Analysis of IDCT and Motion-Compensation Mismatches Between Spatial-Domain and Transform-Domain Motion-Compensated Coders

Seung-Kyun Oh and HyunWook Park, *Senior Member, IEEE*

***Abstract*—Many studies have performed transcoding of compressed video streams in the discrete cosine transform (DCT) domain to reduce computational complexity. On the other hand, most video compression standards recommend motion compensation in the spatial domain. Thus, compressed data that is motion-compensated in the spatial domain should be motion-compensated in the transform domain for the transform-domain transcoding. However, this combination of encoding in the spatial domain and decoding in the transform domain degrades image quality due to mismatches of the inverse DCT and motion compensation. There have been few investigations into this mismatch problem. In this paper, we establish that the rounding in calculation is the major cause of the mismatch. We investigate the mismatch problem in detail and propose a precision lifting method to reduce the effects of the mismatch problem. Experimental results show that our proposed method can effectively reduce the mismatch problem.**

***Index Terms*—Inverse discrete cosine transform (IDCT), mismatch, motion compensation (MC), precision lifting (PL), rounding, transcoding.**

I. Introduction

With the increasing demand for digital media, digital videos are becoming widely available in compressed forms such as MPEG and H.26x [1]–[4]. These digital videos are used in various applications including digital TV, mobile communications, and video-on-demand (VOD). In some applications, we need various image manipulations such as filtering, bit-rate change, downsampling, and data format conversion.

There are two categories in the manipulation of digital videos according to the operation domain: one is accomplished in the spatial domain and the other is accomplished in the transform domain. The existing videos are usually stored in a compressed form that is motion-compensated in the spatial domain. To manipulate these digital video data in the spatial domain, several operations of an inverse discrete cosine transform (IDCT), a spatial-domain manipulation, a spatial-domain motion compensation (MC), and a DCT are sequentially applied to the compressed data. Thus, the spatial-domain processing has a heavy computation load of video decoding and re-encoding.

In order to reduce heavy computation load, several methods of image manipulation in the DCT domain have been introduced [5]–[10]. In DCT-domain processing, the compressed bit stream is decoded to DCT coefficients by a variable length decoder (VLD). Then, the MC is performed in the DCT domain and the reconstructed data in the DCT domain is used as a reference in the recursive motion-compensation loop. Recently, Koc *et al.* studied the mismatch problem in transcoders [11], [12]. Motion compensation of the decoder that is performed in the same domain as the encoder has no mismatch problem, whereas there is a mismatch problem when MC in a decoder is performed in a different domain from the encoder. Koc established two conditions for the mismatch problem: one is the distributive property of IDCT and the other is the commutative property of MC and IDCT. In this paper, we investigate in detail the mismatch problem and two conditions in conventional video coders such as MPEG [13], and then propose a precision lifting (PL) method to reduce the effects of the mismatch problem.

In [14], the integer transform with scaling was introduced. However, the mismatch of DCT-IDCT pair is inevitable when using previous video coding standards with DCT and IDCT. Our proposed method temporarily lifts the precision of operation in the reconstruction loop to reduce the mismatch, and the range of transformed coefficients conforms with the previous video coding standards. Although the DCT and IDCT with scaling introduced in [14] can reduce the mismatch, it is incompatible with the previous video coding standards because the range of transformed coefficients with scaling is different from that in the previous video coding standards.

This paper is organized as follows. In Section II, we introduce the mismatch problem of spatial domain and transform domain motion-compensated coders. Then, we analyze the mismatch problem and propose the PL method to reduce the effects of this mismatch problem in Section III. Experimental results are shown in Section IV. Finally, we conclude this paper in Section V.

II. Mismatch Between Spatial-Domain and Transform-Domain Motion-Compensated Coders

Koc *et al.* studied the mismatch problem between encoders and decoders with different operation domains [11], [12], where establishing two conditions to overcome the problem. In conventional encoders such as MPEG and H.26x, MC is usually performed in the spatial domain. The encoder includes a recursive loop of the image reconstruction for MC.

Figs. 1 and 2 respectively depict a spatial-domain encoder (SE) and a DCT-domain encoder, referred to as a transform-domain encoder (TE). A spatial-domain decoder (SD) and a

Manuscript received August 20, 2002; revised January 20, 2004. This work was supported in part by the university IT research center program of the government of Korea. This paper was recommended by Associate Editor H. Watanabe.

The authors are with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Korea (e-mail: hwpark@athena.kaist.ac.kr).

Digital Object Identifier 10.1109/TCSVT.2005.848353

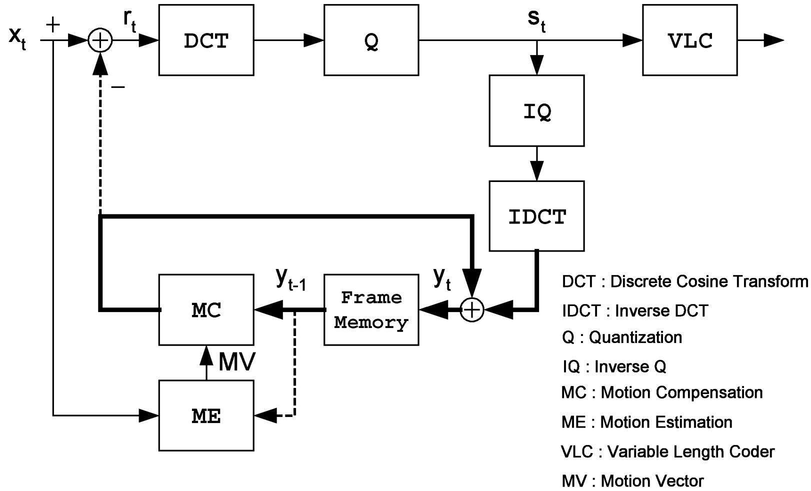


Fig. 1. Conventional encoder with spatial-domain MC (shortly, spatial-domain encoder, SE).

