Fast Implementation of the Integer Transform in H.264 /AVC Video Encoders

C.S. Lubobya, M.E. Dlodlo, G. De Jager, K.L. Ferguson
University of Cape Town, Department of Electrical Engineering, Rondebosh Cape Town, 7700,
Tel: +27788648756.
Emails: cslubobya@yahoo.com, mqhele.Dlodlo@uct.ac.za,Gerhard.DeJager@uct.ac.za, kferguson@csir.co.za

Abstract- **This paper investigates fast software implementations of integer transforms in video encoders. The ever increasing demand for video applications on the public Internet and particularly in environments of low bandwidth necessitate the need for video compression. In a video encoder a hybrid of data compression units and algorithms are used to remove redundancy in the video. The 4x4 integer transform used in the H.264/AVC coding standard is considered here and the aim is to improve the computation speed while maintaining subjective quality of the video data.**

Keywords **– H.264/AVC, integer transforms, Discrete Cosine Transform, video compression**

I. INTRODUCTION

Previous video compression standards such as JPEG, MPEG-1, MPEG-2, H.261 and H.263 used Discrete Cosine Transforms (DCT) as transform coding on video [1]-[8]. The current standard H.264/ MPEG-4 part 10 or Advanced Video Coding (AVC) also defined by the ITU-T as H.264/AVC has four profiles; baseline, main, extended and high. Each of theses profiles uses the integer transform as the primary transform coding techniques to reduce redundancy in the video signal.

The core matrix of the integer transform can be implemented as a direct 2-D Matrix multiplication or as 1-D using adds and shift operations along the rows and then down the columns, also called the butterfly technique [1][3].Both methods are considered here. The research aims to improve the computation speed of the core integer transform matrix in software.

The paper is organised as follows: Section II, gives related work on integer transforms, section III shows the detailed derivation of the mathematical model of forward integer transforms. Section IV discusses the methodology used in achieving results and the conclusion is given in V.

II. RELATED WORK

(Marvar et.al) suggests treating 2-D 4X4 integer transforms of the core matrix as row-column application of 1-D methods, that is, separately implement the 1-D four dot integers in each of the rows and column [2]. (Fan) introduced an algorithm based on sparse factorisations of the 4x4 matrix and reduced the computational complexity by having 64 additions and 16 shifts [6]. (Fan) further reduced the adds to 8, 32 and shifts to 2, 32 for 4x4 and 8x8 block sizes respectively [7]. (He) also combined the two blocks and reduced the adds to 40 and shifts to 12 [9]. (Honey et.al) reduced adds to 73 and multiplications to zero [4]. (Xuihau et.al) also reduced the average adds to 12.78 and average shifts to 1.6784 [5]. All the above methods were aimed at reducing additions, shifts and/or multiplications. The proposed method aims to decrease the computation time (in seconds) using Single Instruction Multiple Data (SIMD) or Streaming SIMD Extension (SSE) instructions.

III. MATHEMATICAL MODEL

This section describes the derivation of the mathematical model which will be used in the experiments.

A. 4X4 H.264/AVC INTEGER TRANSFORMS

For the 4x4 2-D baseline profile the integer transform can be derived from the 4x4 2-D DCT as shown in [1]-[8]. In general the transform can be expressed in the form:

$$Y = AXA^T \quad \text{.......} (1)$$

From Equation (1), matrices X and Y are the input and output video signal data, respectively. A is the transform matrix and A^T is its transpose. The elements of A are:

$$A = \begin{pmatrix} a & a & a & a \\ b & c & -c & -b \\ a & -a & -a & a \\ c & -b & b & -c \end{pmatrix} \quad \text{.......} (2)$$

Where: $a = 1/2$, $b = \sqrt{1/2}\cos \pi/8$, $c = \sqrt{1/2}\cos 3\pi/8$

Substituting equation (2) into (1) and factorising the result gives:

$$Y = \left(CXC^T\right) \otimes E \quad \text{.......} (3)$$

Where:

$$C = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & d & -d & -1 \\ 1 & -1 & -1 & 1 \\ d & -1 & 1 & -d \end{pmatrix} \text{ and } E = \begin{pmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{pmatrix}$$

E is a matrix of scaling factors and the symbol $\otimes$ indicates coordinate wise multiplication by the scaling factor in the same position in matrix E [1][8]. ($d = c/b$). In H.264 the implementation of the transform is simplified by approximating d to 0.5 and b is modified to $b = \sqrt{2/5}$ to make it orthogonal [1].

