Perspectives in Communication, Embedded-Systems and Signal-Processing (PiCES) – An International Journal
ISSN: 2566-932X, Vol. 1, Issue 11, February 2018
Proceedings of National Conference on Emerging Trends in VLSI, Embedded and Networking (NC-EVEN 17), May 2017

Architecture of algorithm conceptualised and reconfigurable for an efficacious and mutable orthogonal type approximation of DCT

Chethana S
4th sem M-tech, RNSIT, Bangalore,
chethanashivanand18@gmail.com

Srikanth G N
Associate Professor EIE Dept, RNSIT, Bangalore

Abstract:** Approximation of the discrete cosine type transform can be considered beneficial when debasing its estimation complicatedness without much of the impact on its performance of coding. Most of the earlier worked algorithms for the estimation of the DCT targets only the DCT of small lengths of transform and few of them identified as non-orthogonal. In the suggested design, the respective inputs are either video or the image, which of size of 256x256 where each pixel being eight bits. The coefficient or txt file of the respective input image or regulation of the video considering MATLAB is worked. Suggested design presents a regularised recursive type algorithm to procure the orthogonal estimation of the respective DCT where an approximate DCT as to the length can be worked from DCTs of length, especially at the cost incurred at the summation of the input at preprocessing. We do perform the decomposition which is iterative matrix of spart type and considers forth of those symmetrices of basis DCT type vectors for arriving at the suggested algorithm of approximation. Suggested algorithm can be considered highly changeable for the chosen hardware as well those of software at an enactment of DCT particularly of the greater length and it does makes the utilisation of the earlier approximation respectively of the eight point DCT to procure the approximate DCT of any power with two lengths where N being greater than eight and even such approximation of DCT does provides comparable or the finer image and even compression of the video at the performance than those of the earlier method. . It is exhibited that the suggested algorithm comprises of the lower complicatedness in the arithmetic when is compared with the other existing algorithms of approximation. We have exhibited absolutely scalable reconfigurable parallel type architecture for the guesstimation of the respective approximation DCT reckoned, which is relied on the suggested algorithm. One uniquely zeal promoting feature of the suggested design is that it can be configurable for the evaluation of thirty two point DCT or for all the parallel type guesstimation of the two of sixteen point DCTs or the four of eight point DCTs with the marginal overhead administering. The suggested architecture is found primarily to offer many of the benefits in terms of the respective hardware as the complexity, regularity and also the modularity can be known. The output can be procured fundamentally from the respective approximation of the DCT transformed back to an image or video considering the MATLAB. The whole of the design has been performed utilising the verilog HDL and tool of Xilinx and tool of MATLAB.

***Keywords:** Algorithm-architecture codesign; DCT approximation; discrete cosine transform (DCT); high efficiency video coding (HEVC).*

I. Introduction

The DISCRETE cosine transform (DCT) is popularly used in image and video compression. Since the DCT is computationally intensive, several algorithms have been proposed in the literature to compute it efficiently.

The main objective of the approximation algorithms is to get rid of multiplications which consume most of the power and computation time, and to obtain meaningful estimation of DCT as well. In this paper, we propose an algorithm to derive approximate form of DCTs which satisfy all the three features. We obtain the proposed approximate form of DCT by recursive decomposition of sparse DCT matrix. It is observed that proposed algorithm involves less arithmetic complexity than the existing DCT approximation algorithms. The proposed approximate form of DCT of different lengths is orthogonal, and result in lower error-energy compared to the existing algorithms for DCT approximation. The decomposition process allows generalization of the proposed transform for higher-size DCTs. Interestingly, proposed algorithm is easily scalable for hardware as well as software implementation of DCT of higher lengths, and it can make use of the best of the existing approximations of 8-point DCT.

Based on the proposed algorithm, we have proposed a fully scalable, reconfigurable, and parallel architecture for approximate DCT computation. One uniquely interesting feature of proposed design is that the structure for the computation of 32-point DCT could be configured for parallel computation of two 16-point DCTs or four 8-

point DCTs. The proposed algorithm is found to be better than the existing methods in terms of energy compaction and hardware complexity, as well.

A scheme of approximation of DCT should have the following features:

i. It should have low computational complexity.

ii. It should have low error energy in order to provide compression performance close to the exact DCT, and preferably should be orthogonal.

iii. It should work for higher lengths of DCT to support modern video coding standards, and other applications like tracking, surveillance, and simultaneous compression and encryption.

In this section, we take a brief review of the works that is related to the domain of DCT. The proposed transformation matrix contains only zeros and ones; multiplications and bit-shift operations are absent[1]. Haweel [3] has proposed the signed DCT (SDCT) for 8* 8 blocks where the basis vector elements are replaced by their sign, i.e,± 1. modern video coding standards such as high efficiency video coding (HEVC) [2] uses DCT of larger block sizes (up to 32* 32) in order to achieve higher compression ratio.