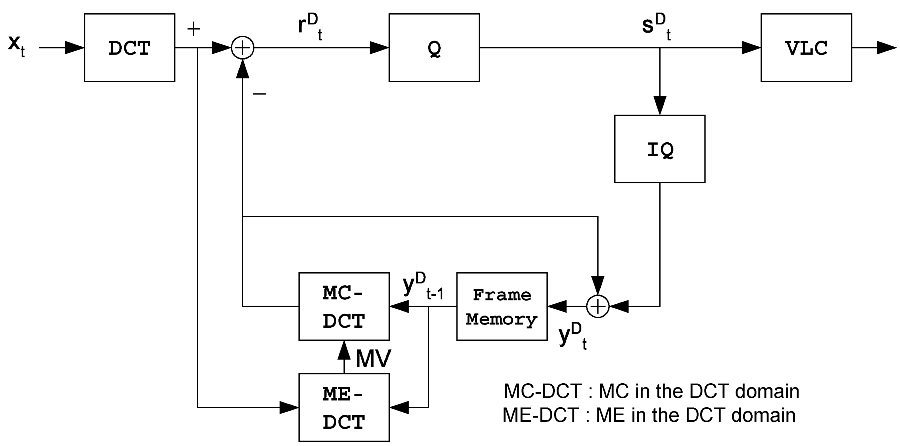


Fig. 2. Encoder with DCT-domain MC (shortly, transform-domain encoder, TE).

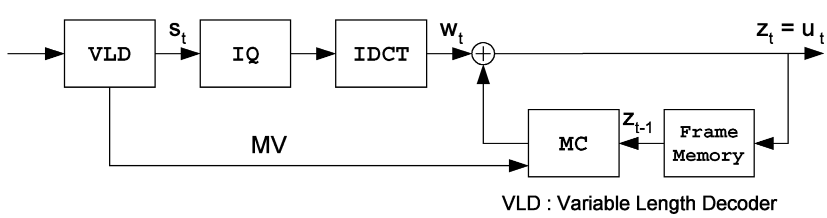


Fig. 3. Conventional decoder with spatial-domain MC (shortly, spatial-domain decoder, SD).

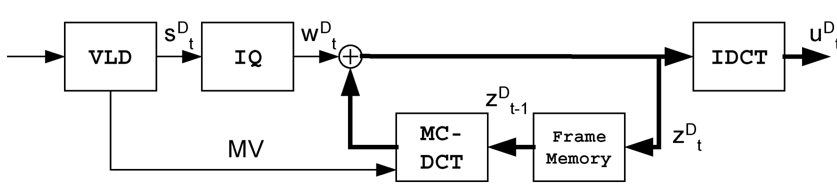


Fig. 4. Decoder with DCT-domain MC (shortly, transform-domain decoder, TD).

DCT-domain decoder, or transform domain decoder (TD), are presented in Figs. 3 and 4, respectively. A mismatch can occur in an SE-TD pair or a TE-SD pair. However, SE-SD and TE-TD pairs do not have mismatch problems.

The mismatch problem is generated when MC in the decoder is not equivalent to that in the encoder. The mathematical relation between the mismatched coders is derived as follows. The image data at each stage in SE (Fig. 1) can be described by iterative functions as follows:

- Intraframe (I-frame) in SE

$$s_t = Q\left(DCT(x_t)\right)$$
$$y_t = IDCT\left(IQ(s_t)\right), \quad \text{for } t = 0. \tag{1}$$

- Interframe (P-frame) in SE

$$s_t = Q\left(DCT\left(x_t - MC(y_{t-1})\right)\right)$$
$$y_t = IDCT\left(IQ(s_t)\right) + MC(y_{t-1}), \quad \text{for } t \geq 1. \tag{2}$$

where x_t, y_t, and s_t are the input image frame, the reconstructed frame, and the quantized DCT coefficients of the motion-compensated residual at t, respectively. In (1) and (2), $DCT(\cdot)$, $IDCT(\cdot)$, $Q(\cdot)$, and $IQ(\cdot)$ are the DCT, the IDCT, the quantization, and the inverse quantization, respectively. MC in (2) is the MC in the spatial domain. In this analysis, we do not consider B-frames.

Fig. 4 presents the DCT-domain decoder (TD). The output images can be reconstructed as follows:

- Intraframe (I-frame) in TD

$$\begin{aligned} u_t^D &= IDCT\left(IQ\left(s_t^D\right)\right) \\ z_t^D &= IQ\left(s_t^D\right), \quad \text{for } t = 0 \end{aligned} \tag{3}$$

- Interframe (P-frame) in TD

$$\begin{aligned} u_t^D &= IDCT\left(IQ\left(s_t^D\right) + MC^D\left(z_{t-1}^D\right)\right) \\ z_t^D &= IQ\left(s_t^D\right) + MC^D\left(z_{t-1}^D\right), \quad \text{for } t \geq 1 \end{aligned} \tag{4}$$

where MC^D is the MC in the DCT domain. u_t^D, z_t^D, and s_t^D are the reconstructed frame in the DCT domain, the reference frame in the DCT domain for MC, and the DCT coefficients at t that are transferred from encoder and decoded by a VLD, respectively.

Because a variable length coder (VLC) is a lossless coder, the following equality is satisfied in the SE-TD pair.

$$s_t^D = s_t. \tag{5}$$

In order to match SE with TD, i.e., $y_t = u_t^D$ from (2) and (4), we need to satisfy the following two conditions: one is the distributive property of the IDCT as follows:

$$IDCT(A + B) = IDCT(A) + IDCT(B) \tag{6}$$

and the other is the commutative property of the MC and IDCT as follows:

$$MC\left(IDCT(C)\right) = IDCT\left(MC(C)\right). \tag{7}$$

where A, B, and C are any 8 × 8 DCT coefficients blocks.

