IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

INEXACT COMPUTING TECHNIQUE FOR IMAGE COMPRESSION

Ajay Patel[1], Deepak mishra[2], VNS institute of technology, Bhopal

Abstract

This paper proposes a new framework for digital image processing; it relies on inexact computing to address some of the challenges associated with the discrete cosine transform (DCT) compression. The proposed framework has three levels of processing; the first level uses approximate DCT for image compressing to eliminate all computational intensive floating-point multiplications and executing the DCT processing by integer additions and in some cases logical right/left shifts. The second level further reduces the amount of data (from the first level) that need to be processed by filtering those frequencies that cannot be detected by human senses. Finally, to reduce power consumption and delay, the third level introduces circuit level inexact adders to compute the DCT. For assessment, a set of standardized images are compressed using the proposed three-level framework. Different figures of merits (such as energy consumption, delay, power-signal-to-noise-ratio, average-difference, and absolute-maximum-difference) are compared to existing compression methods; an error analysis is also pursued confirming the simulation results. Results show very good improvements in reduction for energy and delay, while maintaining acceptable accuracy levels for image processing applications.

Keywords: Approximate computing, DCT, inexact computing, image compression

1. INTRODUCTION

TODAY'S computing system usually process a significant amount of information that is computational and power intensive. Digital Signal Processing (DSP) systems are widely used to process image and video information, often under mobile/wireless environments. These DSP systems use image/video compression methods and algorithms.

However, the demands of power and performance remain very stringent. Compression methods are often utilized to alleviate such requirements. Image/video compression methods fall into two general categories: lossless and lossy.

The latter category is more hardware efficient but at the expense of quality of the final decompressed images/videos.

For image processing, the Joint Photographic Experts Group (JPEG) method is the widely used lossy method while the Moving Picture Experts Group (MPEG) method is the widely used lossy method for video processing. Both standards use the Discrete Cosine Transform (DCT) algorithm as basic processing step. Many different fast algorithms for DCT [1], [2] computation have been developed for image and video applications; however as all these algorithms still need floating point multiplications; they are computationally intensive requiring extensive hardware resources. To address these concerns, coefficients in many algorithms such as [3] can be scaled and approximated by integers such that floating-point multiplications can be replaced by integer multiplications [4], [5]. The resulting algorithms are significantly faster than the original versions and, therefore, they are extensively used in practical applications. Consequently, the design of good approximations of the DCT for implementation by narrower bus width and simpler arithmetic operations (such as shift and addition) has received considerable attention over the last few years [6]. An advantageous feature of image/video processing is its highly error-tolerant nature; human senses cannot often perceive degradation in performance, such as quality of visual and audio information. Therefore, imprecise computation can be used in many applications that tolerate some loss of precision and some degree of uncertainty, [7], [8], such as for example image/video processing.

The introduction of inaccuracy at circuit level in the DCT computation targets specific figures of merit (such as power dissipation, delay and circuit complexity [9], [10]) and it is very challenging. This scheme targets low power and process tolerance is based on a logic/gate/ transistor level redesign of an exact circuit. A logic synthesis approach [9] has been proposed to design circuits for implementing an inexact version of a given function by considering the so-called error rate (ER) as metric for error tolerance. Reduction in circuit complexity at transistor level of an adder circuit (such as by truncating the circuits at the lowest bit positions) provides a reduction in power dissipation higher than conventional low power design techniques [10]; in addition to the ER, new figures of merit for estimating the error in an inexact adder have been presented in [11].

Therefore, this paper presents a new framework for approximate DCT image compression; this framework is based on inexact computing and consists of three levels.

- Level 1 consists of a multiplier-less DCT transformation, so involving only additions;
- Level 2 consists of high frequency component (coefficient) filtering;
- Level 3 consists of computation using inexact adders.

