

A Computationally Efficient IDCT Algorithm

Seong-Ok Bae and Seung-Jai Min
DSP Group, Information Technology Lab. LG Corporate Institute of Technology
16 Woomyen-Dong, Seocho-Gu, Seoul, 137-140, KOREA

Abstract—This paper describes a computationally efficient inverse discrete cosine transform (IDCT) algorithm using the sparsity of IDCT input coefficients. IDCT input has many zero-valued coefficients by nature. Computing only non-zero valued coefficients results in a significant reduction in IDCT computation. Our new row-column decomposition based algorithm is suitable for calculating only non-zero valued coefficients in IDCT input. The simulation results show that the new algorithm is remarkably faster than conventional algorithms based on row-column decomposition. The fast computation feature enabled the use of only one 1-D IDCT unit for 2-D transform and the reduction of power consumption.

I. INTRODUCTION

Inverse discrete cosine transform (IDCT) is an integral part of the video decoder in several standards such as JPEG, MPEG, H.261 for video telephony and HDTV system. In designing such decoders, IDCT is a computationally intensive operation. To meet the required throughput for IDCT operation, its hardware design usually employs a dedicated parallel architecture and occupies a considerable area. Therefore designing a fast and compact IDCT core is indispensable for low cost image processing system.

We introduce a new IDCT algorithm which greatly reduces the number of IDCT computation cycle, and this is accomplished by the utilization of the inherent IDCT input characteristic. The input of video decoder is a compressed bitstream and most high frequency coefficients are quantized to be zero which have no effect on the IDCT final results. It is reported that more than 80% of 2-D IDCT input coefficients are zeros for the commonly used images in the image processing field [1]. Table I depicts the percentage of zeros in the IDCT input coefficients of widely used MPEG2 sequences. It is apparent that most input coefficients are zeros. Although the number of zeros is decreased after the first 1-D IDCT operation, there are still considerable amount of zeros in the input of the second 1-D IDCT. As shown in the table, by computing only non-zero coefficients of IDCT input, the number of operation to perform IDCT can be greatly reduced by a factor of 2 to 10.

Because of its regular structure, the most commonly used IDCT method is row-column decomposition. This

TABLE I
Zero coefficient percentage of MPEG2 images

Images (bitrate)	First 1-D IDCT			Second 1-D IDT		
	I	P	B	I	P	B
flower(4Mbps)	80%	94%	97%	62%	69%	81%
football(4Mbps)	85%	95%	97%	69%	72%	82%
football(6.5Mbps)	82%	89%	96%	55%	71%	90%
cheer(15Mbps)	58%	85%	90%	44%	46%	58%
average	76%	91%	95%	58%	65%	78%

method allows the 2-D IDCT to be implemented with two 1-D IDCTs. Since conventional row-column decomposition based algorithms require N point input coefficients simultaneously to compute IDCT, there is no chance to selectively read non-zero input coefficient. These algorithms have to process all $N \times N$ input coefficients [2] [3]. Most of the previous researches are focused on the structure which supports both DCT and IDCT computation and the inherent characteristic of IDCT input coefficient is not considered. Recently, IDCT architectures exploiting this characteristic have been reported [1][4]. These architectures perform 2-D $N \times N$ IDCT directly so that they accumulate the effect of non-zero valued coefficients to all $N \times N$ outputs. However, these structures have some drawbacks for VLSI implementation and cost effective system because of the hardware cost for N^2 accumulator array and the complicated data bus interconnection between these accumulators and output buffer.

In this paper, we present a fast IDCT algorithm and its compact VLSI architecture. Row-column decomposition method is applied to achieve a regular hardware structure, and a symmetry of IDCT kernel matrix is used to halve the number of multipliers. The hardware of the proposed algorithm requires $N/2$ multipliers and N accumulators.