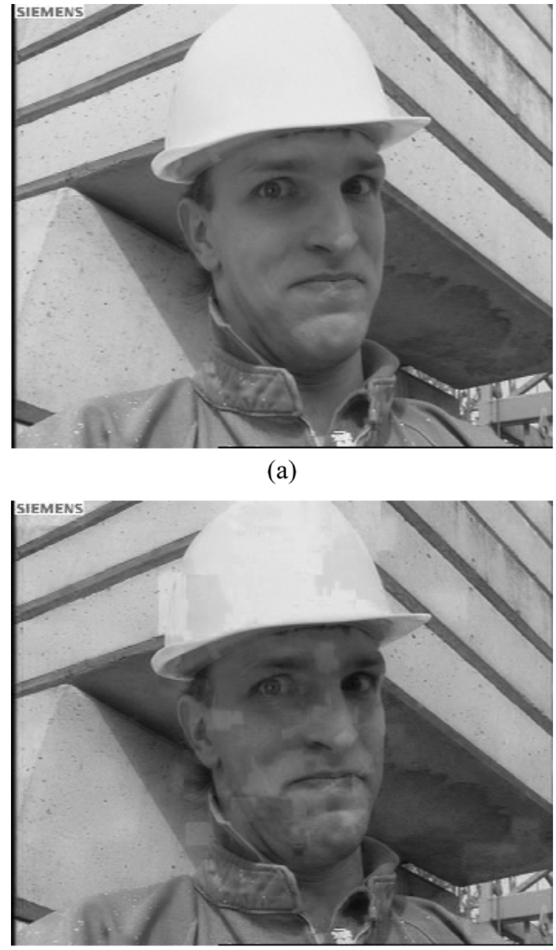
For a TE-SD pair, the reconstructed image is described as follows.

- The reconstructed interframe in TE (Fig. 2)

$$\begin{aligned} y_t^D &= MC^D\left(y_{t-1}^D\right) \\ &\quad + IQ\left(Q\left(DCT(x_t) - MC^D\left(y_{t-1}^D\right)\right)\right) \\ IDCT\left(y_t^D\right) &= IDCT\left(IQ\left(Q\left(DCT(x_t) - MC^D\left(y_{t-1}^D\right)\right)\right) \right. \\ &\quad \left. + MC^D\left(y_{t-1}^D\right)\right). \end{aligned} \tag{8}$$

- The reconstructed interframe in SD (Fig. 3)

$$u^t = IDCT\big(IQ\big(Q\big(DCT(x_t) - MC^D(y_{t-1}^D)\big)\big)\big) + MC(u_{t-1}). \tag{9}$$



(a)

(b)

Fig. 5. (a) The 99th P-frame image of Foreman sequence decoded by SD. (b) The 99th P-frame image of Foreman sequence decoded by TD (Foreman sequence, $N = 100$, $M = 1$).

Similarly, the conditions of (6) and (7) must be satisfied to match the TE-SD pair.

Koc *et al.* established conditions of (6) and (7) to match the SE-TD pair or the TE-SD pair [11], [12]. In Sections III and IV, we investigate the mismatch problem in detail and experiment on each case under various conditions.

III. Analysis of IDCT and MC Mismatches

Fig. 5 shows how the mismatch degrades the decompressed image. The "Foreman" sequence was encoded at 3 Mb/s by using a TM5 encoder [13] in the spatial domain, where there is an I frame for every 100 frames. The compressed data from the TM5 encoder was decoded in the spatial domain [Fig. 5(a)] and the DCT domain [Fig. 5(b)]. Image degradation of Fig. 5(b) arises from the mismatch of the SE-TD pair. Fig. 6 shows peak signal-to-noise ratio (PSNR) curves of the decoded "Foreman" sequence. The PSNR of the SE-TD pair is decreasing continuously along P frames in a sequence of video. This severe degradation occurs because the reference image decoded in the DCT domain is used for MC of the following P-frames so that the MC error propagates.

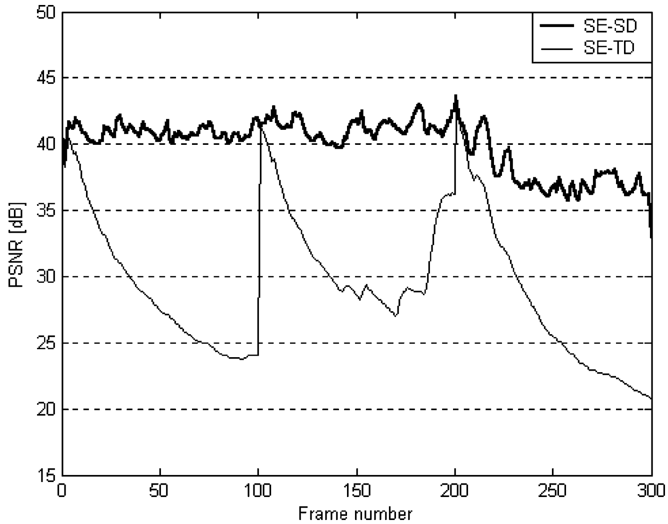


Fig. 6. PSNR curve of Foreman sequence in SD and TD (300 frames, $N = 100$, $M = 1$).

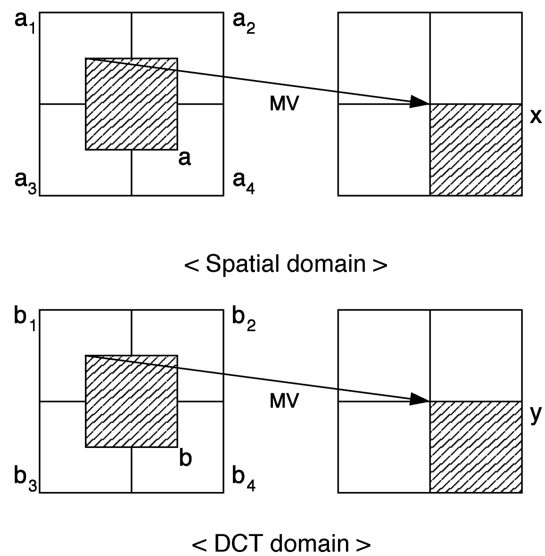


Fig. 7. MC in the spatial domain and the DCT domain.