Level 1 has been widely studied in the technical literature [16], [17], [18]; Level 2 is an intuitive technique to reduce the complexity of computation while attaining only a marginal degradation in image compression. Level 3 follows a circuit level technique by which inexact computation is pursued (albeit new and efficient inexact adder cells are utilized in this manuscript). Therefore, the contribution of this manuscript is found in the combined effects of these three levels.

The proposed framework has been extensively analyzed and evaluated. Simulation and error analysis show a remarkable agreement in results for image compression as an application of inexact computing. Hereafter to avoid confusion the word “approximate” is used only for the DCT algorithms while the word “inexact” is used for circuits and design involving non-exact hardware for computing the DCT.

2. Data Compression Model

An information pressure framework basically comprises of three significant advances and that are evacuation or decrease in information excess, decrease in entropy, and entropy encoding. A regular information pressure framework can be named utilizing the square outline appeared in Figure 1. It is performed in steps, for example, picture change, JPEG is one of the most utilized picture pressure standards which utilize discrete cosine transform (DCT) to change the picture from spatial to recurrence space [2]. A picture consists of very low visualization in its high frequencies for which overwhelming quantization should be possible so as to decrease the size in the changed portrayal. Entropy coding follows to additionally decrease the excess in the changed and quantized picture information.

3. ADVANTAGES OF DATA COMPRESSION

The primary bit of leeway of pressure is that it diminishes the information stockpiling prerequisites. It additionally offers an appealing way to deal with diminish the correspondence information over long stretch connections by means of higher compelling use of the accessible transfer speed in the information joins. This fundamentally helps in diminishing the expense of correspondence because of the information rate decrease. Due to the information data compression, information pressure additionally expands the nature of sight and sound introduction through restricted transfer speed correspondence channels, Because of the decreased information rate. Advertised by the pressure methods, PC system and Internet use is getting more and more picture and realistic cordial, instead of being simply information and content driven wonders. In short, elite pressure has made new chances of innovative applications such as advanced library, computerized chronicling, video remotely coordinating, telemedicine and advanced amusement to give some examples. There are numerous other auxiliary focal points in information pressure. Data compression may upgrade the database execution since increasingly compacted records can be pressed in a given cradle space in a customary PC execution. This possibly builds the likelihood that a record being looked through will be found in the principle memory. Data security can likewise be significantly improved by scrambling the translating parameters and transmitting them independently from the compacted database documents to confine access of restrictive data. An additional degree of security can be accomplished by making the pressure and decompression forms absolutely straightforward to unapproved clients. The pace of information yield tasks in a processing gadget can be enormously expanded because of shorter portrayal of information. Image compression clearly diminishes the expense of reinforcement and recuperation of information in PC frameworks by putting away the reinforcement of enormous database documents in packed structure. The benefits of information pressure will empower more interactive media applications with reduce cost [19,21].

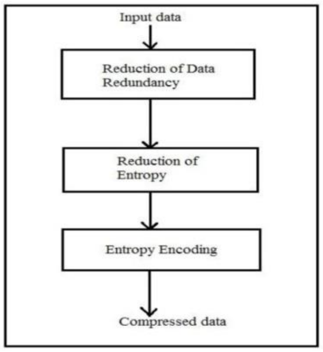


Fig 1Data compression model

4. DISCRETE COSINE TRANSFORM (DCT)

To obtain the ith and jth DCT transformed elements of an image block (represented by a matrix p of size N), the following equation is used:

