Design And Implementation Of Iterative DCT Using Concurrent Loading For Image Encoding

Dr. D Jessintha*, P Aishwaryalakshmi2, UDivya3, H Gayathri4
1-4Department of Electronics and Communication Engineering, Easwari Engineering College, Ramapuram, Chennai, India.

Corresponding Author: jessintha@eec.srmrmp.edu.in

Abstract: Discrete Cosine Transform (DCT) is the most widely used transform for signal compression in many image and video processing applications. Spatial signals are transformed to frequency domain in which the signal is broken down into different frequency components through DCT. DCT is generally used for image compression but also for noise reduction in images. But various digital architectures developed for computing DCT for images require large computations resulting in a complex and costly design. So a design of an advanced model of parallel DCT with iterative looping structures is proposed to process images. The benefit of such 8-point iterative DCT architecture is that it improves the resolution of the input image thereby utilizing low power principle and finally transforms the image data with reduced errors. The iterative DCT used in this system results in reduced processing time. The digital architecture used for the image processing follows distributed arithmetic algorithm. As a result, the proposed system achieves an area reduction of 47% and an energy reduction of 65.55%. Keywords: Compression, Discrete Cosine Transform (DCT), distributed arithmetic, noise reduction, resolution.

1. Introduction
Discrete Cosine Transform (DCT) is a processing technique of the signal into various frequency components. It is mostly employed in image compression modalities like JPEG compression. DCT modifies the picture element of an image into its corresponding frequency component values. JPEG process is the one which is a broadly used type for lossy image compression that targets DCT. The lossy image compression is utilized in digital images to increase the storage space [1]. DCT divides the image into various frequency sections. Another major feature of the DCT is the ability to quantify the DCT coefficients using weighted quantization values. During the quantization stage, the lower range of frequencies is discarded and only the dominant frequencies are used to get the original image [2,4]. The need for image compression is to compress the size of the image and to keep most of the information incorporated within the original image. DCT based coding/decoding systems play a crucial role in real-time applications in the proposed system; the design of an advanced model of parallel DCT with iterative looping structure is used. The design of the architecture includes an algorithm which is of distributed arithmetic in nature to achieve very high speed with low power. Also the architecture aims to achieve high accuracy with efficient hardware realization. Colored Objects of an image is identified by its hue, saturation and brightness [9]. The output obtained for DCT operation improve the pixel values to round off the quantized value and applied to the filter. The iteration is done to adjust the weights of the coefficients especially to remove the noise in the image. The iterative DCT used in this system results in reduced processing time in
computing and also the processed image is of high resolution. In this architecture low power design uses parallel processing units that allow power savings from depletion in clock speed.

2. Existing System

In the existing system an approximated DCT is applied to the image for encoding it. The approximation is done in order to reduce the hardware cost. To have a good coding efficiency the approximation is done without changing the transform matrix. The main methods used are:

- LSB Truncation (LT)
- MSB Truncation (MT)
- Zero-Column Truncation (ZCT)

According to Chen’s algorithm for 32-point DCT [3], it can be splitted into three stages as shown in Fig.1.

![Fig. 1. Chen’s Algorithm for 32-point DCT](image)

A butterfly structure is used which divides the input terms into odd and even parts where each has 16 inputs and outputs. Even part is further divided into even-even (EE) and even-odd (EO) parts and the results are permuted. The EO and EE parts do 64 multiplications so the total even part does 128 multiplications and the odd part does 256 multiplications to produce 16 outcomes. It aims to minimize the hardware cost that occurs from the odd part and the EO part as it is similar to the odd part.

2.1 LSB Truncation

In the odd part two main operations take place: Sum of product (SOP) and right-shift. Most of the LSB (Least Significant Bit) operations of SOP do not have much effect on the final result after the right shift operation. So LSB truncation is implemented. In the SOP, operation done in the Odd part each product can be written as in Equation (1).

$$Q_j = S_j Y_j$$  \hspace{1cm} (1)

where $Y_j$ are the inputs and $S_j$ equals transform coefficients.

In Equation (2), the final output (summation) can be achieved where $M$ is the number of products.

$$\sum_{j=0}^{M-1} S_j Y_j = \sum_{j=0}^{M-1} Q_j$$  \hspace{1cm} (2)

After truncating LSBs, an accurate product term $Q_j$ will be approximated as the term $\hat{Q}_j$. Hence, the final result can be given by Equation (3)

$$\hat{Q} = \sum_{j=0}^{M-1} \hat{Q}_j$$  \hspace{1cm} (3)

The truncation error and the overall errors are described in Equation (4) and (5) respectively as follows:

$$E_{\text{trun}} = \sum_{j=0}^{M-1} (Q_j - \hat{Q}_j) \leq 2^{\log_2^2 M - p1}$$  \hspace{1cm} (4)

where, $p1$ is the precision for each product term and

$$E = E_{\text{trun}} + E_{\text{round}} \leq 2^{\log_2^2 M - p1} + 2^{-p2-1}$$  \hspace{1cm} (5)
$p_2$ is the precision for final output and $E_{\text{round}}$ is the rounding error.