II. ALGORITHM DESCRIPTION

The 2-D IDCT can be defined in a separable form as a product of :

$$Z = \underline{CXC}^T = YC^T = (\underline{CY}^T)^T \quad (1)$$

where C denotes the cosine kernel matrix, X and Y^T are the input matrix of the first and second 1-D IDCT, respectively and Z is the output matrix. Figure 1 illustrates the computation concept of conventional IDCT algorithms. An element, Y_{uv} of 1-D IDCT output matrix is computed as inner product between u -th row vector

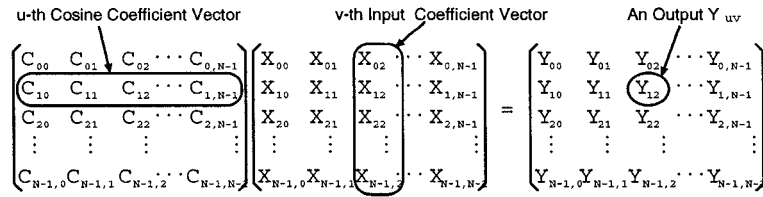


Fig. 1. IDCT computation concept of conventional algorithms

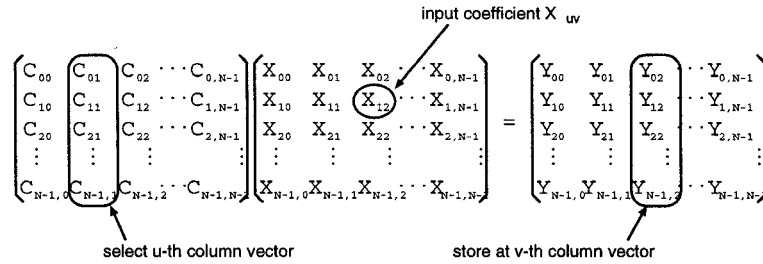


Fig. 2. IDCT computation concept of new IDCT algorithms

of cosine kernel matrix and v -th column vector of input matrix. Because one dimensional vector is used as basic input unit in conventional algorithm, zero-valued characteristic of each input element can not be considered in IDCT computation. To utilize the sparsity of input matrix, we should focus on each input element instead of a group of input elements.

Figure 2 represents another point of view to IDCT computation. Focusing on an incoming input coefficient X_{uv} , we can see that input coefficient X_{uv} is multiplied only by the u -th column elements of the cosine matrix and affects the v -th column elements of the output matrix. Every input coefficient is read, multiplied and accumulated to the output until all the input coefficients are accessed. Then we obtain the same $N \times N$ output as conventional method. The difference is that we can selectively read IDCT input coefficient so that only non-zero valued coefficients are calculated. This is the key idea of this algorithm.

The multiplication between the non-zero input X_{uv} and elements of the u -th column of the cosine kernel matrix can be represented as follows:

$$Y_{nv}^u = C_{nu} X_{uv} \quad n = 0, 1, \dots, N - 1 \quad (2)$$

$$\text{where } C_{nu} = \sqrt{\frac{2}{N}} C_u \cos\left(\frac{(2n+1)u\pi}{2N}\right),$$

$$C_u = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } u = 0 \\ 1 & \text{otherwise} \end{cases}$$

where C_{nu} is one of the u -th column elements of the kernel matrix and Y_{nv}^u is the partial product which represents the effect of the non-zero input X_{uv} to the output Y_{nv} . The element of 1-D IDCT output matrix, Y_{nv} , is obtained by accumulating these partial products.

$$Y_{nv} = \sum_{u \in \{X_{uv} \neq 0\}} Y_{nv}^u \quad \text{where } n = 0, 1, \dots, N - 1 \quad (3)$$

In equation 3, only the partial outputs generated by non-zero inputs are accumulated. Therefore, computation cycles to complete the proposed IDCT algorithm is proportional to the number of non-zeros in the IDCT input.