II. OBJECTIVE OF THE PROJECT & DESCRIPTION

The Discrete cosine transform (DCT) is popularly used in image and video compression. Since the DCT is computationally intensive, several algorithms have been proposed in the literature to compute it efficiently. Recently, significant work has been done to derive approximate of 8-point DCT for reducing the computational complexity.

The main objective of the approximation algorithms is to get rid of multiplications which consume most of the power and computation- time, and to obtain meaningful estimation of DCT.

1) Existing DCT Algorithms:

Existing DCT algorithms do not provide the best of all the above three requirements. Some of the existing methods are deficient in terms of scalability, generalization for higher sizes, and orthogonality.

III. PROPOSED DESIGN

DCT matrix C_N can be represented by the following matrix product

$$C_N = \frac{1}{\sqrt{2}} M_N^{per} T_N M_N^{add} \quad (1)$$

Where T_N is a block sparse matrix expressed by

$$T_N = \begin{bmatrix} C_{\frac{N}{2}} & 0_{\frac{N}{2}} \\ 0_{\frac{N}{2}} & S_{\frac{N}{2}} \end{bmatrix} \quad (2)$$

Where $0_{N/2}$ is the ((N/2) x (N/2)) zero matrix. Block sub matrix $S_{N/2}$ consists of odd rows of the first N/2 columns of.$\sqrt{2}C_N.M_N^{per}$ is a permutation matrix expressed by:

$$M_N^{per} = \begin{bmatrix} P_{N-1,\frac{N}{2}} & 0_{1,\frac{N}{2}} \\ 0_{1,\frac{N}{2}} & P_{N-1,\frac{N}{2}} \end{bmatrix} \quad (3)$$

Where $0_{1,N/2}$ is a row of N/2 zeros and $P_{N-1,N/2}$ is a (N-1) × (N/2) matrix defined by its row vectors as:

$$P^{(i)}_{N-1,\frac{N}{2}} = \begin{cases} 0_{1,\frac{N}{2}} & if \quad i = 1,3,5,\ldots,N-1 \\ I_{\frac{N}{2}}\left(\frac{i}{2}\right) & if \ i = 0,2,4,\ldots,N-2 \end{cases} \quad (4)$$

Where $I_{N/2(i/2)}$ is the (i /2) th row vector of the ((N/2) x (N/2)) identity matrix. Finally, the last matrix M_N^{add} is defined by:

$$M_N^{add} = \begin{bmatrix} I_{\frac{N}{2}} & J_{\frac{N}{2}} \\ I_{\frac{N}{2}} & -J_{\frac{N}{2}} \end{bmatrix} \quad (5)$$

where $J_{N/2}$ is an ((N/2) x (N/2)) matrix having all ones on the anti-diagonal and zeros elsewhere . To reduce the computational complexity of DCT, the computational cost of matrices presented (1) is required to be assessed. Since M_N^{per} does not involve any arithmetic or logic operation, and M_N^{add} requires N/2 additions and N/2 subtractions, they contribute very little to the total arithmetic complexity and cannot be reduced further. Therefore, for reducing the computational complexity of -point DCT, we need to approximate T_N.

Let $\hat{\hat{C}}_{N/2}$ and $\hat{\hat{S}}_{N/2}$ denote the approximation matrices of $C_{N/2}$ and $S_{S/2}$ respectively. To find these approximated submatrices we take the smallest size of DCTmatrix to terminate the approximation procedure to 8, since 4-point DCT and 2-pointDCT can be implemented by adders only.

Consequently, a good approximation of C_N , where N is an integral power of two, for N ≥ 8, leads to a proper approximations of $C_8$and S_8 . For approximation of C_8 we can choose the 8-point DCT since that presents the best trade-off between the number of required arithmetic operators and quality of the reconstructed image. The trade-off analysis shows that approximating C_8 by $\hat{\hat{C}}_8$=⌊2 .$\hat{C}_8$] where [.] denotes the rounding-off operation outperforms the current state-of-the-art of 8-point approximation methods.

When we closely look at (1) and (2), we note that C_8 operates on sums of pixel pairs while S_8 operates on differences of the same pixel pairs. Therefore, if we replace Let $\hat{\hat{C}}_8$ and $\hat{\hat{S}}_8$ denote we shall have two main advantages. Firstly, we shall have good compression performance due to the efficiency of $\hat{\hat{C}}_8$ and secondly the implementation will be much simpler, scalable and reconfigurable.

IV. SCALABLE AND RECONFIGURABLE ARCHITECTURE FOR DCT COMPUTATION

In this section, we discuss the proposed scalable architecture for the computation of approximate DCT of

N=8, 16 and 32 and discuss in the reconfiguration scheme.

A. Proposed Scalable Design

The basic computational block of algorithm for the proposed DCT approximation, $\hat{\hat{C}}_8$ is given in [3]. The block diagram of the computation of DCT based on $\hat{\hat{C}}_8$ is shown in Fig. 1. For a given input sequence{X(n)},n€[0,N-1], the approximate DCT coefficients are obtained by F= $\hat{\hat{C}}_N$. X^T.

