Available online at www.sciencedirect.com

ScienceDirect

Materials Today: Proceedings XX (2016) XXX–XXX

ELSEVIER

materialstoday: PROCEEDINGS

www.materialstoday.com/proceedings

PMME 2016

Image encoder architecture design using dual scan based DWT with vector quantization

Thirumaraiselvi Chandraraju[a], Sudhakar Radhakrishnan[b]

[a,]*Assistant Professor,* [b]*Professor and Head*

[a]*Department of ECE, Sri Krishna College of Engineering and Technology, Coimbatore. Tamilnadu, India- 641008*

[b]*Department of ECE, Dr Mahalingam College of Engineering and Technology, Pollachi, Coimbatore, Tamilnadu, India- 641021*

Abstract

Vector quantizer block requires additional hardware resources, it was demonstrated that the codebook generated by coding the residual block makes the image codec more robust. This efficient algorithm is mainly based on flipping technique. First, a three level DWT is performed on the original image resulting in ten separate sub bands. These sub bands are then vector quantized. To implement a hardware efficient and modular architecture with a very simple control path this proposed method is presented. To minimize the critical path to one multiplier delay and to achieve 100% hardware utilization efficiency, the serial operation is optimized using parallel computation in advance with pipeline operation of independent path. This algorithm uses only five transposition register and it is repeatable architecture. Data path can be reduced to six multipliers and eight adders by folding this architecture without affecting the critical path.

© 2016 Elsevier Ltd. All rights reserved.

Selection and Peer-review under responsibility of International Conference on Processing of Materials, Minerals and Energy (July 29th – 30th) 2016, Ongole, Andhra Pradesh, India.

Keywords: DWT, Dual scan, Lifting, Image coder, Vector quantizer, Image compression.

2214-7853 © 2016 Elsevier Ltd. All rights reserved.

Selection and Peer-review under responsibility of International Conference on Processing of Materials, Minerals and Energy (July 29th – 30th) 2016, Ongole, Andhra Pradesh, India.

1. Introduction

Data compression techniques will continue to be a design challenge for future communication systems and advanced multimedia applications. The data compression algorithms can be broadly divided into two categories: lossless and lossy. Generally, lossless data compression techniques are applied to text data or scientific data. The discrete wavelet transform (DWT) is being increasingly used for image coding. It is due to the fact that DWT contributes superior features such as progressive image transmission by resolution or by quality. In the recent works, lifting scheme is popularly used for DWT leads to speed up and a fewer computation compared to classical convolution method. Swelden [1] derived the lifting based DWT to reduce complex operations. Recently, numerous VLSI architectures for the 2D DWT have been proposed to meet the requirements of real time processing. The design of DWT in practical systems has few issues. First, the complexity of wavelet transform is many times higher than that of discrete cosine transform (DCT). Second, DWT needs additional memory for storing the computation results. Furthermore, for real time image compression, DWT has to process large amount of data at faster speeds. Software implementation of DWT compression offers flexibility for manipulation but it does not meet few timing constraints. Hardware design of DWT, however, also has some issues. The important difficulty among them is high cost of hardware implementation of DWT. This paper proposes high performance architecture for image compression which is based on the frequency domain representation. The proposed architecture is developed using VHDL and has been tested for various still images.

The structure of this paper is as follows: Section two sketch the Discrete wavelet transform ; Section three illustrate the Modified lifting DWT scheme , Section four the proposed structure and analyze the VQ structure; Section five the experiment result; Section six the conclusion and research direction in the future.

Discrete wavelet Transform

Classical DWT architectures are based on convolution method [2][3]. Then the next generations DWTs are based on lifting algorithms. Compared with convolution method, lifting based [4] architectures require lesser computation complexity and also requires less memory. Direct implementation of these algorithms to hardware leads to relatively long data path and less efficiency. Several architectures based on the lifting scheme have been developed. An efficient folded architecture with less hardware complexity has been discussed by Liu et al [5]. Mohanty et al.[6] have been proposed a generic convolution based DWT with increased throughput. The proposed structure does not involve frame memory and only uses line memory. The major drawback of this structure involves more multipliers. A modified lifting algorithm have been proposed by Zhang et al,[7] which recombines the intermediate results and reduces the number of pipelining stages. As a result, the number of registers are reduced to 18 without increasing the critical path. The main drawback of this system is, it could not extended for multiwavelet. An efficient fast 2D DWT architecture have been proposed by Xiong et al [8]. This fast architecture introduced low hardware control complexity. A dual mode lifting DWT architecture was developed by Wu et al [9]., provides low transpose memory with reduced speed.

