A 3D DCT ARCHITECTURE FOR COMPRESSION OF INTEGRAL 3D IMAGES

I. Jalloh, A. Aggoun and M. McCormick
3D Imaging Group
Faculty of Computing Sciences and Engineering
De Montfort University
The Gateway, Leicester LE1 9BH

Abstract - A VLSI architecture for three-dimensional discrete cosine transform (3D DCT) is proposed. The 3D DCT is decomposed into 1D DCTs computed in each of the three dimensions. The focus of this paper is in the design of the matrix transpose required prior to the computation of the final 1D DCT which corresponds to the third dimension. In this paper, this matrix transpose is divided into N memory units each performing the row-column transpose and switching networks to allow correct read and write. This architecture uses 3N multiplier-accumulators and N+1 (N×N)-words memory transpose to evaluate an (N×N×N)-point DCT at a rate of one complete 3D transform per N^3 clock cycles, where N is even.

INTRODUCTION

Applications of 3D DCT have been reported by Rao and Yip [1]. For example, it has been applied for compression of multi-spectral scanner data based on 4x4x4 cubes composed of 4x4 blocks from each of the four spectral bands. Another well-known application is the 3D DCT of 3D blocks, displaced by motion estimation, performed on each frame. These blocks are generated from non-interlaced, High Definition Television (HDTV) frames.

In recent years, the 3D DCT has found another very important application, namely compression of integral 3D video signals [2], [3], [4], [5], [6]. Integral 3D imaging is an imaging system based around Lippmann's integral photography techr:.que [7]. Several advances have been reported by the 3D Imaging group at De Montfort University to record a one stage orthoscopic integral image [5], [6]. A single 'camera' unit encodes a true optical model of the scene as a single planar intensity distribution suitable for electronic capture in real time. A flat panel display system, such as one using LCD technology, is used to reproduce the captured intensity modulated image. A two-dimensional microlens array, identical to that used for capture, re-integrates the captured rays to replay the original scene in full colour and with continuous parallax in all directions. This system offers the potential of stress-free viewing by more than one person. These features make this imaging system very suited to live entertainment applications such as 3D TV.

0-7803-6488-0/00/$10.00 © 2000 IEEE

The planar intensity distribution representing an integral image is comprised of 2D array of NxN subimages due to the structure of the microlens array used in the capture. Due to the small angular disparity between adjacent microlenses, a significant cross correlation exists between neighbouring subimages. Additionally a significant correlation exists between neighbouring pixels within each subimages. A lossy compression scheme for use with unidirectional integral 3D images, making use of a three dimensional discrete cosine transform (3D-DCT) has been developed [2], [3], [4]. Unidirectional integral imaging is a special case where 1D cylindrical microlens array is used for capture and replay. Use of the 3D DCT allows decorrelation in the three spatial dimensions. N neighbouring subimages are assembled to form a volume of input data to the 3D DCT processing unit, as shown in figure 1, for N=8 and in the case of unidirectional integral image. It was shown that the performance with respect to compression ratio and image quality is vastly improved compared with that achieved using baseline JPEG for compression of full parallax integral 3D image data [2], [3], [4].

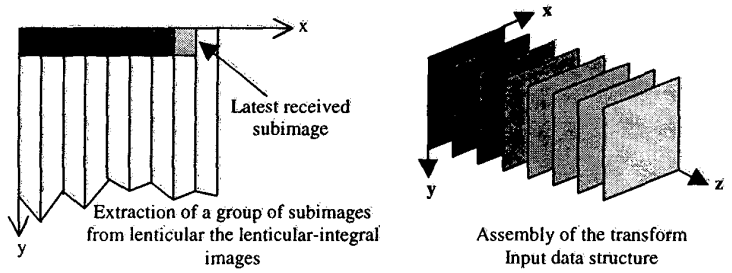

Figure 1 Assembly of 3D DCT Input Data Structure

In this paper, a VLSI architecture for the computation of the 3D DCT is presented. It uses three 1D DCTs and two matrix transposes to reduce the computation requirements. An NxN-point 2D DCT is computed using the well established row-column technique. The output of N 2D DCTs are stored in a N^2xN memory, termed T_2, which is shuffled to allow the correct reading for the final N-point 1D DCT. The transposition operation of the N^2xN memory required prior to the third 1D DCT can not be performed in the conventional manner i.e. row-column transpose. This is due to the fact that this matrix is not square and that each element of the N-point data fed to the final 1D DCT are collected every N^2 cycles. In this paper, this memory is divided into N distinct NxN memory and a switching network to enable a fast and simple read/write system. The advantage of the proposed solution is that all matrix transpose are similar and carried out using the row-column transpose.