Where, p(x, y) is the image's x, y^{th} element. The result is an

$$D(i,j) = \frac{1}{\sqrt{2N}} C(i)C(j) \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} p(x,y)\, cos\left[\frac{\pi(2x+1)i}{2N}\right] cos\left[\frac{\pi(2y+1)j}{2N}\right]$$

$$C(u) = \begin{cases} 1/\sqrt{2}, & u = 0 \\ 1, & u > 0 \end{cases}$$

input I jth) from the pixel values in the original image matrix in the transformed image. N is equal to 8 and x and y for the frequently used 8x8 block for JPEG compression. Therefore, the following equation is also given for D (i, j).:For matrix calculations, the DCT matrix is obtained from the following:

$$T_{DCT}(i,j) = \begin{cases} 1/\sqrt{N}, & i = 0 \\ \sqrt{\frac{2}{N}} cos\left[\frac{\pi(2j+1)i}{2N}\right], & i > 0 \end{cases}.$$

$$D(i,j) = \frac{1}{4} C(i)C(j) \sum_{i=0}^{7}\sum_{j=0}^{7} p(x,y) cos\left[\frac{\pi(2x+1)i}{16}\right] cos\left[\frac{\pi(2y+1)j}{16}\right].$$

Joint Photographic Experts Group (JPEG)

The JPEG processing first starts with the use of the DCT to transform an image into the frequency domain, separating images into parts of different frequencies. The quantification is then done in such a manner that small frequencies are omitted. This reflects people's capacity to see small differences in light over a relatively large area, but normally the exact strength of a vastly variable brightness variation is not distinguished. This quantization process takes place during which each variable is separated by a constant, and then rounded to the nearest integer in the frequency region.. It leads to other high-frequency variable values that are at best very low or possibly zero. During the decoding process, the image is then recovered with only the appropriate preserved frequencies. The following steps should be taken for JPEG processing: The compressed image (block array) is decompressed using Inverse DCT (IDCT) for image retrieval. Various fast algorithms have been developed to provide [1] computation for video and image applications. The system suggested by LOEFLER et al.[19] involves 11 multiplications and 29 additions, which are considered most effective since the theoretical lower limit for 1-D 8- point DCT is 11[20][21]. Some DCT operations can be incorporated into the quantization step to further reduce calculation complexity. For example the Arai method needs only five multiplications, and a limited number of additions 29 for accuracy.[3] Such so-called scaled DCTs will result in substantial changes.

Even these fast algorithms also need to be multiplied by floating points, making their execution sluggish and complex. To obtain faster computation the coefficients can be scaled down and approximated in various algorithms such as the Arai process [3], so that multiplication of the floating point by the integer multiplication can be replaced [4], [5]. The algorithms resulting are much faster. The fixed-point multiplication specifications of these fast algorithms typically need a broad data bus (often 32 bit) such that both hardware and power consumption need a expensive VLSI implementation. The design of a narrower bus width and simple arithmetic operations (specifically, shift and addition) for the implementation of good DCT approximations therefore is extremely attractive [6].Low-complexity performance methods in recent years.

In the scientific literature, measurement of the 8-point DCT can be found [22]. Signed DCT (SDCT) [23], Lengwehasatit-Ortega approximation level 1 [24], Bouguezel-Ahmad-Swamy (BAS) round-off rough-off rough-overs [25], [26],[27] and DCT approximation [28] are the approximate techniques for approximations. However, the application is not beneficial to all approximation techniques found in the literature; e.g., [24] requires More measures to create the correct approximation before DCT is computed. In general, for approximate DCT methods, transformation matrix inputs only are {0,+-1/2,+-1,+-2}, which means that the so-called null multiple complexity is feasible because the operations involved can be performed solely by additional operations and Bit shift operations [28].

These approaches to DCT are based on the SDCT as defined simply by applying the sign function operator to the DCT matrix coefficients, There are only positive / negative ones and zeros in the DCT matrix, which implies addition / subtraction or no operation. By limiting all TSDCT coefficients to zero, The 88 transform C matrices can be obtained with a two-step process: (1) introduce a few zeroes and a quadruple of a few in the 8X8 SDCT matrix for the generation of a modified D to orthogonally, i.e., C-1 = T'D. For this reason the DCT is carried out only through a transformation matrix and its transposition. The signed DCT [23] needed two separate (forward and reverse) transformations as T is not orthogonal. the first BAS2008 approximate transformations[25];

$$T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 1 & 1/2 & -1/2 & -1 & -1 & -1/2 & 1/2 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1/2 & -1 & 1 & -1/2 & -1/2 & 1 & -1 & 1/2 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \end{bmatrix},$$

