A Fast DCT Domain Downsampling Technique for Video Transcoder

Nam-Hyeong Kim, Yoon Kim, Goo-Rak Kwon,
and Sung-Jea Ko, *Senior Member, IEEE*

Department of Electronics Engineering, Korea University, Seoul, Korea
Email: nhkim@dali.korea.ac.kr

***Abstract*— In this paper, we propose a fast DCT domain downsampling technique for video transcoder. Experimental results show that the proposed method is not only computationally efficient, but also gives significant improvements in PSNR.**

I. Introduction

Video transcoding is an operation for converting a video sequence from one format to another, where the input and output video sequences are compressed video bit streams compliant with existing video compression standards. If the bandwidth required for a particular video is not available due to congestion or other causes, the video must be recoded at a reduced bit rate. A solution to this problem is to generate a new compressed video with a lower spatial resolution from the original precoded video bit stream. Conventional methods to downscale a video sequence decompress the video bit stream, perform the downscaling in the pixel domain, and then recompress it for efficient delivery. This method requires computationally intensive DCT/IDCT operation. To reduce the computational load, some downsampling methods in the DCT domain are proposed [1], [2], [3], [4]. These methods require computationally inefficient matrix multiplication for downsampling, eliminating any gains obtained by operating directly in the DCT domain.

In this paper, we propose a fast method to downsample in the DCT domain. Compared with conventional pixel domain methods, the proposed downsampling method is not only computationally more efficient, but also gives significant improvements in PSNR.

II. Proposed Downsampling Method

Fig. 1 shows the system diagram of DCT domain downsampling based on the block-based video standards. It consists of three main parts: partial decoding in the DCT domain, downsampling in DCT domain, and reencoding. Since, unlike the pixel domain method, this method does not require full decompression and recompression, the complexity and time delay can be significantly reduced. In Fig. 1, the DCT-MC1 block performs DCT domain motion compensation (DCT-MC) using incoming motion vectors and the reference DCT frame stored in the frame memory (FM). The DCT-MC2 block performs DCT-MC with the refined motion vector obtained by the motion vector refinement (MVR) operation.

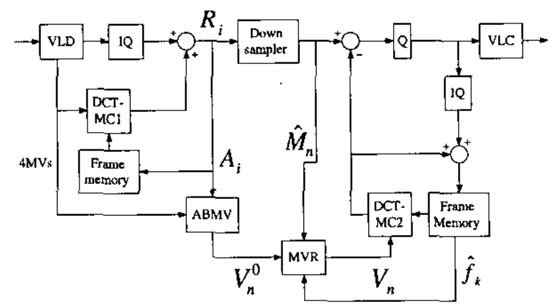

Fig. 1. System diagram of proposed DCT domain downsampling.

The proposed downsampling approach is processed directly in the DCT domain. The outline of our scheme is shown in Fig. 2 for 1-D signals. Let B_1 and B_2 denote 8-

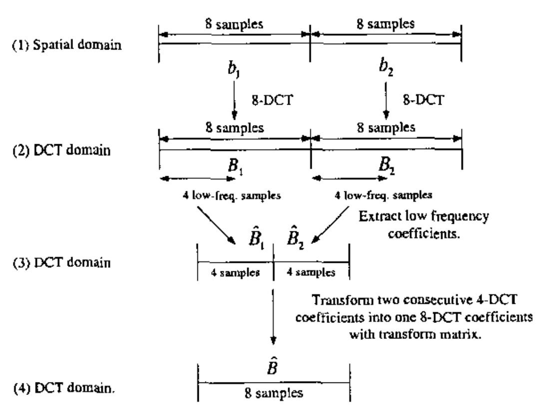

Fig. 2. Downsampling in the DCT domain.

point DCT coefficients of two consecutive 8-sample blocks b_1 and b_2. Downsampling (b_1, b_2) with an appropriate transform filter produces the 8-point DCT $\hat{B}$ of the 8-point block. In principle, the proposed scheme can be viewed as follows: Extract four low-pass coefficients in B_1 and B_2. Concatenate these two consecutive 4-point blocks and then transform two 4-point DCT coefficients into one 8-point DCT coefficients with transform matrix A as follows:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ a & b & -c & d & -a & b & c & d \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ -e & f & g & -h & e & f & -g & -h \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ i & -j & k & l & -i & -j & -k & l \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ -m & n & o & p & m & n & -o & p \end{pmatrix}, \quad (1)$$

where a = 0.9061, b = 0.4157, c = 0.0747, d = 0.0229, e = 0 .3182, f = 0.7911, g = 0.5133, h = 0.0975, i = 0.2126, j = 0 .3524, k = 0.7682, l = 0.4904, m = 0.1802, n = 0.2778, o = 0.3753, and p = 0.8657. Then, the resulting block becomes the desired downsampled block $\hat{B}$. Note that this series of operations (starting from B_1 and B_2 in DCT-domain to $\hat{B}$ also in DCT domain) require a significantly reduced computation as compared with matrix multiplication required for downsampling in the DCT domain. In 2-D signals, computation $\hat{B}$ is (256M + 512A), where M is a multiplication and A is a addition. Thus, (1M + 2A) is required per pixel in the original 16 × 16 image.

