The 23rd International Technical Conference on Circuits/Systems,
Computers and Communications (ITC-CSCC 2008)

A Multi-Dimensional Transform for Future Video Coding

Woo-Jin Choi[1], Su-Yeol Jeon[1], Chang-Beom Ahn[2], and Seoung-Jun Oh[1]
[1] Department of Electronics Engineering , Kwangwoon University
447-1, Wolgye-Dong, Nowon-Gu, Seoul, Korea
Tel: +82-2-940-5102, Fax: +82-2-943-5381
[2] Department of Electrical Engineering , Kwangwoon University
447-1, Wolgye-Dong, Nowon-Gu, Seoul, Korea
Tel: +82-2-940-5148, Fax: +82-2-943-5381
E-mail: [1]wjchoi@media.kw.ac.kr, [1]sjoh@media.kw.ac.kr

Abstract: In this paper, we propose a Multi-Dimensional Transform(MDT) technique for future video coding. Unlike traditional video codecs, H.264/AVC uses an integer 4×4 block transform. Using small block size coding, H.264/AVC obtains high temporal prediction efficiency, but it has limitation in utilizing spatial redundancy. The proposed MDT is a new method for additionally reducing spatial redundancies of image. Experimental results indicate MDT can provide better R-D performance than the conventional H.264/AVC transform.

1. Introduction

H.264/AVC is a novel video coding technology standardized by both ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group. In H.264/AVC, an integer 4×4 block transform is used in an intra as well as an inter prediction mode instead of the conventional 8×8 block transform.[1],[2] This allows the encoder to represent signals in a more locally-adaptive fashion, which reduces artifacts known colloquially as "ringing". Even though H.264/AVC obtains high temporal prediction efficiency using a small size block, it inherently has a limitation in utilizing spatial redundancy because of that small block size.

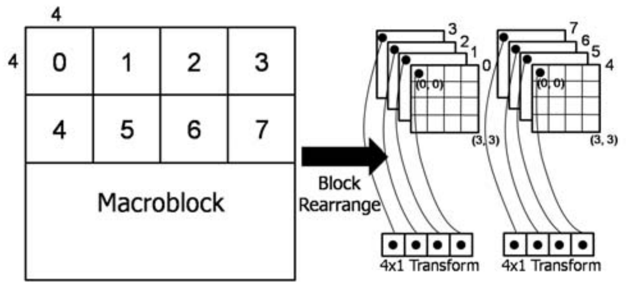
Hence, a new method for additionally reducing spatial redundancy has to be considered for future video coding. We propose a multi-dimensional transform (MDT) which can provide bit-rate reduction by reducing spatial redundancy with keeping H.264/AVC temporal redundancy reduction accuracy[2-5]. The proposed MDT achieves higher energy compactness than H.264 2-dimensional modified DCT.[3],[5] We design an integer-based transform and a quantizer for the MDT as well. The rest of this paper is organized as follows. In section 2, we provide a detailed description of proposed MDT. Experimental results are described in section 3. Finally, conclusion is given in section 4.

2. Multi-Dimensional Transform Scheme

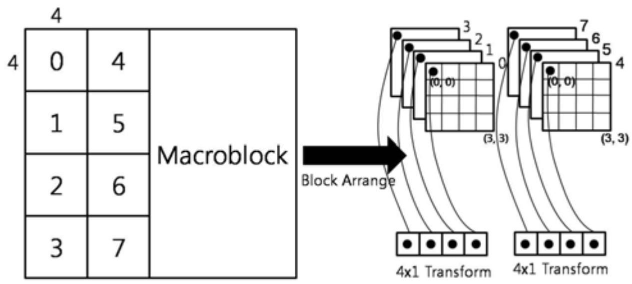
Since there exists strong spatial correlation among neighbor blocks in a video sequence [4], which means that the transform is more effective in a block as large as possible, there may still exist spatial redundancy after a regular H.264/AVC transform and quantization. In the proposed MDT, an additional transform process is applied to each rearranged 4x4 coefficient block after an integer 4x4 transform is applied to the corresponding 4x4 block and transform coefficients are rearranged at a macro block level. We propose several types of coefficient rearrangement, which decides the dimensionality of the MDT: 3- and 4-dimensions.