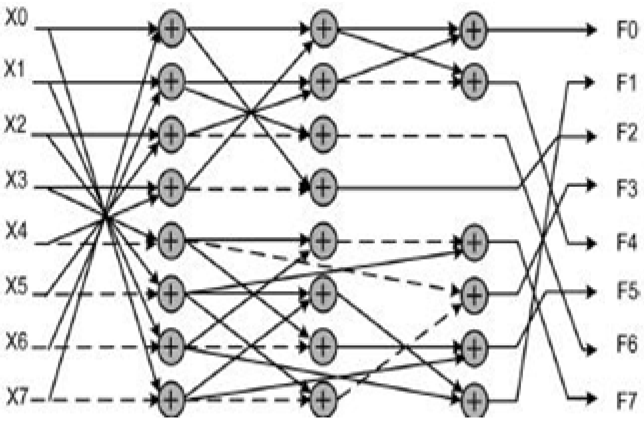

Fig 1. Signal flow graph (SFG) of($\hat{\hat{C}}_8$).Dashed arrow represent multiplications by -1

A reconfigurable design for the computation of 32-, 16-, and 8-point DCTs is presented in Fig. 4. It performs the calculation of a 32-point DCT or two 16-point DCTs in parallel or four 8-point DCTs in parallel. The architecture is composed of 32-point input adder unit, two 16-point input adder units, and four 8-point DCT units. The reconfigurability is achieved by three control blocks composed of (64) 2:1 MUXes along with (30 3:1 MUXes. The first control block decides whether the DCT size is of 32 or lower. If Sel 32=1, the selection of input data is done for the 32-point DCT, otherwise, for the DCTs of lower lengths. The second control block decides whether the DCT size is higher than 8. The output permutation unit which re-orders the output depending on the size of the selected DCT. Sel 32 and Sel 16 are used as control signals to the 3:1MUXes. Specifically, for {Sel 32,Sel 16}2equal to {00}, {01} or {11} the 32 outputs correspond to four 8 point parallel DCTs, two parallel 16-point DCTs, or 32-point DCT, respectively. Note that the throughput is of 32 DCT coefficients per cycle irrespective of the desired transform size.

Application:

Image Compression

Video Compression

B. Block Diagram:

In this block diagram we can see the conversation image to hex and then hex to image converter.

In this first convert the picture in to .jpeg format and then in the MATLAB we convert image to hex (we write the code in matlab and executed it) so in that we got the input value of it. And then we move on to Xilinx or MODELSIM so in this we stimulate the program and take the output of it again we go back to MATLAB in that we can convert it in to hex to image. And in this we reconstructed the image in N=8,16,32.as shown below.

C. Image Compression:

The color image is converted to gray scale image with pixel values ranging from 0 to 255. The value 0 and 255 represents black and white of an image. It is applied along the row then column of the image.

Fig 2. Original image

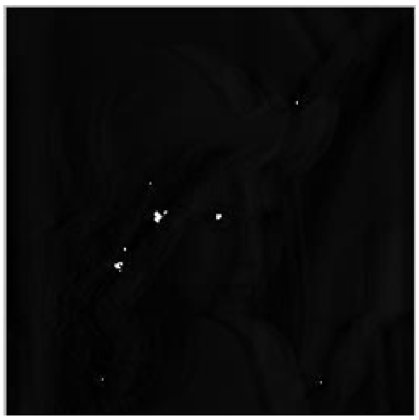

Fig 3. Reconstructed of N=8, 16, 32

D. Video compression:

To evaluate the performance of the proposed algorithm for video coding we have integrated the proposed approximated DCT into HEVC reference software HM12.1, in the same way as has been done in [4]. Moreover, we have integrated the existing methods to have a comparison in real time video coding.

I am now started the video compression, so i am not completed it.

V. CONCLUSION

In this paper, we have proposed a algorithm to obtain orthogonal approximation of DCT where approximate DCT of length N could be derived from a pair of DCTs of length (N/2). The proposed approximated DCT has several advantages, such as of regularity, lower-computational complexity, structural simplicity, and scalability. Comparison with recently proposed competing methods shows the effectiveness of the proposed approximation in terms of error energy, hardware resources consumption, and compressed image quality. We have also proposed a fully scalable reconfigurable architecture for approximate DCT computation where the computation of 32-point DCT could be configured for parallel computation of two 16-point DCTs or four 8-point DCTs.

REFERENCES

[1] R. J. Cintra and F. M. Bayer, "A DCT approximation for image compression," IEEE Signal Process. Lett., vol. 18, no. 10, pp. 579–582, Oct. 2011.

[2] G. J. Sullivan, J.-R. Ohm,W.-J.Han, and T.Wiegand, "Overview of the high efficiency video coding (HEVC) standard," IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[3] U. S. Potluri, A.Madanayake, R. J. Cintra, F.M. Bayer, S. Kulasekera, andA. Edirisuriya, "Improved 8-point approximate DCT for image and video compression requiring only 14 additions," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 61, no. 6, pp. 1727–1740, Jun. 2014.