The first condition of (6) represents the distributive property of the IDCT. An 8 × 8 block $\mathbf{x}$ can be transformed by the DCT as follows:

$$\mathbf{y} = DCT(\mathbf{x}) = \mathbf{A}\mathbf{x}\mathbf{A}^T \tag{10}$$

where $\mathbf{A}$ is the 8 × 8 DCT matrix with entries $a(i,j)$ given by

$$a(i,j) = \frac{1}{2}k(i)\cos\frac{\pi(2j+1)i}{16} \tag{11}$$

where

$$k(i) = \begin{cases} \frac{1}{\sqrt{2}}, & i = 0 \\ 1, & \text{otherwise.} \end{cases}$$

Its IDCT is described as follows:

$$\mathbf{x} = IDCT(\mathbf{y}) = \mathbf{A}^T\mathbf{y}\mathbf{A}. \tag{12}$$

The distributive property of the IDCT can be written as follows:

$$\begin{aligned} IDCT(\mathbf{y_1}+\mathbf{y_2}) &= \mathbf{A}^T(\mathbf{y_1}+\mathbf{y_2})\mathbf{A} \\ &= \mathbf{A}^T\mathbf{y_1}\mathbf{A} + \mathbf{A}^T\mathbf{y_2}\mathbf{A} \\ &= IDCT(\mathbf{y_1}) + IDCT(\mathbf{y_2}). \end{aligned} \tag{13}$$

When the matrix multiplications and additions are performed in floating-point operation, this property will be satisfied without any objection. In conventional encoder and decoder, the short integer (2 bytes) arithmetic operations are usually used, and thus the distributive property of IDCT can not be satisfied as follows:

$$\begin{aligned} \lfloor IDCT(\mathbf{y_1}+\mathbf{y_2})\rfloor &= \lfloor \mathbf{A}^T(\mathbf{y_1}+\mathbf{y_2})\mathbf{A}\rfloor \\ &\neq \lfloor \mathbf{A}^T\mathbf{y_1}\mathbf{A}\rfloor + \lfloor \mathbf{A}^T\mathbf{y_2}\mathbf{A}\rfloor \\ &= \lfloor IDCT(\mathbf{y_1})\rfloor + \lfloor IDCT(\mathbf{y_2})\rfloor \end{aligned} \tag{14}$$

where $\lfloor x \rfloor$ represents the largest integer value that is less than or equal to x. In most DCT-based coders, all the computation results must be rounded as above. This rounding operation makes the distributive property of the IDCT not be satisfied, i.e.,

$$\lfloor 0.4 + 0.6 \rfloor = 1 \neq \lfloor 0.4 \rfloor + \lfloor 0.6 \rfloor. \tag{15}$$

This rounding operation in the recursive motion-compensation loop can induce error propagation. To eliminate the rounding effect, arithmetic processing results should be rounded after the MC, and then the distributive property can be preserved in the recursive loop.

In order to analyze the mismatch of the MC in the spatial domain and the DCT domain, let us assume $\mathbf{a_i}$, $1 \leq i \leq 4$, and $\mathbf{x}$ are 8 × 8 blocks in the spatial domain and $\mathbf{b_i}$, $1 \leq i \leq 4$, and $\mathbf{y}$ are 8 × 8 DCT coefficients of the corresponding blocks as shown in Fig. 7. Then, MC in the spatial domain is described as follows:

$$MC\left(IDCT(\mathbf{b})\right) = MC(\mathbf{a}) = \mathbf{x}. \tag{16}$$

On the other hand, MC in the DCT domain followed by IDCT can be described as follows:

$$IDCT\left(MC(\mathbf{b})\right) = IDCT(\mathbf{y}) = \mathbf{A}^T\mathbf{y}\mathbf{A} = \mathbf{x}. \tag{17}$$

The second condition of (7) is also satisfied when the MC is performed in floating-point. Thus, the MC in the spatial domain (16) is equivalent to that in the DCT domain (17).

In general, the MC in the DCT domain can be performed by matrix multiplications [5] as follows:

$$MC(\mathbf{b}) = \sum_{i=1}^{4} \mathbf{s_{i1}}\mathbf{b_i}\mathbf{s_{i2}} \tag{18}$$

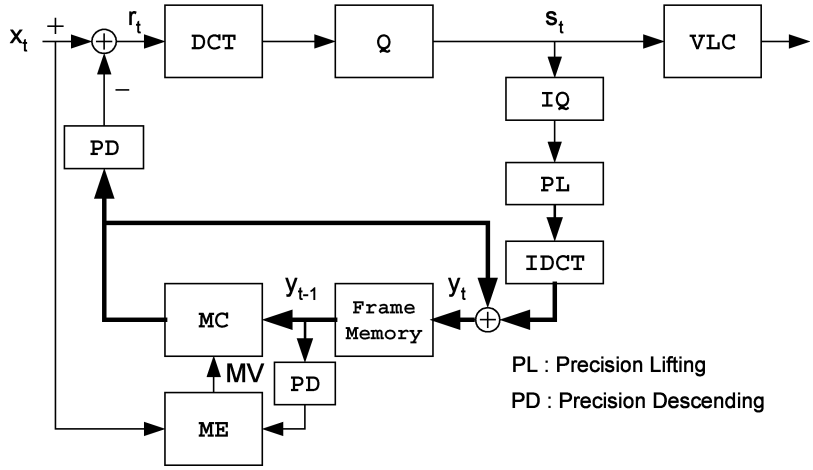


Fig. 8. Spatial-domain encoder with the proposed PL method.

where $\mathbf{s_{i1}}$ and $\mathbf{s_{i2}}$ are the shifting matrices given by $DCT(\mathbf{l_n})$ and $DCT(\mathbf{r_m})$, respectively, and $\mathbf{l_n}$ and $\mathbf{r_m}$ are defined as follows:

$$\mathbf{l_n} = \begin{pmatrix} 0 & 0 \\ I_n & 0 \end{pmatrix} \qquad \mathbf{r_m} = \begin{pmatrix} 0 & I_m \\ 0 & 0 \end{pmatrix}. \tag{19}$$

In (19), I_n is an identity matrix whose size is $n \times n$, and n and $m(1 \leq n, m \leq 7)$ are determined according to motion vector (MV), i.e., MV *modulo* 8 is (n, m).