There are two types in the 3-D transform (3DT): horizontal direction (H3DT) and vertical direction (V3DT). The H3DT and the V3DT are applied to a 16x8 and an 8x16 submacroblock, respectively. We will explain the H3DT only in this paper since the V3DT is the same as the H3DT except the direction. There exist two 4x4 block rows in the 16x8 submacroblock.

The block diagrams of a H3DT and a V3DT operation are shown in Figure 1.



(a) H3DT



(b) V3DT

Figure 1. Block diagrams of H3DT and V3DT

The first H3DT step is to construct sixteen 4x1 transform coefficient arrays at each 4x4 block row just after 4x4 H.264/AVC transforms at the submacroblock as shown in Figure 1(a). Every block in Figure 1 consists of 2-D transform (2DT) coefficients. The second step is to execute each 4x1 1-D transform (1DT). At the third step, sixteen coefficients corresponding to the same position among sixteen 4x1 blocks are collected in a 4x4 block. Finally, four 4x4 blocks are constructed. These three steps are executed twice in each submacroblock. The process for the V3DT is the same as the above except the direction as shown in Figure 1(b).

The first 4DT step is to construct sixteen 4x4 transformed blocks shown in Figure 2, each of which consists of sixteen coefficients taken at the same spatial frequency in sixteen 4x4 blocks of the macroblock, just after sixteen 4x4 H.264/AVC transforms at each macroblock. The second step is to execute a 2DT at each 4x4 transform coefficient block. Thus, each result is the transform of 4x4 coefficients in the corresponding spatial frequency band.

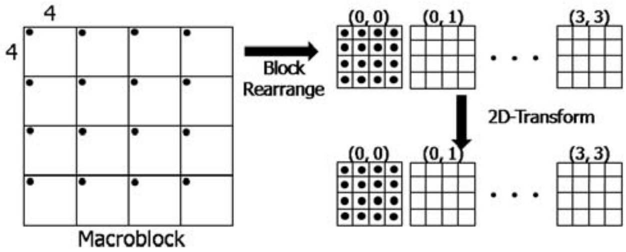


Figure 2. Block diagrams of 4DT

2.1 Integer MDT and Quantization

In H.264/AVC, an integer 4×4 block transform is used. This transform is based on the DCT but with some fundamental differences[2]:

1. It is an integer transform (all operations can be carried out with integer arithmetic, without loss of accuracy).
2. The inverse transform is fully specified in the H.264 standard and if this specification is followed correctly, mismatch between encoders and decoders should not occur.
3. The core part of the transform is multiply-free, i.e. it only requires additions and shifts.
4. A scaling multiplication (part of the complete transform) is integrated into the quantizer (reducing the total number of multiplications).

To implement The MDT, the above fundamental differences are also considered. Thus, it uses only integer calculation without a scaling multiplication.

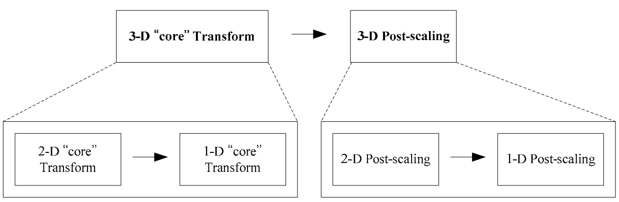


Figure 3. Transform and Scaling in the 3DT

To implement integer calculation of transformation, the MDT can be divided into a core-transform part and a post-scaling part. In the 3DT case, a core-transform part consists of H.264/AVC integer transform and additional 1D transform, and a post-scaling part can be separated into two sub-parts as shown in Figure 3. The general 4x4 forward DCT transform can be calculated as (1)[3-5]:

$$Y = AXA^T \tag{1}$$
$$= \begin{bmatrix} a & a & a & a \\ b & c & -c & -b \\ a & -a & -a & a \\ c & -b & b & -c \end{bmatrix} \begin{bmatrix} x_{00} & x_{01} & x_{02} & x_{03} \\ x_{10} & x_{11} & x_{12} & x_{13} \\ x_{20} & x_{21} & x_{22} & x_{23} \\ x_{30} & x_{31} & x_{32} & x_{33} \end{bmatrix} \begin{bmatrix} a & b & a & c \\ a & c & -a & -b \\ a & -c & -a & b \\ a & -b & a & -c \end{bmatrix}$$

where A is the transform matrix whose entries are

$$a = 1/2$$
$$b = \sqrt{1/2} \cdot \cos(\pi/8)$$
$$c = \sqrt{1/2} \cdot \cos(3\pi/8)$$

and Y is the transformed 4x4 coefficients matrix. The 4x4 DCT is substantially modified for a separated integer calculation in H.264/AVC as (2).