For examples, X must be an 8x8-block matrix of an image and F must be its corresponding block can be carried out forward and reverse with an approximate transform in the form of F = CXC '= D(TXT')D and X = C' FC [DFD]T. However, it can be done by mixing the diagonal matrix D together with all other transformation in the literature because the block matrix is quantized / de- quantified in the transform domain. The approximate transform is not multiplication-free, as well as any other transforms found in the literature. F = TXT 'and the reverse are X = T'F'T, where F' is decoded by the DCT is given by the conventional DCT transformation. Results demonstrate the 8-point DCT code sophistication and compare with the original DFT, Cooley – Tukey DFT and the original DCT approximation methods. In addition to the approximation of manipulation of the DCT matrix, fast DCT computational algorithms,

$$T_{SDCT}(i,j) = \frac{1}{\sqrt{N}} sign\{T_{DCT}(i,j)\}$$

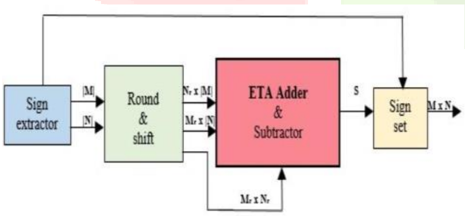
$$sign\{u\} = \begin{cases} +1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0. \end{cases}$$

iii.

Mean Square Error (MSE):

$$\text{MSE} = \frac{1}{m \times n} \sum_{x=1}^{m} \sum_{y=1}^{n} (p_{x,y} - \hat{p}_{x,y})^2$$

including multiplier-free approximations[18] and the behavioral judgment at pixel level[30] were suggested.



These

Peak Signal to Noise Ratio (PSNR):

$$\text{PSNR} = 10 \log \frac{(2^n - 1)^2}{\text{MSE}}$$

5. PROPOSED WORK

This section introduces a new frame of image compression consisting of three approach stages as follows.

Phase 1 is DCT transition without multipliers, Phase 2 involves high-frequency converter, and Phase 3 is the incorrect calculation. In previous sections, Phase 1 and 3 were explained. Although the high-frequency (Phase 2) filtration is not a new approach, it should also be described for completeness purposes because it helps to reduce the time and energy used for the proposed framework. Therefore the process is carried out on the set of coefficients only for the low- frequency components of the transformed block instead of quantifying all resulting DCT coefficients..

i. High Frequency Filtration

It is possible to compact an image with this function. As mentioned above, a DCT changes the picture to the frequency field, so that it can be ignored those coefficients which encode the high- frequency (not human-eye) components, while retaining other coefficients..

A number of retained coefficients have been considered in applications for image compression; 0.34-24.26% of DCT 92112 is enough in applications for high-speed face recognition [27]. Examples are as follows for 8X 8 image blocks:

Compression image with a supporting vector machine that only considers the first 8–16 coefficients,

1. Only three coefficients are used as proposed in an image reconstruction approach,
2. Assess and evaluate the different methods of compression using only 10 coefficients [25],[26].

ii. Approximate DCT Implementation

As opposed to the implementation of approximate DCT methods, the following calculations are performed on bit level with the corresponding logical functions: addition and subtraction. The operator's length is 32-bit and implementations are simulated ***by***

Use the logical functions of MATLAB Boolean. The image shall simulate selected approximate DCT approaches and the results will be shown in Chapter-4, where the Power Signal to Noise Ratio (PSNR) of every method is compared. During the compression quantization phase. The Middle Square Error (MSE) calculates the PSNR as follows: When pj, k is the exact pixel value at column y and x in rows of the pixel image, px, y is the approximate pixel.