$$C = \frac{1}{2} \begin{pmatrix} A & B & C & D & A & E & F & G \\ A & D & F & -G & -A & -B & -C & -E \\ A & E & -F & -B & -A & G & C & D \\ A & G & -C & -E & A & D & -F & -B \\ A & -G & -C & E & A & -D & -F & B \\ A & -E & -F & B & -A & -G & C & -D \\ A & -D & F & G & -A & B & -C & E \\ A & -B & C & -D & A & -E & F & -G \end{pmatrix} \quad (4)$$

$$\text{where } \begin{cases} A = \cos(\pi/4), & B = \cos(\pi/16), & C = \cos(\pi/8), \\ D = \cos(3\pi/16), & E = \cos(5\pi/16), & F = \cos(3\pi/8), \\ G = \cos(7\pi/16), \end{cases}$$

Symmetry of the kernel matrix C is also exploited to reduce the number of multiplications. The symmetry of IDCT kernel matrix C for $N=8$ is illustrated in equation 4 where even columns are even-symmetric and odd columns are odd-symmetric. The magnitude of lower $N/2$ elements of each column can be derived from the upper $N/2$ elements and their signs are determined by whether a variable u of X_{uv} is even or odd. This symmetry reduces the number of multiplications by half.

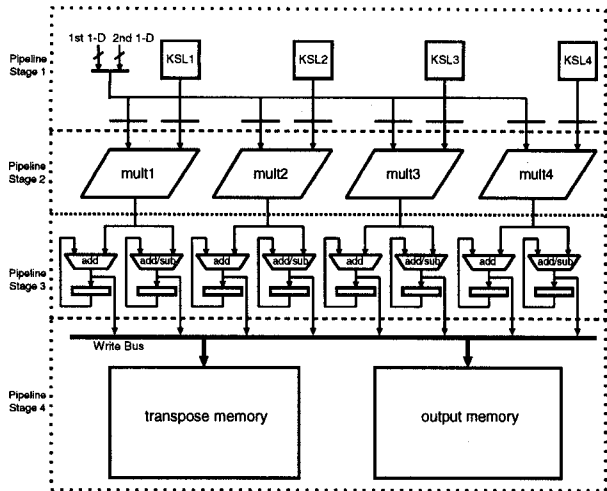


Fig. 3. 8×8 1-D IDCT hardware block for proposed algorithm

III. VLSI ARCHITECTURE

The configuration of 2-D IDCT architecture using row-column decomposition can be either a cascaded connection of first 1-D IDCT block, transpose memory, and second 1-D IDCT block or a multiplexed form of 1-D IDCT block. The latter structure was chosen to halve the hardware cost without doubling the clock frequency. Figure 3 shows the hardware realization of the proposed 8×8 IDCT algorithm. The non-zero input is fed one by one and is dispatched to each multiplier as one operand. The other operand of each multiplier is selected from the corresponding kernel select logic ($KSL_1, KSL_2, \dots, KSL_4$). Four kernel select logics receive the row index u of input X_{uv} and generate the upper 4 elements of the u -th column in the kernel matrix. These outputs of kernel select logics are scaled by multipliers and then accumulated in the accumulator registers. Because cosine coefficient matrix has the even symmetry for even columns and odd symmetry for odd columns, the accumulator function is selected between addition and subtraction according to the row index u of the input coefficient X_{uv} .

The accumulator registers are initialized to zero before every first non-zero element of each input vector is accessed. After all non-zero elements of the input vector are processed, the content of the accumulator register is stored in the output buffer through the write bus.

Because of the proposed algorithm's computational efficiency, the hardware cost is greatly reduced by the use of one 1-D IDCT unit for 2-D IDCT without doubling the clock frequency. Clocking at 27Mhz, the IDCT core can meet the the MPEG2 real time processing requirement. The hardware architecture for the proposed algorithm has been implemented into layout using $0.5 \mu m$ CMOS library. Due to its simple and regular hardware structure, the clock speed of IDCT core can be increased to handle the currently proposed HDTV systems without changing its architecture and with minor increase.