$$Y = \left(CXC^T\right) \otimes E_f = \tag{2}$$
$$\left(\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} \begin{bmatrix} x_{00} & x_{01} & x_{02} & x_{03} \\ x_{10} & x_{11} & x_{12} & x_{13} \\ x_{20} & x_{21} & x_{22} & x_{23} \\ x_{30} & x_{31} & x_{32} & x_{33} \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{bmatrix} \right) \otimes \begin{bmatrix} a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \\ a^2 & \frac{ab}{2} & a^2 & \frac{ab}{2} \\ \frac{ab}{2} & \frac{b^2}{4} & \frac{ab}{2} & \frac{b^2}{4} \end{bmatrix}$$

CXC^T is a core 2DT. E_f is a matrix of scaling factors and the symbol $\otimes$ indicates that each element of CXC^T is multiplied by the scaling factor in the same position in matrix E_f. The 3DT can be represented by adding both the 1DT and the 1-D scaling process as (3) where W is the calculated matrix of CXC^T.

$$Y_{3DT} = (CW) \otimes E_f \otimes E_{1DT} \tag{3}$$
$$= \left(\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} \begin{bmatrix} W_{00} & W_{01} & W_{02} & W_{03} \\ W_{10} & W_{11} & W_{12} & W_{13} \\ W_{20} & W_{21} & W_{22} & W_{23} \\ W_{30} & W_{31} & W_{32} & W_{33} \end{bmatrix} \right) \otimes E_{3DT}$$

The 3DT can be represented by (4), where R_T is a matrix after the core 2DT and the additional 1DT.

$$\begin{aligned} Y &= W \otimes E_f \\ Y_{3DT} &= R_T \otimes E_{3DT} \end{aligned} \tag{4}$$

The scaling process for the 3DT can be represented by (5).

$$E_{3DT} = \begin{bmatrix} aE_f & \frac{b}{2}E_f \\ \frac{b}{2}E_f & aE_f \end{bmatrix}$$

$$= \begin{bmatrix} \alpha & \beta & \alpha & \beta & \beta & \delta & \beta & \delta \\ \beta & \delta & \beta & \delta & \delta & \rho & \delta & \rho \\ \alpha & \beta & \alpha & \beta & \beta & \delta & \beta & \delta \\ \beta & \delta & \beta & \delta & \delta & \rho & \delta & \rho \\ \beta & \delta & \beta & \delta & \alpha & \beta & \alpha & \beta \\ \delta & \rho & \delta & \rho & \beta & \delta & \beta & \delta \\ \beta & \delta & \beta & \delta & \alpha & \beta & \alpha & \beta \\ \delta & \rho & \delta & \rho & \beta & \delta & \beta & \delta \end{bmatrix}_{8\times 8} \quad (5)$$

$$\alpha = a^3, \beta = \frac{a^2 b}{2}, \delta = \frac{ab^2}{4}, \rho = \frac{b^3}{8}$$

Z_{ij} in (6) is a quantized value of each 3DT coefficient, where *Qstep* is a quantization step size.

$$Z_{ij} = round\left(\frac{Y_{ij}}{Qstep}\right) = round\left(\frac{R_T^{ij} \otimes E_{3DT}^{ij}}{Qstep}\right) \quad (6)$$

Z_{3DT}^{ij} in (8), the final value as the result of the transform and quantization module, can be expressed by substituting (7) for (6) using *qbits* and MF_{3DT}^{ij}.

$$\frac{E_{3DT}^{ij}}{Qstep} = \frac{MF_{3DT}^{ij}}{2^{qbits}}, \text{ where } qbits = 15 + floor(QP/6) \quad (7)$$

$$Z_{3DT}^{ij} = (\left|R_T^{ij}\right| \cdot MF_{3DT}^{ij} + F) >> qbits \quad (8)$$

$$F = \begin{cases} 2^{qbits}/3, & \text{intra block} \\ 2^{qbits}/6, & \text{inter block} \end{cases}, \quad sign(Z_{3DC}) = sign(W)$$

Since we can easily expand the 3DT to the 4DT, the detail process for the 4DT is not shown in this paper.