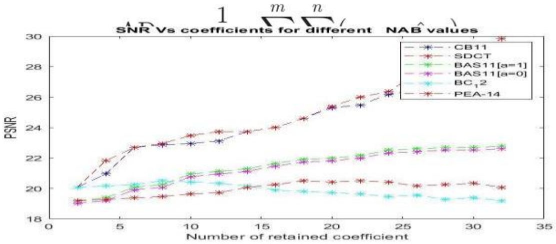
Fig 2 Block diagram of adder

The data shows that compression using CB11 generally delivers the best result in terms of PSNR except for the non-orthogonal SDCT method. There are three types of behaviour.

1. Increasing production quality by increasing the number of coefficients retained (RC). For CB11, BAS08, BAS09, BAS11 (a=0 and a=1), this is done for the following purposes.
2. A nearly permanent PSNR by increasing the RC. For BC12 and PEA14.

The quality of the output degradation is increased by RC. For the BAS11 (a=2) as well as for the PEA12.The

Average Difference (AD):



resulting quality, i.e. the average difference (AD) and the max. Absolute difference (MD) is better known in two additional steps. These measurements are defined as

6. Approximate DCT Using Inexact Computing

Then consider about Lena DCTs using inaccurate adders; the NAB 's value is raised from 3 to 5, as before. The PSNR of compressed images (a quality measurement) is traced with an inaccurate adder (for example the top row uses AMA1 as the incorrect adder) to execute all approximated DCT methods. By performing all of the approximate DCT methods with only inaccurate adders (for a NAB value), each column records the quality of the compressed images. For instance, for NAB =3 the left-most column is. As anticipated, the PSNR deteriorates with the NAB increasing (an acceptable PSNR level of a NAB value of 4 is achieved).

The architecture proposed was adopted from (11), it is based on the singed rounding and unsigned number to the shape 2. In order to reduce the energy consumption, it is the main idea to use an approximate adder in place of extracts ones. The two input value M and N are multiplied by the following:

7. RESULTS

The values of CR are very critical it can be modified so as to achieve different image quality and characteristics. CR must be as maximum as possible and higher values are achieved by simply illuminating the detail coefficients. If we have to get higher values of CR we have to take some constant values called SF(scalar factor).The quantize output is divided by SF and rounded to nearest integer value (MATLAB round function).The result of this process creates more zero values which are further illuminated during compression.CR higher values signifies the lower reconstruction characteristics of an image. As it can be seen at higher compression ratio we are obtaining blurred image as compare to the image having less compression ratio. In the next section the results for every algorithm and their comparison has been discussed. MATLAB 2018 has been used for implementation and visualization purpose. Different logical function is used to implement adder and sub tractor at bit level. We have taken 32 bit data at a time for all the operations. Different images are considered such as natural, benchmark and high capacity images, but most frequently Lena image is taken into consideration. All the DCT approach uses the same image. The analysis and results for PSNR Vs number of Retained Coefficients (RC) used in the quantization stages fig3.