The matrix multiplications of (18) should be performed in floating-point to guarantee that the MC in the DCT domain is equivalent to that in the spatial domain. However, most encoders and decoders use fixed-point or integer operations to save computation time, where the MC in the DCT domain is not equivalent to that in the spatial domain.

In the SE-TD pair, the mismatch problem of the IDCT and MC can be overcome by floating-point operations. In Figs. 1 and 4, the bold line denotes the floating-point data paths and the dashed line denotes the rounding operation of floating-point to integer. The two conditions of (6) and (7) are satisfied and there is no mismatch in the SE-TD pair. Also, the TE-SD pair can satisfy two conditions in the same way as the SE-TD pair.

In general, the floating-point operation requires a long-word buffer with floating-point format (4 bytes) and heavy computation load. If the fast processing is required, these floating-point arithmetic operations in the recursive motion-compensation loop are not appropriate. Therefore, we propose the PL method to reduce the mismatch in the SE-TD or TE-SD pair as well as to save computation time.

The proposed PL method can reduce the rounding effect for the distributive property of IDCT as follows:

$$\frac{\lfloor IDCT(\alpha \times \mathbf{y_1} + \alpha \times \mathbf{y_2}) \rfloor}{\alpha} = \frac{\lfloor \mathbf{A}^T(\alpha \times \mathbf{y_1} + \alpha \times \mathbf{y_2})\mathbf{A} \rfloor}{\alpha} \tag{20}$$

$$\frac{(\lfloor IDCT(\alpha \times \mathbf{y_1}) \rfloor + \lfloor IDCT(\alpha \times \mathbf{y_2}) \rfloor)}{\alpha} = \frac{(\lfloor \mathbf{A}^T(\alpha \times \mathbf{y_1})\mathbf{A} \rfloor + \lfloor \mathbf{A}^T(\alpha \times \mathbf{y_2})\mathbf{A} \rfloor)}{\alpha} \tag{21}$$

where α is a lifting constant for lifting the precisions. This PL method can reduce the difference of (14), for example, when $\alpha = 10$

$$\frac{\lfloor 10 \times 0.4 + 10 \times 0.6 \rfloor}{10} = \frac{\lfloor 10 \rfloor}{10} = 1 \tag{22}$$

$$\frac{(\lfloor 10 \times 0.4 \rfloor + \lfloor 10 \times 0.6 \rfloor)}{10} = \frac{(4+6)}{10} = 1. \tag{23}$$

The rounding of the decimal part from floating-point to integer causes an error in the distributive property. The distributive property can be satisfied by the PL with 10 as in (22) and (23). However, the PL method cannot remove all the rounding effect for more complicated operations. As α becomes large, the rounding error decreases. Although there are slight rounding errors, our proposed method can reduce the mismatch without severe computational overhead.

The proposed PL method is also used to reduce the mismatch in the recursive motion-compensation loop. The data type in the recursive MC loop is a short integer except the MC in the DCT domain (MC-DCT). Though the input data type of the MC-DCT is a short integer, internal computation is performed in long integer type in order to overcome the overflow during the matrix multiplications. MC-DCT in long integer arithmetic operations requires shifting matrices with long integer type elements. Most elements of the shifting matrix are fractional numbers less than 1. To use long integer arithmetic operations in the MC-DCT, (18) should be modified as follows:

$$MC(\mathbf{x}) = \sum_{i=1}^{4} \frac{\left\{ \left\{ \frac{(\lfloor \alpha_m \times \mathbf{s_{i1}} \rfloor \times \mathbf{x_i})}{\alpha_m} \right\} \times \lfloor \alpha_m \times \mathbf{s_{i2}} \rfloor \right\}}{\alpha_m}. \tag{24}$$

The lifting constant for MC, α_m, must guarantee the precision of the MC-DCT.

Figs. 8 and 9 show block diagrams of SE and TD block diagrams, respectively, with the proposed PL method. The bold lines are the data paths with PL and the precision descending (PD) lowers the lifted precision. In this structure, the IDCT module is performed in floating-point. This can increase computation time in the encoder and decoder. In order to reduce the computation time, many practical DCT/IDCT modules use

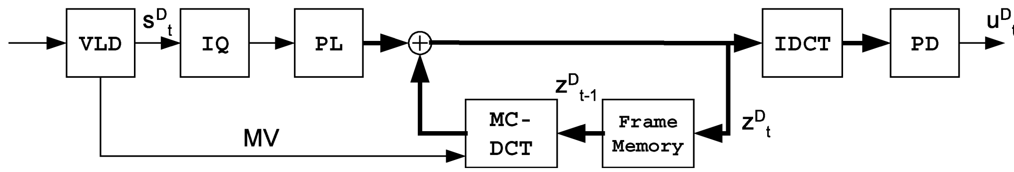


Fig. 9. DCT-domain decoder with the proposed PL method.

integer coefficients instead of fractional ones. In order to reduce the rounding effect of the integer DCT/IDCT, the DCT and IDCT can be performed with the PL method as follows:

$$\mathbf{y} = DCT(\mathbf{x}) = \frac{\left\{\left\{\frac{(\lfloor\alpha_d \times \mathbf{A}\rfloor \times \mathbf{x})}{\alpha_d}\right\} \times \lfloor \alpha_d \times \mathbf{A}^T \rfloor\right\}}{\alpha_d} \quad (25)$$

$$\mathbf{x} = IDCT(\mathbf{y}) = \frac{\left\{\left\{\frac{(\lfloor\alpha_d \times \mathbf{A}^T\rfloor \times \mathbf{y})}{\alpha_d}\right\} \times \lfloor \alpha_d \times \mathbf{A} \rfloor\right\}}{\alpha_d} \quad (26)$$

where α_d is the lifting constant for integer DCT/IDCT.