2.2 An MDT type Selection Scheme

We design the MDT which can be applied in case that a chosen block mode is one of 16x16, 16x8, and 8x16, that is, neighboring 4x4 blocks have strong spatial correlation. In H.264/AVC, variable block-size motion compensation has been adopted.[1,2] There exist 7 different sizes from 4x4 up to 16x16. According to our various experiments, the redundancy is sufficiently removed by the H.264 transform and quantization in case of either smaller than or equal to the size of 8x8. Thus, without losing any generality, we can insist that there exists little correlation in residual values if the block size chosen as the best prediction mode is equal to or smaller than 8x8. However, we can investigate that there still exists strong correlation as much as making some effect in terms of coding efficiency in residual values of the block whose size is larger than 8x8. Therefore, the proposed MDT is applied to the larger than 8x8 block: 16x8, 8x16, and 16x16. The 16x8 and the 8x16 sub-macroblock inherently have strong correlations in the horizontal and vertical direction, relatively, while the 16x16 macroblock has in both directions. According to those characteristics, an MDT type selection scheme can be suggested; in case of the sizes of 16x8 and 8x16, the H3DT and the V3DT shown in Figure 1 are selected, respectively. The 4DT is applied to the size of 16x16. A new encoder based on the MDT is shown in Figure 4.

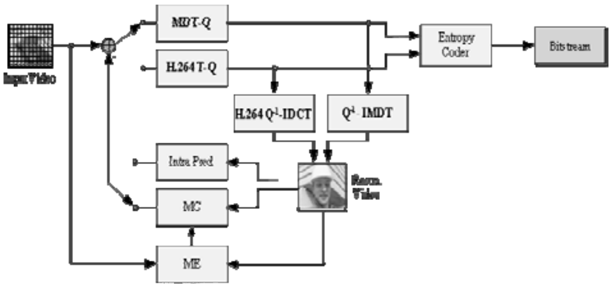
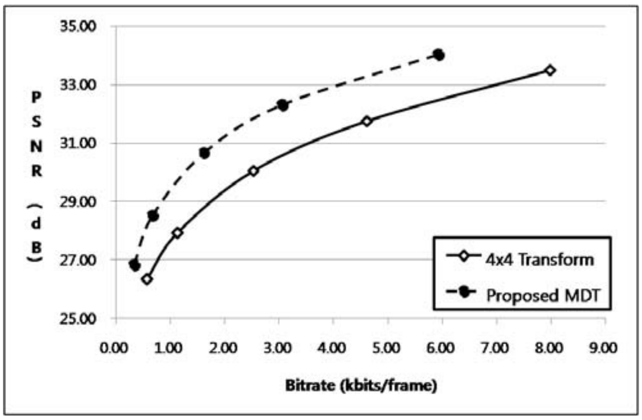


Figure 4. A block diagram of the video encoder based on the MDT

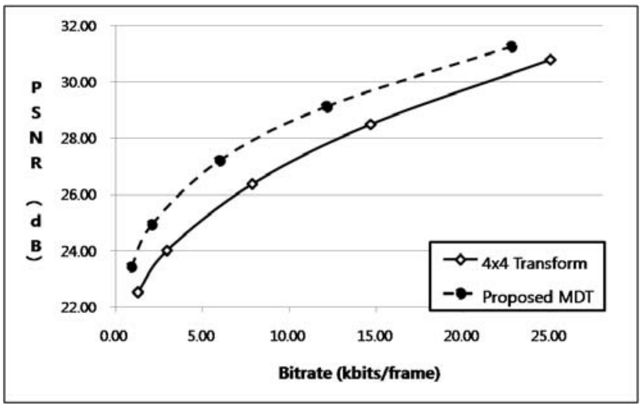
3. Experimental Results

We have implemented the proposed method on the JM 10.2[6] and performed extensive experiments to verify the effectiveness of the proposed MDT. We will show in this section the high energy compactness of the MDT in 4 different test sequences: Foreman, Harbour, Carphone, and Container. An experimental platform is a Windows XP PC with Pentium 4 3.2-GHz CPU and 2–GB random access memory (RAM). In our experiments, a baseline profile in H.264/AVC is considered.[1] All test sequences consist of 300 frames at CIF resolution encoded at 30 fps, and only the first frame is an intra frame. A group of experiments are carried out on the test sequences with the 5 different quantization parameters (QP): 32, 35, 38, 42, and 45. Figure 5 shows that the proposed MDT has outperformed the H.264/AVC transform and quantization in terms of energy compactness.