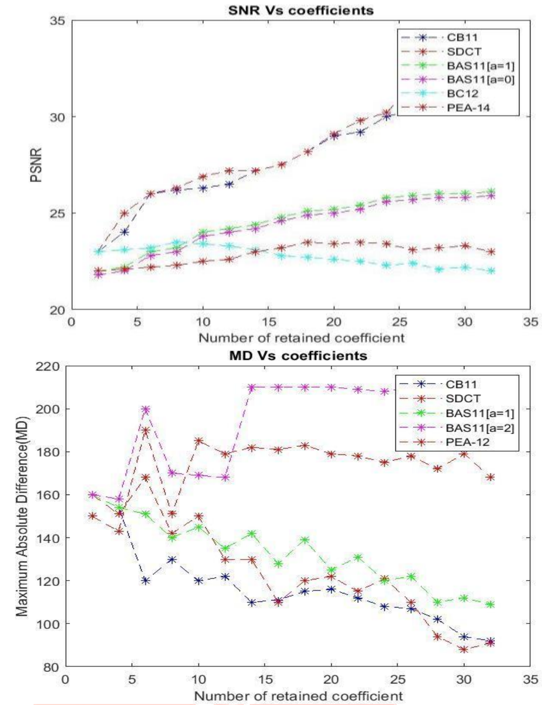


Fig 3 Graphs of achieve results

Fig4. Compression of Lena under considered approximate DCT methods and using an InXA2-based inexact adder with RC = 10; (a) SDCT [23], (b) BAS08 [25], (c) BAS09 [26], (d) BAS11 [27]a=0, (e) BAS11 [27] a=1, (f) BAS11 [27] a=2, (g) CB11 [28], (h) BC12 Exact DCT compression with RC=10, and (l) Original uncompressed image.

8. CONCLUSION

In this thesis a new approach for compressing images by approximate compression using the Discrete Cosine Transform (DCT) algorithm has been studied and implemented. The complete implementation consists of three subdivisions which are: DCT implementation without using multiplier (includes only adder and shift operations), high frequency component filtering and the last step is computing by using inexact approach. In the first step normal DCT is applied to the image, only difference is that no multipliers are used. Any input image is divided into 8 X8 block and this passed through all the levels of implementations. At the end a good quality compressed image is achieved by using these three steps.

The overall effect of these steps have been studied and understood completely. As an application of inexact computing it gives satisfactory results. The proposed approach has been implemented in MATLAB 2018.This thesis also gives comparative analysis among all the approximate DCT methods and results shows that CB11 provides best compression in terms of highest PSNR values. In this case exact 16 bit adder has been used. Other image manipulation quality measures (AD and MD) confirmed the PSNR results. Methods BAS08, BAS11 with a =0 and BAS11 with a=1 are the next best methods.

There are various types of inexact adders as discussed in previous chapters but InXA2 gives best results among all. Various steps of compression such as approximate DCT JPEG compression, truncation are implemented using inexact adders. It is clear from the results that non-truncation based methods gives better results as compared to the truncation schemes, especially when higher values of NABs are taken into consideration. Different images have been given as input and computed DCT by inexact adder; it is found that inexact adder gives consistent performance for all the cases. In general acceptable compression has been achieved with NAB values up to 5. Then it has been shown that the quality of the results decreases substantially when larger NAB values are used.

REFERENCE

[1] R. E. Blahut, Fast Algorithms for Digital Signal Processing. Reading, MA, USA: Addison-Wesley, 1985

[2] V. Britanak, P. Yip and K. R. Rao, Discrete Cosine and Sine Transforms. New York, NY, USA: Academic, 2007

[3] Y. Arai, T. Agui, and M. Nakajima, “A fast DCT-SQ scheme for im-ages,” IEICE Trans., vol. E-71, no. 11, pp. 1095–1097, Nov. 1988.

[4] D. Tun and S. Lee, “On the fixed-point-error analysis of several fast DCT algorithms,” IEEE Trans. Circuits Syst. Vid. Technol., vol. 3, no. 1, pp. 27–41, Feb. 1993.

[5] C. Hsu and J. Yao, “Comparative performance of fast cosine transform with fixed-point roundoff error analysis,” IEEE Trans. Signal Process., vol. 42, no. 5, pp. 1256–1259, May 1994.

[6] J. Liang and T. D. Tran, “Fast multiplierless approximations of the DCT with the lifting scheme,” IEEE Trans. Signal Process., vol. 49, no. 12, pp. 3032–3044, Dec. 2001.

[7] Y. Dote and S.J. Ovaska, “Industrial applications of soft computing: A review,” Proc. IEEE, vol. 89, no. 9, pp. 1243–1265, Sep. 2001.

[8] K. V. Palem, “Energy aware computing through probabilistic switching: A study of limits,” IEEE Trans. Comput., vol. 54, no. 9, pp. 1123–1137, Sep. 2005.