Given DCT coefficients of the two 4-point blocks, the DCT of their composed 8-point block can be directly derived in the DCT domain as follows: $B = \{x_i\}$, $i = 0, 1, ..., 7$, $B_0 = \{x_i\}$, $i = 0, 1, 2, 3$, $B_1 = \{x_i\}$, $i = 4, 5, 6, 7$. $C(u)$, $u = 0, 1, 2, ..., 7$, is the DCT of B, and $C_0(u)$, $u = 0, 1, 2, 3$, and $C_1(u)$, $u = 0, 1, 2, 3$, are the DCT of B_0 and B_1. From the standard N-point DCT formula, the DCT of B, B_0, and B_1 can be expressed as

$$C(u) = \frac{1}{2}\alpha(u)\sum_{i=0}^{7} x(i)\cos(\frac{(2i+1)u\pi}{16}), \quad (2)$$

$$C_0(u) = \frac{1}{\sqrt{2}}\alpha(u)\sum_{i=0}^{3} x(i)\cos(\frac{(2i+1)u\pi}{8}), \quad (3)$$

and

$$C_1(u) = \frac{1}{\sqrt{2}}\alpha(u)\sum_{i=0}^{3} x(4+i)\cos(\frac{(2i+1)u\pi}{8}), \quad (4)$$

where

$$\alpha(u) = \begin{cases} \sqrt{1/2}, & for\ u = 0, \\ 1, & otherwise. \end{cases} \quad (5)$$

(3) and (4) can be combined as follows:

$$C_p(u) = \begin{cases} \frac{1}{\sqrt{2}}\alpha(u)\sum_{i=0}^{3} x(i)\cos(\frac{(2i+1)u\pi}{8}), \\ \qquad u = 0, 1, 2, 3, \\ \frac{1}{\sqrt{2}}\alpha(u)\sum_{i=4}^{7} x(i)\cos(\frac{(2(i-4)+1)(u-4)\pi}{8}), \\ \qquad u = 4, 5, 6, 7. \end{cases} \quad (6)$$

Finally, the relationship between $C(u)$ and $C_p(u)$ can be rewritten in matrix form as follows:

$$\begin{pmatrix} C(0) \\ C(1) \\ \vdots \\ C(7) \end{pmatrix} = A * \begin{pmatrix} C_p(0) \\ C_p(1) \\ \vdots \\ C_p(7) \end{pmatrix}^T. \quad (7)$$

Because $C(u)$ and $C_p(u)$ can be cosidered as 1×8 row vectors from (2) and (6), transform matrix A can be calculated as (1).

TABLE I

PSNR VALUES AFTER DOWNSAMPLING USING BILINEAR INTERPOLATION AND OUR SCHEME.

Image (Unit : dB)	Lena	Pepper	House	Cap
Bilinear Interpolation	27.76	29.57	28.07	27.53
Proposed Method	29.59	31.38	30.31	29.23

III. Experimental Results and Conclusions

In Table I, we present the PSNR values obtained after downsampling a gray-level image using the spatial domain approach with the bilinear interpolation [1] and the proposed method. The PSNR values obtained by the proposed method is much better when compared to the spatial domain approach . Also, the proposed downsampling scheme requires 1 multiplication and 2 additions per pixel, while the spatial domain approach requires 3.44 multiplications and 9.81 additions. Experimental results indicate that the proposed method is not only computationally efficient, but also gives significant improvements in PSNR.

REFERENCES

[1] N. Merhav and V. Bhaskaran, "Fast algorithms for DCT-domain image down-sampling and for inverse motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 468–476, June 1997.

[2] B. K. Natarajan and B. Vasudev, "A fast approximate algorithm for scaling down digital images in the DCT domain," *in Conf. Rec. IEEE Int. Conf. Image Processing (ICIP)*, vol. 2, pp. 241–243 Oct. 1995.

[3] T. Shanableh and M. Ghanbari, "Heterogeneous video transcoding to lower spatio-temporal resolutions and different encoding formats," *IEEE Trans. Multimedia*, vol. 2, pp. 101–110, June 2000.

[4] S. Chang and D. Messerschmitt, "Manipulation and compositing of MC-DCT compressed video," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 1–11, Jan. 1995.