2. Modified Lifting Discrete Wavelet transform

The mathematical design steps for the hardware implementation of 9/7 lifting based DWT are explained by the following equations. The input sequence x_i is splitted in to even and odd components known as s_i^0 and d_i^0 respectively. Then lifting steps are performed over the splitted components and produces the output parts such as s_i^n and d_i^n, where n=1, 2. Finally, through the normalization factors K_1 and K_2, the low pass and hirgh pass coefficients s_i and d_i are obtained.

(i) Splitting step

Odd component: $di0=x2i+1$ (1)

Even component: $si0=x2i$ (2)

(ii) Lifting step

First lifting step

Prediction: $di1= di0+\alpha\ x\ (si0+ si+10)$ (3)

Updation: $si1= si0+\beta\ x\ (di-11+di1)$ (4)

Second lifting step

Prediction: $di2= di1+\gamma\ x\ (si1+ si+11)$ (5)

Updation: $si2= si1+\delta\ x\ (di-12+di1)$ (6)

(iii) Scaling step

$di=K2\ x\ di2$ (7)

$si= K1\ x\ si2$ (8*)*

Here, the lifting coefficients and scaling coefficients are $\alpha\approx-(3/2)$, $\beta\approx-(1/6)$, $\gamma\approx(5/4)$, $\delta\approx(32/15)$ and $K_1\approx,K_2\approx1.419$. The direct mapping architecture of the prediction and updation equations increases the critical path delay due to one multiplication and two adder delay. To optimize the critical path delay of the lifting based hardware implementation by changing the coefficients in lifting formulas a modified algorithm is employed. The modified lifting algorithm simplifies the computation process and reduces the number of registers. Thus, arithmetical substitution merges the predictor and updater in to one equation which will ease the time delay problem as given in the following equations.

$si1= si0+ (\beta\ x\ di-10 +\beta\alpha\ x\ si-10 +\beta\alpha\ x\ si0) +(\beta\ x\ di0 + \beta\alpha\ x\ \ si0 +\beta\alpha\ x\ si+1)$ (9)

$si2 = si1+ (\delta/\beta\ x\ \beta\ di-1i +\delta\gamma\ x\ si-11)+(\alpha/\beta\ x\ \beta di1+\delta\gamma+si1+\delta\gamma\ \ x\ si-11)$ (10)

$di = K2/\delta\ x\ (\delta di2)$ (11)

$si = K1\ x\ si2$ (12)

3. Proposed Image coder architecture

The proposed image coder shown in Fig.1 consists of two modules, namely two dimensional DWT architectures and vector quantizer. Here, 2D DWT processor consists of two 1D DWT processors, known as row processor and column processor. According to the scanning procedure for image pixel, the image pixels are subdivided in to 2 x 2 blocks. Then for every 2 x2 block, simultaneous row and column processing is performed, which removes the necessity for transpose buffer. Fig. 2explains the data flow sequence for the 4x4 image block operation .In this, during the first clock cycle first row odd, even element and second row odd and even pixels were scanned and its approximate coefficients were calculated. For the second clock cycle first row, 3rd pixel, 4th pixel and second row, 3rd ,4th pixel were scanned and computes the detail coefficients.

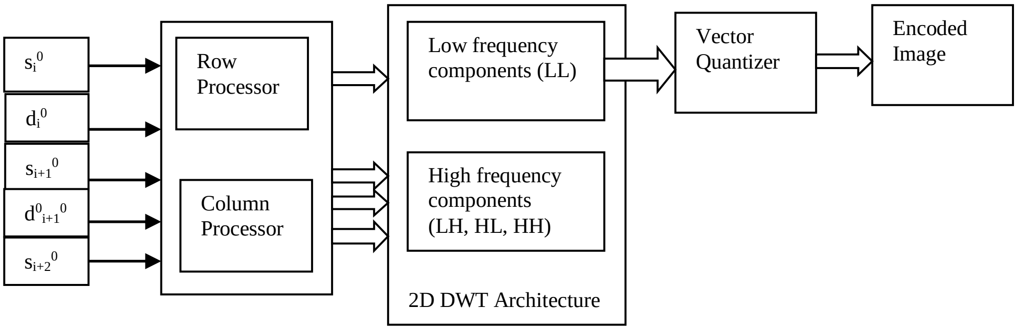
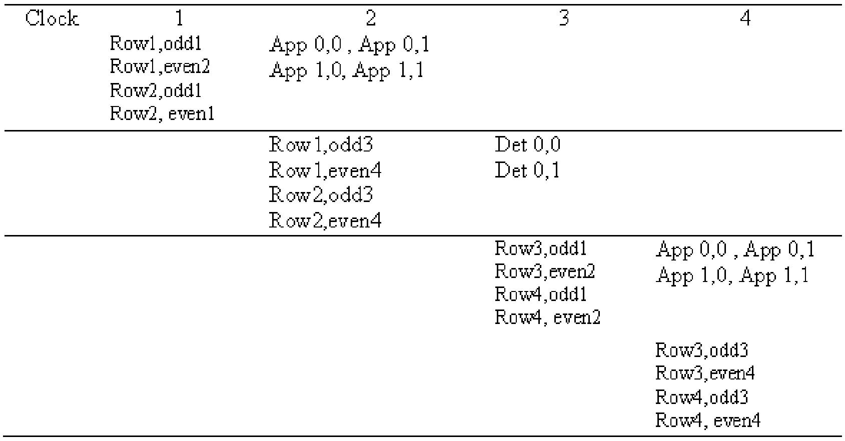