IV. Experimental Results and Discussions

This section describes experimental results of a SE-TD pair using the modified MPEG-2 TM5 encoder/decoder. A "Foreman" test sequence (CIF format of 352 $\times$ 288, 300 frames) was encoded at 3 Mb/s with the TM5 encoder, where modification of the TM5 is performed only in DCT, IDCT, and MC as shown in Figs. 8 and 9. Group-of-picture (GOP) size is set to 100 frames and there are no B-pictures ($N = 100$, $M = 1$), i.e., there is one I-picture for every 100 frames and the other 99 frames in a GOP are all P-pictures. Experiments were conducted on a Pentium III computer running at 1 GHz with 512 Mbytes of memory.

As shown in Fig. 5(a) and (b), image quality in the decoded image from the TD is highly degraded in a conventional SE-TD pair due to the mismatch problem. Two conditions of (6) and (7) must be satisfied to remove the mismatch problem. Fig. 10 shows PSNR curves when the mismatch problem is removed by floating-point operations. The PSNRs of the SE-SD and SE-TD pairs are exactly the same in Fig. 10. To reduce the mismatch problem as well as to save computational cost and buffer size, we proposed a SE-TD pair using the PL method and integer operations. In Fig. 11, "fSE" is a spatial-domain encoder using floating-point operations for MC and DCT/IDCT, and "rSE" is a spatial-domain encoder using integer operations of MC with PL method and rounding of the floating-point DCT/IDCT. Similarly, "fTD" is a transform-domain decoder using floating-point operations for MC and IDCT and "rTD" is a transform-domain decoder using integer operation of MC with the PL method and rounding of the floating-point IDCT, respectively. The experimental results show the performance of the rSE-rSD pair is almost the same as that of the fSE-fSD pair and better than that of the rSE-rTD pair. We can see the mismatch in the rSE-rTD pair is greatly improved in comparison with the SE-TD pair in Fig. 6. Since the floating-point operations are still used for DCT/IDCT in the rSE-rTD pair, it is not adequate for fast encoder and decoder. The term "iSE" in Fig. 12 means that all the operations of DCT/IDCT and MC are performed by integer processing with

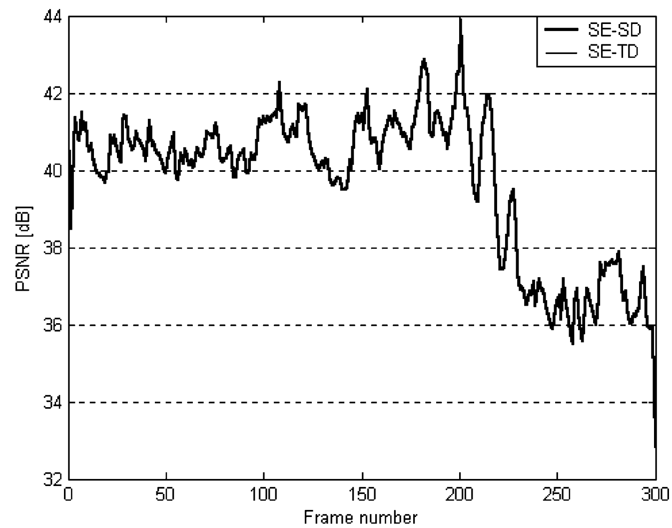


Fig. 10. PSNR of SE-SD and SE-TD with floating-point processing.

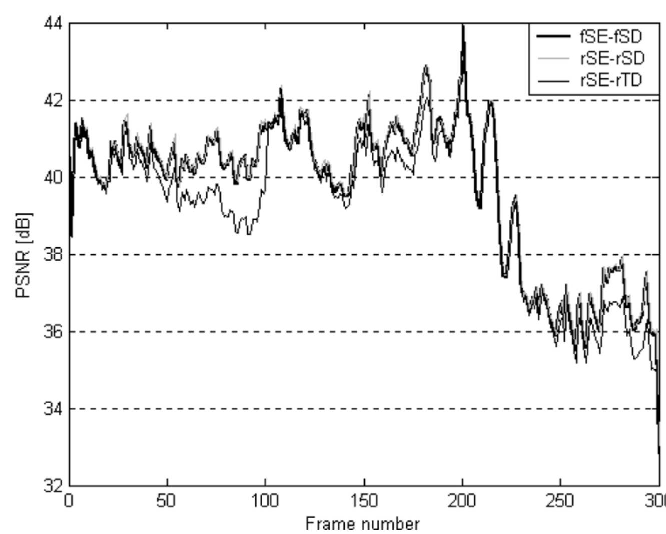


Fig. 11. PSNR of SE-SD and SE-TD with rounding of floating-point DCT/IDCT and integer MC. ($\alpha = 10, \alpha_m = 200\,000$).

the PL method, where the corresponding lifting constants are $\alpha = 10$, $\alpha_m = 200\,000$, and $\alpha_d = 2048$. As shown in Fig. 12, PSNR of the iSE-iTD pair is much higher than that of the SE-TD pair in Fig. 6, and it is lower than that of the fSE-fSD pair. Table I lists the computation time to decode the "Foreman" sequence with the fTD and the iTD, respectively. The iTD achieves about 43% computational savings in comparison with the fTD. In the iTD, computation overhead for PL is about 0.1 s.