### 2.2 MSB Truncation

The wide set of high frequency elements for the conversion is generally small and so truncation of most significant bits (MSBs) is done. DCT results follow Gaussian Central Limit Theorem distribution. The variance of the pixels in the block is proportional to the variance of the distribution and mean equals zero. The DCT result $G(w, z)$ is given in the probability density function (pdf) to change in the variance of pixels in a block as follows in the equation (6):

$$p(G_{w,z}) = \int_0^\infty p(G_{w,z}|\sigma^2) p(\sigma^2) d(\sigma^2)$$

Where $p(G_{w,z}|\sigma^2)$ follows Gaussian distribution as given in Equation (7). However the variance of pixels is noted as exponential distribution form[9] given in Equation (8).

$$p(G_{w,z}|\sigma^2) = \frac{1}{\sqrt{2\pi}} e^{-\frac{G_{w,z}^2}{2\sigma^2}}$$

$$p(\sigma^2) = \lambda e^{-\lambda}$$

On substituting (7) and Equation (8) in Equation (6), it gives the result that laplacian distribution followed by the variance of pixels as given by Equation (9).

$$p(G_{w,z}) = \frac{\sqrt{2\pi}}{2} e^{-\sqrt{2\lambda|G_{w,z}|}}$$

Since the probability of small values are greater than the large values in Laplacian Distribution, the MSB truncation can be done. But the coefficient of index 0 is the lowest frequency component in the Odd part so that the probability of the value is to be high. To inhibit from the loss of performance, the elements at index 0 does not undergo MSB truncation.

### 2.3 Zero Column Truncation

After encoding, quantization is done where some high-frequency coefficients become zero resulting in zero columns which can be removed to reduce the computation power consumption. Thus a less expensive approximate DCT architecture is developed with maximum of 6 LSBs truncation with MSB truncation done with respect to the index number (except 0) and the zero values are skipped for further processing.

### 3. Proposed System

In the proposed system, an advanced model of 32-point concurrent DCT is employed to remove noise from the image as well as compression is done. The concurrent architecture is developed using distributed arithmetic algorithm thereby introducing an iterative scheme to adjust the weight of the DCT coefficients to make DCT act as a noise filter.

### 3.1 Phases in the Proposed System

- **Step I:** The input image is vectorised into matrix. A specified type of image noise is added to it.
- **Step II:** Then the concurrent DCT architecture is established through VHDL (Verilog Hardware Definition Language) coding. The iterative scheme continues to apply DCT to every pixel in the input matrix and continues the process for the next inputs until it is stopped by the programmer.
- **Step III:** The cosine transform matrix is calculated by distributed arithmetic algorithm. This results to obtain the output with reduced processing time. This process is represented in Fig. 2.
This type of architecture does not involve approximation of Discrete Cosine Transform (DCT) and does not require the complex method of bit truncation. Moreover, the transformed output has less errors when compared to the approximated DCT output. In addition to that, the area and power consumption is also nominal. The input image is vectorised using MATLAB software and the Digital architecture is developed.

The three main techniques used in our system are:
- Iterative DCT
- Distributed Arithmetic (DA) algorithm
- Concurrent Loading Architecture

3.2 Iterative DCT
DCT helps in breaking in the image into pixels by mainly focusing on the image quality. Commonly DCT is similar to DFT. DFT focuses on transforming signal, whereas DCT focuses on transforming the images to get a better resolution. Iteration is the repetition process of data which helps in generating series of outcomes. All the process is a single iteration and the output of one sequence is the input of another sequence. This process is repeated until the required result is obtained. It is a computational process. An iterative process should be convergent. Iteration in DCT image compression helps in getting more accurate pixel of the image. It is the method of computing images through iterative technique. It represents data in a cosine form. The transformation is orthogonal and energy is preserved. Also the algorithm processing rate is faster compared to other techniques. It uses real computations and it’s hardware is easier to understand.

3.3 Distributed Arithmetic Algorithm
Discrete Cosine Transform (DCT) used in image and video processing needs a huge amount of data transmission. Mostly in all works, 8x8 2-Dimensional DCT is performed for image compression. DCT is an operation which requires many number and stages of arithmetic computation. Many adders and multipliers are required to implement the hardware transformation. Multiplier blocks consume more power. Distributed Arithmetic (DA) is also used to perform a multiplicator free operation [5]. DA is used in the proposed architecture for low hardware and energy consumption. On computing DCT on an image a data array is created that can be compressed by data compaction algorithms. Thus the resultant data is stored in a compacted form. The stored image has its quality dependent on the quantization levels used in the compaction algorithm. On reconstructing the original image the data stored in the memory is retrieved and inverse DCT is performed on it. Distributed Algorithm is a rearrangement of a multiply accumulate to mask the multiplications and implements this at bit level. This is a very effective approach to minimize the size of a multiplying parallel hardware that is suitable for FPGA designs. DA is extended to be used for complex multiplications of Fourier transforms. DA acts as an effective technique to calculate inner products. Look up Tables (LUTs) and accumulators are used instead of multipliers. Since DA replaces the
multiplication blocks with adders and it has wide range of application in Very large Scale Integration (VLSI) implementations of many Signal Processing (DSP) applications. DA simplifies the intensive arithmetic operations with multiply/accumulate (MAC) unit which is being the predominant operation in DCT. DA has many advantages such as computational efficiency, multiplier less architecture, less utilization of arithmetic computing resources. The significant feature of DA technique is that it gives an assumption of using both adders and multipliers for the DCT operation. Designs with DA can be implemented with standard available ICs.