3D DCT ARCHITECTURE

VLSI architectures for the 1D DCT and 2D DCT are very popular [7], [8], [9]. This is not the case for 3D DCT. The most straightforward way of implementing the 3D DCT (or 3D IDCT) is to follow the theoretical equations. However, the straightforward method would require N^3 multiplication and N^3-1 addition operations per coefficient for an (NxNxN)-point DCT. Using the separability property of the multi-dimensional DCT, the 3D DCT can be implemented using a cascade of three N-point 1D DCTs. This is achieved by performing 1D DCT along the rows (columns) of the array followed by 1D DCT along the columns (rows) of the transformed array. The third 1D DCT is performed on corresponding pixels in each of the N subimages that constitute the third dimension. The block diagram is shown in figure 2.

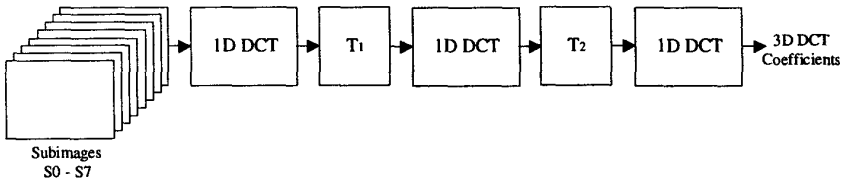

Figure 2 Block Diagram for the Computation of 3D DCT Coefficients

The input to the first 1D DCT is scanned row by row, the 1D DCT is performed on each input value as they are read and the intermediate result is stored in the NxN transpose memory, T_1. The transpose memory acts as a buffer and performs matrix transpose, since rows are written into it and columns are read out. After transposition of the resultant matrix, another 1D DCT is performed on the coefficients to yield the 2D DCT coefficients. This is the conventional row-column 2D DCT computation. The 2D DCT is performed on each subimage, S0 to S7 for N=8, where S0 is the first subimage and S7 is the eighth sub image of the input volume. Figure 3 shows the pixel sequence of the eight subimages after the 2D DCT computation.

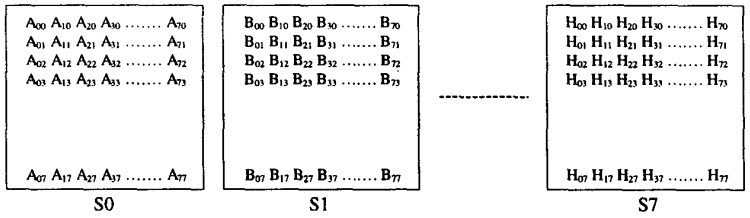

Figure 3 Pixel Sequence of the Eight Subimages After 2D DCT Computation

The output coefficients of 2D DCT are fed into the second matrix transpose, T_2, which is an $N^2 \times N$ memory. In this paper, this memory has been divided into N distinct N×N memory to enable a fast and simple read/write system. The transpose memory, T_2, is thus made up of eight distinct 8×8-word memories, T_{2_0}, T_{2_1}, T_{2_2}.....T_{2_7}, for N=8. Figure 4 shows the block diagram of the proposed architecture for the 3D DCT.