Fig. 13 shows how the distributive property of the IDCT and the commutative property of the MC and IDCT are affected by the proposed PL method. In terms of "xySD" or "xyTD," if

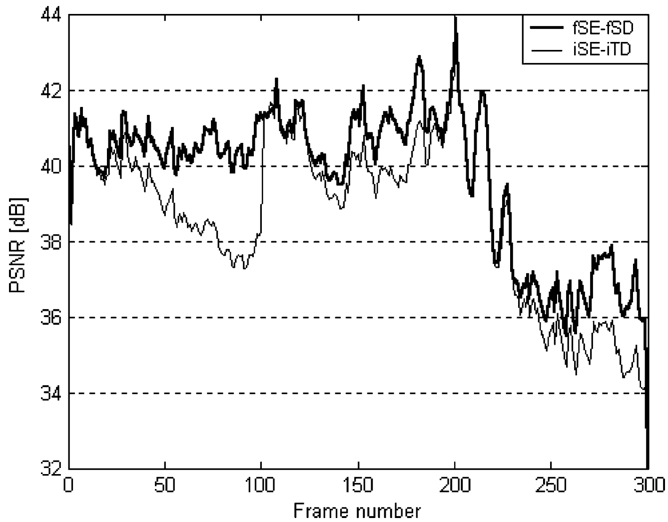


Fig. 12. PSNR of SE-TD from integer processing with the proposed PL method ($\alpha = 10, \alpha_m = 200\,000, \alpha_d = 2048$).

TABLE I
COMPUTATION TIMES [s] FOR DECODING FOREMAN SEQUENCE WITH FLOATING-POINT OPERATIONS AND INTEGER OPERATIONS IN TD (CIF FORMAT OF 352 × 288, 300 FRAMES, $N = 100, M = 1$)

Decoding Method	VLD + IQ	MC	IDCT	Total
Floating-point operations	2.03	8.24	20.26	30.53
Integer operations with PL	2.03	5.82	9.73	17.58

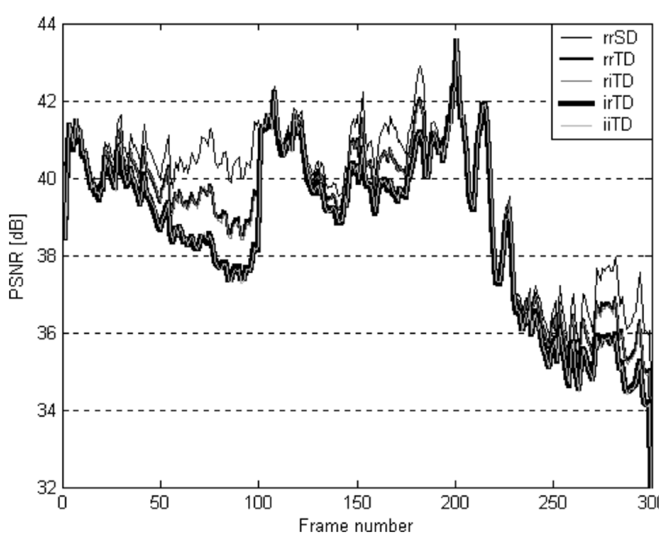


Fig. 13. PSNR of SD and TD with different motion-compensation and IDCT ($\alpha = 10, \alpha_m = 200\,000, \alpha_d = 2048$).

the first letter "x" is "r," the MC is performed by rounding of floating-point operations, whereas the first letter "i" means the MC is performed by integer processing with the PL method. If the second letter "y" is "r," the IDCT is performed by rounding of floating-point operations, whereas the second letter "i" means the IDCT is performed by integer processing with the

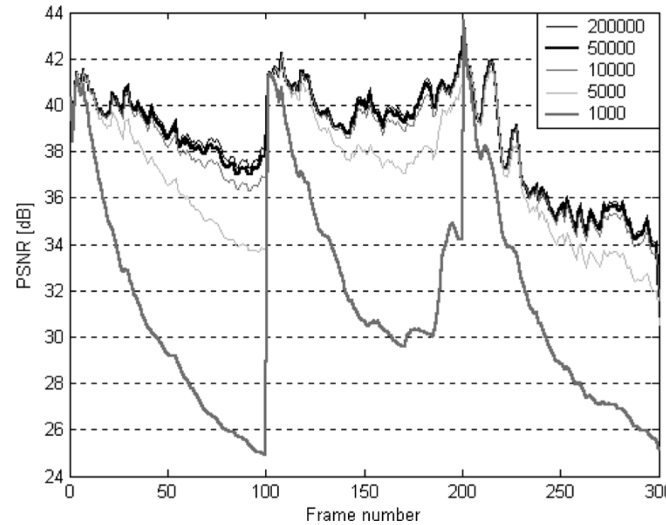


Fig. 14. PSNR of TD with respect to various lifting constants (α_m) ($\alpha = 10$, $\alpha_d = 2048$).

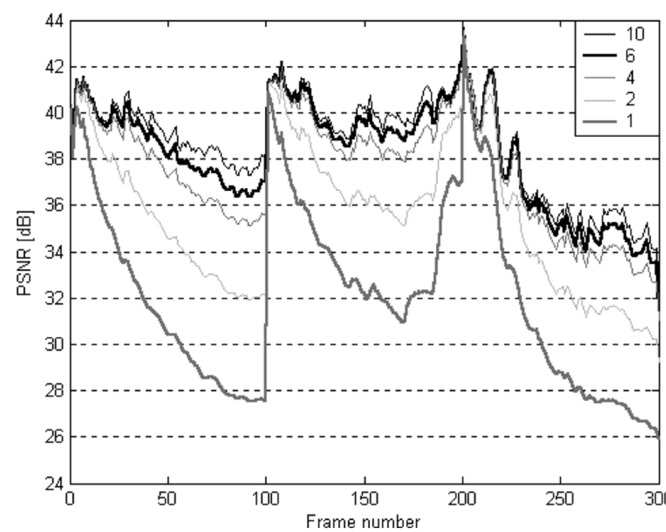


Fig. 15. PSNR of TD with respect to various lifting constants (α) ($\alpha_m = 200\,000, \alpha_d = 2048$).