Fig.1. Block diagram of proposed Image coder

From the 2D DWT processor outputs like LL, LH, HL and HH frequency sub bands, low frequency part (LL) contains the most detail of the image. Then the LL frequency band is coded using vector quantizer. Vector quantizer, creates many code vectors, from the code vectors, reduced set of code vectors are selected.



Clock	1	2	3	4
	Row1,odd1 Row1,even2 Row2,odd1 Row2, even1	App 0,0 , App 0,1 App 1,0, App 1,1		
		Row 1,odd3 Row 1,even4 Row2,odd3 Row2,even4	Det 0,0 Det 0,1	
			Row3,odd1 Row3,even2 Row4,odd1 Row4, even2	App 0,0 , App 0,1 App 1,0, App 1,1
				Row3,odd3 Row3,even4 Row4,odd3 Row4, even4

Fig.2. Data flow sequence for 4x4 size of image block

3.1 Vector Quantization

There are different vector quantization techniques are available, each has its own advantages and disadvantages. Each technique is developed to reduce the parameters like spectral distortion, computational complexity and memory requirements. Existing vector quantization techniques [10] are as follows: Split vector quantization technique (SVQ), Multistage vector quantization technique(MVQ),multistage split vector quantization technique(MVSQ) and switched split vector quantization technique (SSVQ). The performance of the vector quantization mainly depends on the method of code book generation. The codebook is created efficiently using more training data set and using more number of bits for code generation.

Vector quantization is basically an encoding and decoding methodology of data. When applied to image, a codebook is created using training data set. Then by sectionalizing an image into a set of image blocks, which are non-overlapping blocks, image vectors are generated. Then each of the codebook vector is compared to each image code vector in order to find the one which has minimum distortion [11] with an input vector and code word is generated. The index of this closest will be sent to the decoder. The distortion measure can be the mean square error. The transmitted index will be received at the decoder and stored, which has the same code book as the encoder. The decoder then retrieves the code word according to the index received to reconstruct the input vector. Fig 3. shows illustrates the code word generation for the LL sub band output from the DWT. Vectors of larger size produce better quality compared to vectors of smaller size. This is because vectors of smaller size reduce the correlation between the samples.

3.2 K-means Clustering Algorithm

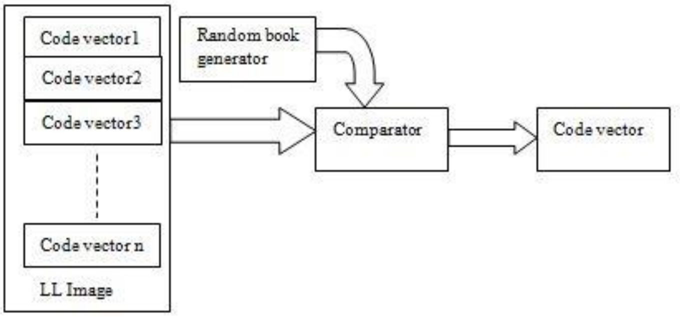


Fig.3. Block diagram of the proposed Vector Quantizer

Step 1: The training vectors are grouped into M clusters based on the distance between the code vectors and the training vectors.

Step 2: Compute the sum vector for each cluster.