Therefore equation (3) now reduces to:

$$Y_{ij} = \left(C_f X C_f^T\right) \otimes E_f \qquad (4)$$

Where by:

$$C_f = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix} \text{ and } E_f = \begin{pmatrix} a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \\ a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \end{pmatrix}$$

$$C_f X C_f^T = W \qquad (5)$$

Equation (5) is the core matrix of forward transform.

B. AVOIDING MULTIPLICATION IN $C_f X C_f^T$

To avoid multiplication in equation (5) ,the 2-D 4X4 input matrix (X) in the core matrix can be computed as 1-D using adds and shift operations along the rows and then down the columns. This technique also known as butterfly method is shown in figure 1 below:

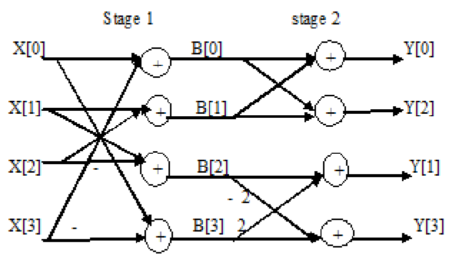

Figure 1: Computations of 2-D input matrix using 1-D add and shifts [3].

IV. RESEARCH METHODOLOGY

Experiments using the C++ software language [9] [10] and/or Streaming SIMD extensions (SSE) with standard videos will be designed to measure the bit rate (Bits per pixel-Bpp), Peak Signal to Noise Ratio (PSNR) and computation time at varying video quality levels.

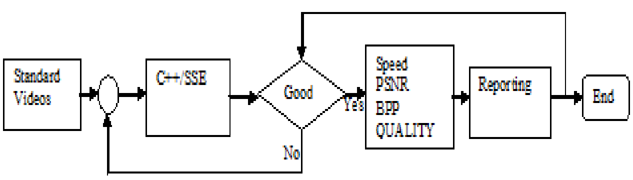

Figure 2: Experimental Model

V. CONCLUSION

This paper has presented the proposed work on integer transforms, the mathematical model and methodology that will be adopted in experiments. Using standard videos, the experiments will investigate the computation speed improvement of integer transforms in video encoders at varying video quality levels for real-time video broadcasting /internet applications.

ACKNOWLEDGEMENT

This work is supported by the Council for Scientific and Industrial Research (CSIR) through the DST-Innovation Fund on project no. T70026: Bandwidth Adaptive Real-Time Video Broadcasting over Internet.

REFERENCES

[1] I.E.G Richardson, "H.264 and MPEG-4 video compression", published by John Wiley and sons, West Sussex, UK, 2003.

[2] H.S.Marver, A.Hallapuro, M.Karczewicz and L.kerofsky, " Low Complexity Transform and Quantisation in H.264/AVC" IEEE Transactions on circuits and systems for video technology, vol. 13,No 7, pp. 560-576,july,2003.

[3] H. Kalva and J.B.Lee, "The VC-1 and H.264 Video compression standards for broadband video services", Springer, New York, USA, 2008.

[4] Honey Durga Tiwari, Ganzorig Gankhuyag, Gi Hyun Kim, Chan Mo Kim, Yong Beom Cho "Multiplier Less Fast Loss Less Integer DCT for H.264" 2008 International SoC Design conference, IEEE, 2008.

[5] Xiuhua Ji, Caiming Zhang and Kai Wang, "A Fast Two-Dimension 4x 4 Inverse Integers Transforms Algorithm for Real-Time H.264 Decoder" ICIC July, 2008.

[6] C.P.Fan, "Fast 2-dimentional 4x4 forward integer transform implementation for H.264/AVC", IEEE Transactions on circuits and systems II: Express briefs, volume 53, Issue 3, pp. 174-177, March, 2006.

[7] Chil-Peng Fan, "Cost Effective Hardware Sharing Architectures of Fast 8x8 And 4x4 Integer Transforms For H.264/AVC" IEEE 2006.

[8] I.E.GRichardson, "overview of H.264 Advanced Video", www.vcodex.com/h264overview.html, Retrieved on 20 August, 2009.

[9] B. Stroustrup, "The C++ Programming Language", Addison-Wesley, special edition, New Jersey, USA.

[10] S.Meyers," Effective C++:55 Ways to Improve Your Programs and Designs", 3rd edition, Addison-Wesley, New Jersey, USA.

BIOGRAPHY

Charles Smart Lubobya received his undergraduate degree in 2009 from the Copperbelt University in Zambia and is currently studying towards his Master of Science degree at the University of Cape Town. His research interests include video compression and streaming, wireless communication and data networks.