PL method (26). As shown in Fig. 13, there is a little difference of PSNR between the two IDCT methods of xrTD and xiTD. The major difference between rrSD and rrTD is that rrTD does not perfectly satisfy the distributive property of the IDCT. If the PL method is not used, the PSNR difference would gradually increase as SE-TD pair in Fig. 6. The PSNR difference between rrTD and irTD depends on whether the commutative property of the MC and IDCT is satisfied. In rrTD, the MC is performed by floating-point processing, and thus the commutative property of MC and IDCT can be satisfied. From these results, we find that the MC using integer processing generates mismatch error in long P-frames.

Fig. 14 shows experimental results for various lifting constants of the MC in the DCT domain as shown in (24). In the MC in the DCT domain, the lifting constant α_m over 10000 can make the rounding error negligible. Fig. 15 presents experimental results for various lifting constants to satisfy the distributive property of IDCT. If we use a lifting constant α over

(a)

(b)

Fig. 16. (a) Error image between the reference image and the reconstructed image decoded by TD with the proposed PL method (the 99th P-frame). (b) Error image between the reference image and the reconstructed image decoded by TD without the PL method (the 99th P-frame).

TABLE II
AVERAGE PSNR [decibels] OF FOUR VIDEO SEQUENCES DECODED BY THREE DIFFERENT DECODING METHODS

Decoding Method	Foreman	Mobile and Calendar	Paris	Tempete
SD	39.82	30.95	40.10	34.54
TD without PL	32.20	28.13	36.19	31.88
TD with PL	38.71	30.68	38.35	34.21

6, the mismatch can be reduced effectively. In Fig. 16, the effectiveness of the proposed PL method is shown. As shown in Fig. 16(a) and (b), the error from the mismatch can be reduced by using the proposed PL method.

We also experimented with several test sequences (CIF format of 352 × 288, 300 frames, $N = 100$, $M = 1$) of "Foreman," "Mobile and calendar," "Paris," and "Tempete". They were encoded at 3 Mb/s in the spatial domain by MPEG-2 TM5 encoder. Table II shows the average PSNR when each sequence was decoded by three different decoding methods: spatial-domain decoding, DCT-domain decoding in integer operations without the PL, and DCT-domain decoding in integer processing with the PL. The PL was performed with α of 10, α_m of 200000, and α_d of 2048. The experimental results show that our proposed PL method can effectively reduce the mismatch problem in the SE-TD pair.

V. CONCLUSION

We analyzed the mismatch problem in the SE-TD and TE-SD pairs in detail and proposed a PL method to reduce the effects of the mismatch problem. Since the distributive property of the IDCT and the commutative property of the MC and IDCT are not satisfied in the integer processing, mismatch can occur and a decoded image can be degraded gradually. We theoretically and experimentally investigated the effects of mismatch in the conventional video coders. In a DCT domain decoder using integer arithmetic operations, the proposed PL method can prevent severe degradation in decoded image quality. When a decoder that is performed in the different domain from the encoder is used, the proposed method can reduce the mismatch problem without severe computation overhead.

REFERENCES

[1] *Information Technology-Coding of Moving Pictures and Associated Audio for Digital Storage Media at Up to About 1.5 Mbit/s—Part 2: Video*, ISO/IEC 11172-2, 1993.

[2] *Information Technology-Generic Coding of Moving Pictures and Associated Audio Information—Part 2: Video*, ISO/IEC 13818-2, 1995.

[3] *Video Codec for Audio Visual Services at* $p \times 64$ Kbits/s, ITU-T H.261, 1990.

[4] *Video Coding for Low Bit Rate Communication*, ITU-T H.263, Jan. 1998.

[5] S. F. Chang and D. G. Messerschmitt, "Manipulation and composition of MC-DCT compressed video," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 3, pp. 1–11, Jun. 1995.

[6] N. Merihav and V. Bhaskaran, "Fast algorithms for DCT-domain image down sampling and for inverse motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 3, pp. 468–476, Jun. 1997.

[7] P. Assuncao and M. Ghanbari, "A frequency-domain video transcoder for dynamic bit-rate reduction of MPEG-2 bit streams," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 8, pp. 953–967, Dec. 1998.

[8] J. Song and B. L. Yeo, "Fast extraction of spatially reduced image sequences from MPEG-2 compressed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 7, pp. 1100–1114, Oct. 1999.

[9] U. V. Koc and K. J. R. Liu, "DCT-based motion estimation," *IEEE Trans. Image Process.*, vol. 7, no. 7, pp. 948–965, Jul. 1998.

[10] H. W. Park, Y. S. Park, and S. K. Oh, "L/M-fold image resizing in block-DCT domain using symmetric convolution," *IEEE Trans. Image Process.*, vol. 12, no. 9, pp. 1016–1034, Sep. 2003.

[11] U. V. Koc and K. J. R. Liu, "Motion compensation on DCT domain," *EURASIP J. Appl. Signal Process.*, vol. 3, pp. 147–162, 2001.

[12] J. Chen, U. V. Koc, and K. J. R. Liu, *Design of Digital Video Coding Systems*. New York: Marcel Dekker, 2002.

[13] *Test Model 5*, ISO/IEC JTC1/SC29/WG11/N0400, MPEG93/457, Apr. 1993.

[14] H. S. Malvar, A. H. Hallapuro, M. Karczewicz, and L. Kerofsky, "Low-complexity transform and quantization in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

Seung-Kyun Oh was born in Seoul, Korea, in 1975. He received the B.S. and M.S. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1998 and 2000, where he is currently pursuing the Ph.D. degree.

His research interests include video transcoding, image/video processing in the compressed domain, and implementation of video codecs.

HyunWook Park (SM'99) received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1981, and the M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejon, in 1983 and 1988, respectively.

He has been a Professor of electrical engineering since 1993 and has been an Adjunct Professor of Biosystems at KAIST since 2003. He was a Research Associate at the University of Washington, Seattle, from 1989 to 1992 and was a Senior Executive Researcher at the Samsung Electronics Co., Ltd., Suwon, Korea, from 1992 to 1993. He has served as Associate Editor for the *International Journal of Imaging Systems and Technology*. His current research interests include image computing system, image compression, medical imaging, and multimedia systems.

Dr. Park is a member of SPIE.