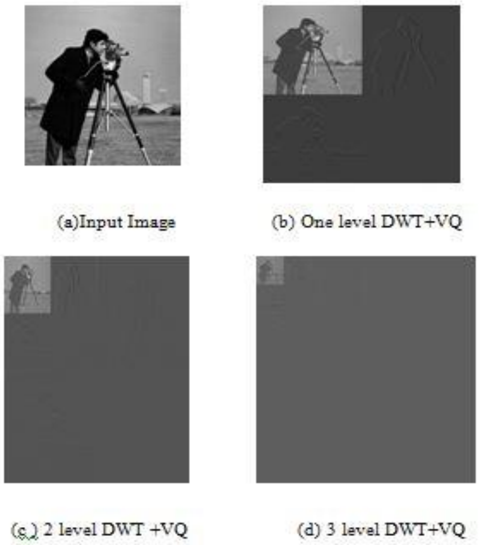
Step 3: Compute the centroid for each cluster.

Step 4: Replace the existing code vector with the new centroid to from the revised codebook.

Step 5: Repeat the steps1 through 4 till the codebooks of consecutive iterations converge.

4. Experiment results and analysis

The experiment platform of this paper is Pentium IV 2.0 GHz processor, 256 MB RAM, Windows 2000 Professional operating system, MATLAB 7.3 and Xilinx 13.2 Software. The DWT and VQ schemes are designed based on modified lifting scheme with new Z scan and K means algorithms.



(a)Input Image

(b) One level DWT+VQ

(c) 2 level DWT +VQ

(d) 3 level DWT+VQ

Fig.3. Three level Image codec Outputs

Table 1. Synthesis results of DWT with VQ based image codec

Parameters	Existing system	Proposed System
Number of Flip flops	904	664
Number of 4 input LUTs	2995	2099
Number of slices	1853	1358
Total gate count	33613	18125
Power	103 mW	92 mW
Delay	36.98 ms	36.78 ms

The performance of the DWT and VQ system are compared in terms of number of slices, flip flops, LUTs, total number of gates, power and delay. The image chosen for this experiment is 512×512 gray scale image of Cameraman. Figure 4 shows. In the image parameter setting, the size of training sample is 4×4, and the codebook size is 256.The main purpose of this experiment is to apply K-Means algorithm on the design of codebooks in vector quantization, and finds out the best codebook. Table 1 summarizes the hardware requirements for the proposed novel hybrid DWT and VQ based image codec with the existing system. Table 2 explains the comparison of 2D DWT architectures in terms of adders, multipliers, Critical path and Hardware utilization efficiency (HUE). The proposed design shows the one multiplier delay and 100% hardware utilization.

Table 2. Comparison results of 2D DWT architectures

2D DWT	Multiplier	Adder	Memory	Critical Path	Control Complexity	Hardware Utilization Efficiency
Existing	10	16	4N	Tm	Complex	-
Proposed	10	161	Few registers	Tm	Simple	100%

5. Conclusion

In this brief, a novel hybrid DWT followed by VQ based codec have been proposed. The proposed hybrid image codec is compared best of with the existing structure in terms of hardware complexity, power, memory, and critical path delay. It is concluded that the proposed method is high speed architecture with a lower hardware complexity. It also requires low power consumption. The Xilinx-VHDL synthesis results show that the proposed image codec suits with real time applications such as JPEG 2000 image standard and MPEG 4.

REFERENCES

[1] I. Daubechies, W. Sweldens, J. Fourier Anal.Appl. 4(1998) 247–269.

[2] K.K .Parhi ,T.Nishitani T, IEEE Trans.Very Large Scale Integr. (VLSI) Syst., 1(1993) 191-202.

[3] P. Wu, L.Chen, IEEE Trans.Circuits Syst. Video Technol., 11(2001) 536-545.

[4] J.M.Jou,Y.H. Shiau, C. Liu , , in Proc. IEEEISCAS, 2(2001), 529–532.

[5] G.Shi,W.Liu, L.Zhang, , IEEE Trans. Circuits Syst. II, Exp. Briefs,56(2009) 290–294.

[6] B.K.Mohanty et al, , IEEE Trans. Circuits Syst.VideoTechnol., 23(2012) 353–363.

[7] W. Zhang et al, IEEE Trans. Circuits Syst. II, Exp. Briefs, 59(2012) 158–162.

[8] C. Xiong et al, IEEE Trans. Image Process., 16(2007). 607–614.

[9] B.F.Wu et al, IEEE Trans. Circuits Syst. VideoTechnol., 15(2005)1615–1628.

[10] Tzu-Chuen Lu et al, Journal of Information hiding and Multimedia signal processing, 1(2010)192-203.

[11] Yan wang et al, Test and Application, (2008) 431-434.