
/testbench/dins/rst     -No Data-
/testbench/dins/enb     -No Data-
/testbench/dins/x0      -No Data-   100
/testbench/dins/x1      -No Data-   110
/testbench/dins/x2      -No Data-   120
/testbench/dins/x3      -No Data-   130
/testbench/dins/x4      -No Data-   140
/testbench/dins/x5      -No Data-   150
/testbench/dins/x6      -No Data-   160
/testbench/dins/x7      -No Data-   170
/testbench/dins/z0      -No Data-   0   380
/testbench/dins/z1      -No Data-   0   -66
/testbench/dins/z2      -No Data-   0   -2
/testbench/dins/z3      -No Data-   0   -10
/testbench/dins/z4      -No Data-   0   -2
/testbench/dins/z5      -No Data-   0   -4
/testbench/dins/z6      -No Data-   0   -2
Now        144815 ps    ps  200 ps  400 ps  600 ps  800 ps  1000 ps  1200 ps  1400 ps  1600 ps
Cursor 1   764022 ps
```

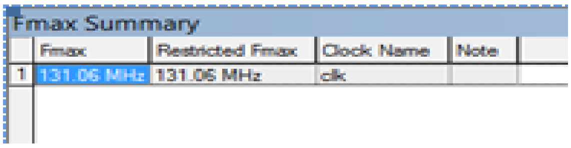
B) Area utilization



Flow Summary

Flow Status	Successful - Tue Mar 24 02:57:33 2015
Quartus II Version	9.0 Build 132 02/25/2009 SJ Web Edition
Revision Name	TOP
Top-level Entity Name	dct_algorithm
Family	Cyclone III
Met timing requirements	N/A
Total logic elements	1,991 / 15,408 (13 %)
Total combinational functions	1,761 / 15,408 (11 %)
Dedicated logic registers	1,178 / 15,408 (8 %)
Total registers	1178
Total pins	171 / 347 (49 %)
Total virtual pins	0
Total memory bits	0 / 516,096 (0 %)
Embedded Multiplier 9-bit elements	46 / 112 (41 %)
Total PLLs	0 / 4 (0 %)
Device	EP3C16U484C6
Timing Models	Final

***C)** Fmax Output*



Fmax Summary

	Fmax	Restricted Fmax	Clock Name	Note
1	131.06 MHz	131.06 MHz	clk	

V. TABLE

Table 5.1

METHOD	Multipliers Used	Logical elements Used
DCT method	32	1,991 LE's
DTT method	NIL	1,752 LE's

VI. CONCLUSION

In this paper, we analyze the performance of DCT and DTT algorithm for prediction based video compression system. Initially we analyzes the DCT   and its funtionlaity   using MODELSIM.The proposed algorithm is based on MC to finely save external memory bandwidth. We also illustrated the performance of DTT algorithm in numerical simulations, and our algorithm shows a significant performance improvement compared to DCT, while the complexity   is much lower compared to DCT based approach.

REFERENCES

[1]. B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, Y.-K. Wang, and T. Wiegand, High Efficiency Video Coding (HEVC) Text Specification Draft 10, document Rec. JCTVC-L1003, 2013.

[2]. L.-M. Po and W.-C. Ma, “A novel four-step search algorithm for fast block motion estimation,” IEEE Trans. Circuits Syst. Video Technol., vol. 6, no. 3, pp. 313–317, Jun. 1996.

[3]. JVT of ISO/IEC MPEG, ITU-T VCEG, MVC Software Reference Manual-JMVC 8.2, document Rec. JVT-B118r2, May 2010.

[4]. J.-C. Tuan, T.-S. Chang, and C.-W. Jen, “On the data reuse and memory bandwidth analysis for full-search block-matching VLSI architecture,” IEEE Trans. Circuits Syst. Video Technol., vol. 12, no. 1, pp. 61–72, Jan. 2002.