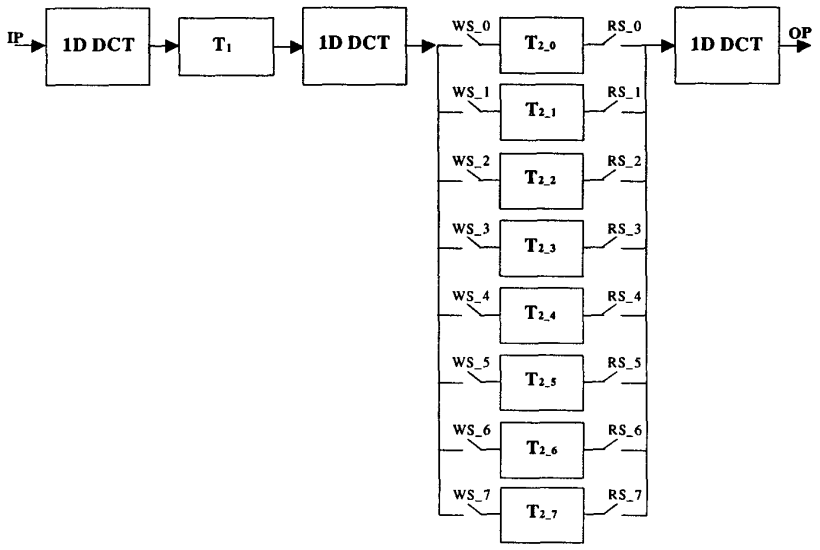

Figure 4 Proposed Architecture for the 3D DCT Computation

As mentioned earlier, the first two 1D DCTs and the matrix transpose T_1 perform the 2D DCT computation. The second 1D DCT block outputs the 2D DCT coefficients one sub image at a time. For clarification each 8×8 matrix (either subimages S0,...S7 or memory transpose T_{2_0},...T_{2_7}) has rows R0 to R7 (where R0 represents the first row of 8 words {i.e. at locations 0, 1, 2,...7,} R1 the second row of 8 words {i.e. at locations 8, 9, 10...15} etc) and columns C0 to C7 (where C0 represents the first column of 8 words {i.e. at locations 0, 8, 16,...56}, C1 the second column of 8 words {i.e. at locations 1, 9, 17...57} etc). The second 1D DCT block outputs the coefficients of S0 first, this is followed by the coefficients of S1, then the coefficients of S2 etc. The rows (R0 to R7) of S0 are written to R0 of each of T_{2_0} to T_{2_7}. S0-R0 is written to R0 of T_{2_0}, S0_R1 is written to the R0 of T_{2_1}, S0-R2 is written to R0 of T_{2_2} and so on. Next, the rows (R0 to R7) of S1 are written to R1 of T_{2_0} to T_{2_7} in a similar manner. The process continues until S7_R0 is written to R7 of T_{2_0}. T_{2_0} is now full, consequently the third 1D DCT can now be initialised. There is an initial delay of (N^3-N^2+N) clock cycles (i.e. 456 clock cycles for N=8) before the

initialisation of the third 1D DCT. This is the delay between writing S0_R0 to R0 of T_{2_0} and S7_R0 to R7 of T_{2_0}. After this initial delay, coefficients are read from transpose memory T_2 every clock cycle. A read from one location of T_2 occurs simultaneously as a write to another location of T_2, this means that the writing process continues whilst coefficients are read for the third 1D DCT.

Whilst S7_R1 is being written to the R7 of T_{2_1}, C0 of T_{2_0} is read and third 1D DCT operation commences. By this time, T_{2_1} is full. Next, C0 of T_{2_1} is read whilst S7_R2 is being written to the R7 of T_{2_2}. A read from C0 is performed on each subimage from S0 to S7, thus the process will continue until C0 of T_{2_7} is read. Figure 5 shows the content of the matrix transpose T_2, after the first set of write, i.e. after the first eight subimages have been written to the second matrix transpose.

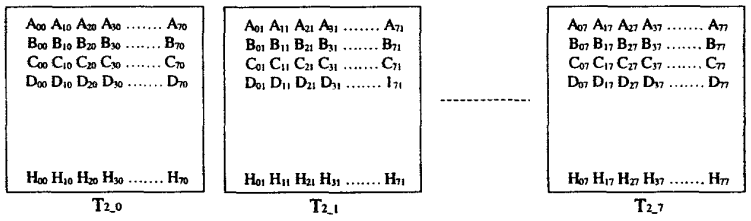

Figure 5 Pixel Sequence of the Eight Subimages after first set of write to T_2

Whilst C0 of T_{2_7} is being read, S0_R0 of the next set of eight subimages (i.e. the first subimage of the next set of eight subimages) is being written to C0 of T_{2_0}. The 'write' process is the same as that of the first set of sub images, however, this time the write is done column-wise as opposed to row-wise. This means that the rows (R0 to R7) of S0 are written to the first columns of T_{2_0} to T_{2_7}. S0_R0 is written to C0 of T_{2_0}, S0_R1 is written to C0 of T_{2_1}, S0_R2 is written to C0 of T_{2_2} and so on. Figure 6 shows the content of the matrix transpose T_2, after the second set of write, i.e. after the second eight subimages have been written to the second matrix transpose.