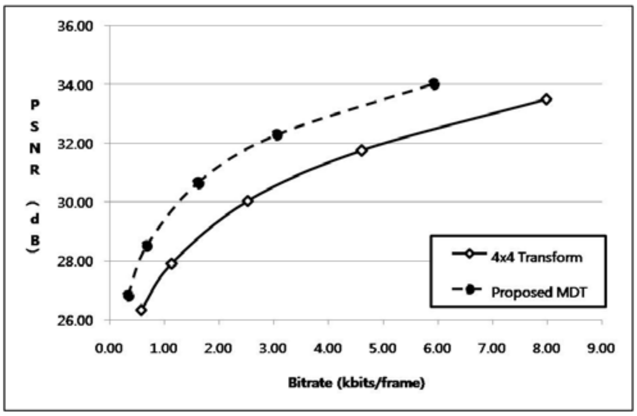
Since H.264/AVC entropy coders such as CAVLC and CABAC may not be suitable for the proposed MDT, it is hard to compare it with the H.264/AVC encoder directly. Thus, total bits per frame of the x-axis in Figure 5 is computed using both the entropy of quantized transformed coefficients and the bits for header information.



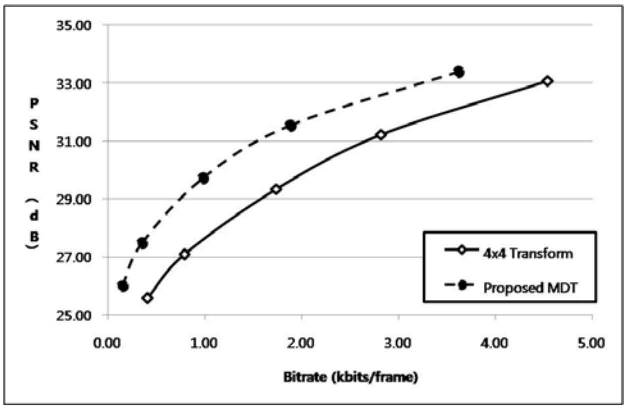
(a) Foreman sequence (CIF)



(b) Harbour sequence (CIF)



(c) Carphone sequence (CIF)



(d) Container sequence (CIF)

Figure 5. R-D performances (X- and y-axes represent the measured total bits of each frame and the PSNR, respectively).

4. Conclusion

In this paper, we proposed a high energy compactness MDT which can generally increase coding efficiency in a future video codec. Three modes of the MDT were designed: H3DT, V3DT and 4DT. The MDT mode was chosen according to the macroblock type. Experimental results showed that the MDT can provide better R-D performance than the conventional H.264/AVC transform at the value of QP>24 .

Acknowledgements

This work was supported by grant from Research Grant of Kwangwoon University, grant No.60012007179 and Electronics and Telecommunications Research Institute, grant No.90262007023 in 2007 .

References

[1] ISO/IEC 14496-10 and ITU-T Rec. H.264, Advanced Video Coding, 2003.
[2] T.Wiegand, Gary J. Sullivan, and A.Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. on Circuit and Systems for Video Tech.*, vol. 13, no.7, pp. 560-576, July 2003.
[3] A.Hallapuro, M. Karczewicz and H. Malvar, "Low Complexity Transform and Quantization – Part 1: Basic Implementation," JVT document JVT-B038, Geneva, February 2002.
[4] Iain E. G. Richardson, H.264 and MPEG-4 Video Compression, John Willey & Sons Ltd, England, 2003\
[5] K. R. Rao and P. Yip. *Discrete Cosine Transform: Algorithms, Advantages, Applications.* Boston: Academic Press, 1990, Chapter 4.
[6] H.264/AVC Reference Software Model (JM 10.2):http://iphome.hi.de/uehring/ml/download/old_jm/