[9] D. Shin and S. K. Gupta, “Approximate logic synthesis for error tolerant applications,” in Proc. Des. Automat. Test Europe, 2010, pp. 957–960.

[10] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, “Lowpower digital signal processing using approximate adders,” IEEE Trans. Comput.-Aided Des. Integrated Circuits Syst., vol. 32, no. 1, pp. 124–137, Jan. 2013.

[11] J. Liang, J. Han, and F. Lombardi, “New metrics for the reliability of approximate and probabilistic adders,” IEEE Trans. Comput., vol. 62, no. 9, pp. 1760–1771, Sep. 2013.

[12] H. Jiang, J. Han, and F. Lombardi, “A comparative review and evaluation of approximate adders,” in Proc. ACM/IEEE Great Lakes Symp. VLSI, May 2015, pp. 343–348.

[13] Z. Yang, A. Jain, J. Liang, J. Han, and F. Lombardi, “Approximate xor/xnor-based adders for inexact computing,” in Proc. IEEE Int. Conf. Nanotechnology, Aug. 2013, pp. 690—693.

[14] H. A. F. Almurib, T. Nandha Kumar and F. Lombardi, “Inexact designs for approximate low power addition by cell replacement,” in Proc. Conf. Des. Autom. Test Europe, Mar. 2016, pp. 660–665.

[15] N. Banerjee, G. Karakonstantis, and K. Roy, “Process variation tolerant low power DCT architecture,” in Proc. Des. Automat. Test Europe, 2007, pp. 1–6.

[16] U. S. Potluri, A. Madanayake, R. J. Cintra, F. M. Bayer, and N. Rajapaksha, “Multiplier-free DCT approximations for RF multi-beam digital aperture-array space imaging and directional sensing,” Meas. Sci. Technol., vol. 23, no. 11, pp. 1–15, Nov. 2012.

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

[18] N. Merhav and B. Vasudev, “A multiplication-free approximate algorithm for the inverse discrete cosine transform,” in Proc. Int. Conf. Image Process., Oct. 24-28, 1999, pp. 759–763.

[19] A. Jain, “Fundamentals of Digital Image Processing” Prentice-Hall, 1989.

[20] K. Rao and J. Hwang, “Techniques and Standards for Image, Video and Audio Coding”, Prentice-Hall, 1996

[21] W. Penebaker and J. Mitchell, ”JPEG Still Image Data Compression Standard”, Van Nostrand, 1993.

[22] E. Whittaker, “On the Functions which are Represented by the Expansions of Interpolation Theory,”Proc. Royal Soc., Edinburgh, Section A 35, pp. 181-194, 1915.

[23] D. LeGall and A. Tabatabai, “Subband Coding of Digital Images Using Symmetric Short Kernel Filters and Arithmetic Coding Techniques,” Proc. ICASSP, IEEE, pp. 761-765, 1988.

[24] D. Huffman, ”A Method for the Construction of Minimum Redundancy Codes, ” Proc. IRE, pp. 1098-1101, 1952.

[25] Mark J. T. Smith and Steven L. Eddins. “Analysis/synthesis techniques for subband image coding” IEEE Trans. Acoust., Speech, Signal Process., pp. 1446–1456, August 1990

[26] R. L. Joshi, H. Jafarkhani, J. H. Kasner, T. R. Fisher, N. Farvardin, M. W. Marcellin, and R. H. Bamberger, “Comparison of different methods of classification in subband coding of images,” submitted to IEEE Trans. Image Processing, 1995.

[27] Jacquin, A.E., “A novel fractal block-coding technique for digital images”, ICASSP‘90, pp. 2225–2228, Albuquerque, NM, 1990.

[28] Jang, J. and Rajala, S.A., “Segmentation-based image coding using fractals and the human visual system”, ICASSP‘90, pp. 1957–1960, Albuquerque, NM, 1990