IV. SIMULATION RESULTS

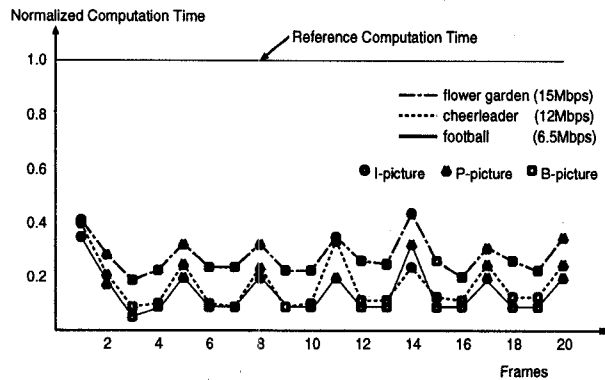


Fig. 4. 2-D IDCT computation cycle time of MPEG2 image sequences

We simulated various MPEG2 image sequences to demonstrate the performance of the proposed algorithm. Figure 4 describes the required computation cycles to perform IDCT of each frame. Each frame consists of 1350 macroblocks (MBs) and each macroblock contains six 8×8 subblocks, four for luminance and two for chrominance. A conventional algorithm based on row-column decomposition is selected as a reference which has one pixel per cycle throughput. A reference processing cycle time to complete 1 frame IDCT data is 1036800 ($1350 \text{ MBs/frame} \times 6 \text{ subblocks/MB} \times 64 \text{ pixels/subblock} \times 2 \text{ 1-D IDCTs}$) cycles and is normalized to 1.0 in the figure. As seen in the figure, the proposed algorithm is faster than the reference by a factor of 2 to 10.

The computation cycle time of the proposed algorithm varies according to the picture type. The proposed IDCT processing time increases in the I-picture and decreases in the P and B pictures. This feature indicates that there are more non-zero IDCT input coefficients in the I-pictures. The processing time difference between I-picture and other picture types such as P and B pictures enables us to utilize the IDCT hardware more efficiently. The P and B pictures require motion compensation as well as IDCT while the I-picture needs only IDCT computation. Thus we can share the adders in the IDCT hardware to perform motion compensation using the time acquired by the relatively faster IDCT operation in P and B pictures. This feature reduces the hardware cost by sharing IDCT and motion compensation hardware.

V. CONCLUSION

In this paper, we proposed a computationally efficient IDCT algorithm by calculating only non-zero IDCT input coefficients instead of all $N \times N$ input matrix. The proposed algorithm enables a significant computation reduction compared with the conventional row-column decomposition algorithms. Based on row-column decomposition and symmetry of cosine coefficient matrix, the regular

hardware structure is also obtained. We also utilize the IDCT processing time difference between I-picture and other picture types to share the IDCT hardware and this feature makes the decoding hardware more efficient. The fast computation time of the proposed algorithm permits the use of only one 1-D IDCT unit for the 2-D transform and this results in a compact VLSI hardware for low cost image processing system. With the efficient computation algorithm and the hardware architecture, the proposed IDCT algorithm can be used in various image processing systems.

REFERENCES

- [1] L. McMillan and L. Westover, "A forward-mapping realization of the inverse discrete cosine transform," *Proc. DCC'92 Data Compression Conference*, pp.219-228, June 1992.
- [2] S. Uramoto et al, "A 100MHz 2-D discrete cosine transform core processor" *IEEE J. Solid-State Circuits*, vol. 27, pp.492-499, Apr. 1992.
- [3] A. Madisetti and A. N. Willson, Jr., "A 100MHz 2-D 8x8 DCT/IDCT Processor for HDTV applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol.5, no. 2, pp. 158-165, Apr. 1995.
- [4] J. F. Yang, B. L. Bai, and S. C. Hsia, "An efficient two-dimensional inverse discrete cosine transform algorithm for HDTV receivers," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 1, pp. 25-30, Feb. 1995.
- [5] IEEE Standard Specification for the Implementation of 8×8 Inverse Discrete Cosine Transform, 1990.