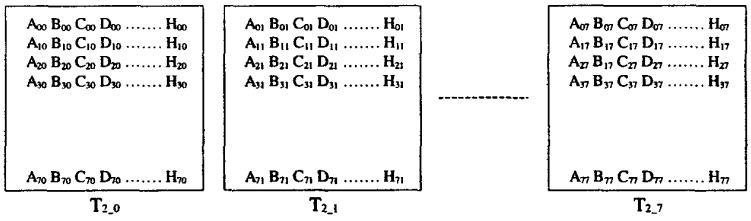

Figure 6 Pixel Sequence of the Eight Subimages after second set of write to T_2

As the write is performed column-wise, the read continues column-wise. During the write process to the first columns (C0) of T_{2_0} to T_{2_7}, the second columns (C1) of T_{2_0} to T_{2_7} are being read for the computation of the third 1D DCT. A write to a column, C_i (where i=0, 1, 2...7), of one transpose, T_{2_n} (n = 0, 1, 2... 7) occurs simultaneously as a read from the next column, C_{i+1}, of the previous transpose, $T_{2\_(n-1)}$. Thus, S0_R1 (of the second set of eight sub images) is written to C0 of T_{2_1} simultaneously as a read from C1 of T_{2_0}. After writing column-wise, the next set of read is done row-wise and this starts after writing to C7 of T_{2_0}, which indicates that T_{2_0} is full. While reading row-wise, writes are also done row-wise to the rows that have been read. This is followed by column-wise read and column-wise write to the columns that have been read. This process of switching between a row-wise read and write to a column-wise read and write then back to a row-wise read and write continues until the 3D DCT of the whole image is achieved. This process is similar to the conventional row-column transposition except that the read or write to the same memory transpose T_{2_n} is carried out every N cycle.

The outputs of the 2D DCT are written to the various transpose memory of T_2 via switches that are controlled by clock signals. The architecture has eight write switches, WS_0 to WS_7, and eight read switches, RS_0 to RS_7. The switches are activated by a positive clock edge (switch closed) and deactivated by a negative clock edge (switch open). A write switch is closed for the duration of the write (to a row or column) period; this is a period of 8 clock cycles. After which the switch opens and the switch to the next T_{2_n} transpose memory closes for the next 8^{th} clock cycles. That is at the arrival of the 2D DCT coefficients at the output of the second 1D DCT, WS_0 is closed for 8 clock cycles. During this period S0_R0 is written to R0 of T_{2_0}, at the end of the period, WS_0 opens and WS_1 closes for the next 8 clock cycles. This sequence is sustained until all eight transposes (T_{2_0} to T_{2_7}) have been written to and then the sequence starts again from T_{2_0}.

The reads from the various transpose memory, T_{2_n}, are also achieved in a similar manner. After the 456^{th} clock cycle for N=8 (this is the initial delay before the start of the first read), the RS_0 closes for the duration of the read from a row or column period of 8 clock cycles. The switch opens after the 8^{th} clock cycle and RS_1 closes for the next eight clock cycles. This sequence is also sustained until all eight transposes (T_{2_0} to T_{2_7}) have been read and then the sequence starts again from T_{2_0}.

CONCLUSION

A VLSI architecture for the implementation of the 3D DCT has been presented. The 3D DCT computation is performed using three 1D DCT and two matrix transpose operations. The second matrix transpose required prior to the third 1D DCT is composed of N memory units and two switching networks to allow appropriate read and write functions. Each memory unit performs the conventional row-column transpose. Due to its widely identical units, it will be relatively easy to implement and very suited to VLSI implementation. After an initial delay, one 3D DCT coefficient is computed every clock cycle.

REFERENCES

[1] K. R. Rao and P. Yip, *Discrete Cosine Transform; Algorithms, Advantages and Applications,* Academic Press Inc., 1990.

[2] M. Forman and A. Aggoun, "Quantisation strategies for 3D_DCT based compression of full parallax 3D images," *Proceedings of IEE 6th Internat. Conf. on Image Processing and Applications,* IPA97, No. 443, pp. 32-35, July 1997.

[3] M. Forman and A. Aggoun, "Compression of full parallax Integral 3D –TV Image Data," *SPIE/IS&T Electronic Imaging, Proceedings of SPIE conference,* vol. 3012, pp. 222–226, 1997.

[4] M. Forman, A Aggoun and M McCormick, "A novel coding scheme for full parallax 3D-TV pictures," *Proceedings of the IEEE Conference (ICASSP),* vol. 4, pp. 2945–2947, 1997.

[5] N Davies, M McCormick and M Brewin, "Design and analysis of an image transfer using microlens arrays," *Optical Engineering,* vol. 33, pp. 3624–3633, Nov. 1994.

[6] N Davies, M McCormick and Li Yang, "3D Imaging Systems ; A New Development," *Appl. Opt. 17,* pp. 4520 – 4528, 1988.

[7] G Lippmann, "Epreuves Reversible," *Photog. Integr. Comp. Rend.,* vol. 146, pp. 446 – 451, 1980.

[8] A. Madisetti and A. N. Willson, "A 100Hz 2-D 8 x 8 DCT/IDCT Processor for HDTV Applications," *IEEE Trans. on Circuits and Systems for Video Tech.,* vol. 5, no. 2, April 1995.

[9] Yu-Tai Chang and Chin-Liang Wang, "New Systolic Array Implementation of the 2D Discrete Cosine Transform and its Inverse," *IEEE Trans. on Circuits and Systems for Video Tech.,* vol. 5, no. 2, April 1995.

[10] G. S. Taylor and G. M. Blair, "Design for the Discrete Cosine Transform in VLSI," *IEE Proc. Comput. Digit. Tech.,* vol. 145, no. 2, March 1998.