3.4 Concurrent Loading Architecture
A modern VLSI architecture for N multiplied by N Discrete Transform Cosine (DCT). This architecture uses the advantages of a bit serial architecture, which is bit parallel and distributed. The architecture that resulted contains only memories, adders and registers. No multipliers are used. It is also most suitable for VLSI implementation. With the proposed architecture, the design uses a concurrent architecture that has distributed arithmetic to achieve high performance with low power.

4. Results
The system is evaluated for the error between the input pixels and the transformed output. It is observed that the error remains negative which illustrates that the transformed output for each pixel has higher value than the input pixel.

The area consumption is obtained from the macro cells and pin usage is 50% lesser than the original DCT and the power utilized in this system is 23.97mW which is 64.95% lesser than the existing power consumption of 68.4mW. The existing system for approximated 32-point DCT the power consumption and the area consumption is compared to the original DCT system as given by Table 1. In comparison with the existing model the proposed system gives efficiency in the means of area at higher rate as shown in Fig. 3. The Table 2 shows the power reduction among the various designs of the system.

| Technique used          | Area Consumption (no. of gates used) | Power Consumption(mW) |
|-------------------------|-------------------------------------|-----------------------|
| Original DCT [8]        | 167                                  | 68.4                  |
| Existing System (approximate DCT) [8] | 103                                 | 26.95                 |
| Proposed System         | 86                                   | 24.97                 |

| Design         | Power Reduction (%) |
|----------------|---------------------|
| [6]            | -23.91              |
| [7]            | -31.6               |
| Existing [8]   | -60.59              |
| Proposed       | -65.55              |
5. Conclusion

Thus the proposed model of DCT operations used MATLAB for preprocessing the image pixels and the architecture is implemented using HDL coding. At first the input image is vectorised and then 8-point DCT is computed for the vectorised image resulting in image compression. Noise reduction has been done and so in addition to that it provides enhanced output image which can be inferred from the error determined between the input and output image. Moreover, the proposed architecture achieves a low power design with the other existing architectures and also a reconfigurable structure. With further improvement, the architecture can be modified with various frequencies of operation and to the next level. With the designed architecture, 46.9% area reduction and 65.55% power minimization are achieved with the same accuracy level.

References

[1] Anitha S "Image Compression Using Discrete Cosine Transform & Discrete Wavelet Transform" International Journal of Scientific & Engineering Research Volume 2 Issue 8 August-2011

[2] Chamman Lal Sahu Chandra Shekhar Singh Thakur Manoj Kumar Xalxo Mohit Thakur Miss Roshti Rathour “Survey on JPEG Image Compression Using DCT” Volume 4 Issue X October 2016 IC Value: 1398 ISSN: 2321-9653

[3] W-H Chen C H Smith and S Fralick “A fast computational algorithm for the discrete cosine transform” IEEE Trans Commun vol COM-25 no 9 pp 1004–1009 Sep 1977

[4] Balamurali R and Chandrasekar A 2019 Multiple parameter algorithm approach for adult image identification Cluster Computing 22(5) pp11909-11917

[5] Venkata Vinetha Kasturi Y S Yamala International "VLSI Architecture For DCT Based On Distributed Arithmetic" Journal of Engineering Research & Technology (IJERT) Vol 2 Issue 5 May – 2013

[6] P K Meher S Y Park B K Mohanty K S Lim and C Yeo “Efficient integer DCT architectures for HEVC” IEEE Trans Circuits Syst Video Technol vol 24 no 1 pp 168–178 Jan 2014

[7] M Masera M Martina and G Masera “Adaptive approximated DCT architectures for HEVC” IEEE Trans Circuits Syst Video Technol vol 27 no 12 pp 2714–2725 Dec 2017

[8] Heming Sun Member Zhengxue Cheng Student Member Amir MasoudGharebaghi Shinji Kimura and Masahiro Fujita”Approximate DCT Design for Video Encoding Based on Novel Truncation Scheme” IEEE Transactions on circuits and systems–iDec2019

[9] E Y Lam and J W Goodman “A mathematical analysis of the DCT coefficient distributions for images” IEEE Trans Image Process vol 9 no 10 pp 1661–1666 Oct 2000
[10] NR Deepa N Nagarajan D PalanivelRajan R Balamurali 2019 July Image Processing System for Detecting and Recovering from Cotton Leaf Disease using The Android Application International Journal of Innovative Technology and Exploring Engineering (IJITEE) 8(9) pp 3366 – 3381