MPEG-2/H.264 transcoding with vector conversion reducing re-quantization noise

Takeshi Yoshitome, Kazuto Kamikura
NTT Cyber Space Laboratories
NTT Corporation
Yokosuka, Japan, 239–0847
Email: yoshitome.takeshi@lab.ntt.co.jp

Shoji Makino, Nobuhiko Kitawaki
Graduate School of Systems and Information Engineering
University of Tsukuba
Tsukuba, Japan, 305–8573

***Abstract*—We propose an MPEG-2 to H.264 transcoding method for interlace streams intermingled with frame and field macroblocks. This method uses the encoding information from an MPEG-2 stream and keeps as many DCT coefficients of the original MPEG-2 bitstream as possible. Experimental results show that the proposed method improves PSNR by about 0.19-0.31 dB compared with a conventional method.**

I. INTRODUCTION

Recently, it has become important to re-compress MPEG-2 video content to decrease its bitstream size by using a high performance encoding tool such as H.264. In this decade, a number of video transcoding studies have been reported [5]–[7] that aim to reduce the H.264 encoding time, which is several times larger than that of MPEG-2. However, the image quality obtained with these approaches becomes lower than that of conventional approaches using a directly connected MPEG-2 decoder and H.264 encoder.

We have focused on the fact that all decisions of the first encoding procedure are repeated during the transcoding procedure to reduce re-quantization noise [1], where "all decisions" means decisions on picture type, macroblock type, motion vectors, DCT type, and so on. This noise suppression mechanism is effective when the compression standards of the first and second encoding are identical, such as for MPEG-2/MPEG-2 or H.264/H.264 conversion. We have tried to expand this approach to apply it to MPEG-2/H.264 transcoding for intra bit-streams [2], [4]. This paper reports our attempt to apply this noise suppression mechanism to inter bit-streams consisting of frame and field macroblocks. First, we discuss the noise suppression mechanism for progressive bit-streams. Next, we show the differences between the MPEG-2 and H.264 standards for frame and field macroblocks, analyze how they affect the noise reduction, and propose an MPEG-2/H.264 transcoding method. Finally, we describe simulation results we obtained and compare the transcoding noises generated by our method and a conventional method.

II. NOISE REDUCTION MECHANISM USING FIRST ENCODING INFORMATION

Data flow examples of transcoding with and without first MPEG-2 encoding information are shown in Fig. 1. The data flow of Enc1→Dec1→Enc2→Dec2 is with encoding information, and that of Enc1→Dec1→Enc3→Dec3 is without it. For simplicity, the macro-block (MB) size is 2x2. In the first MPEG-2 encoding, an input MB that includes the pixel values (A, B, C, and D) is processed by motion compensation and orthogonal transformation. Then, the transformed DCT coefficients (47, 31, 25, and 18) are quantized by a quantization step Δ=10. The quantized DCT coefficients (4, 3, 2, and 1) are converted to bitstream#1 by variable length coding (VLC). The quantization from (47, 31, 25, and 18) to (4, 3, 2, and 1) adds quantization noise so that the decoded MB pixels (a, b, c, and d) differ from the input MB pixels (A, B, C, and D). Until this point, transcoding both with and without encoding information does the same work.

In the transcoding without encoding information, Enc3 searches the best motion vector to minimize motion estimation error (p', q', r', and s'). If the motion search performance of Enc3 is higher than that of Enc1, Enc3 can reduce its motion estimation error (p', q', r', and s') compared with Enc1's motion estimation error (A', B', C', and D'). In the same way, if the orthogonal transformation performance of Enc3 is higher than that of Enc1, Enc3 can reduce the size of bit-

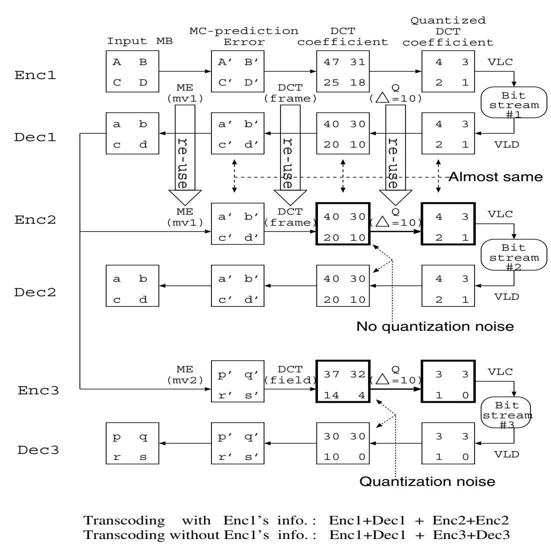

Fig. 1. Transcoding with and without first MPEG-2 encoding information

stream#3 compared with that of bitstream#1. However, the re-quantization noise occurs when DCT coefficients (37, 32, 14, and 4) are quantized to (3, 3, 1, and 0). In short, Enc3 adds re-quantization noise even if the bitstream becomes smaller. On the other hand, in the transcoding with encoding information, every type of encoding information, such as MB type, motion vector, DCT mode, and quantization step, is re-used in Enc2. This re-use of encoding information does not produce any significant difference between motion compensation errors of the first encoding and those of the second encoding, nor does it produce any notable difference between the DCT coefficients of the two encodings. This means that the re-quantization noise added in the second encoding phase is almost zero. As a result, the same motion estimation error and the same DCT quantization coefficients are obtained. Unfortunately, there is no bit reduction because the sizes of bitstream#1 and #2 are almost the same when these encodings make use of the same compression tool. But transcoding using different compression tools, such as MPEG-2 to H.264 transcoding, may reduce the bitstream size because the performance of H.264's entropy coding (CABAC) is higher than that of MPEG-2's entropy coding (VLC).

III. Differences between MPEG-2 and H.264

As depicted in Fig. 1, there is a close resemblance between the decoded images of the first and second decoders when the compression standards of the first and second encodings are exactly the same. To re-enact the effect of this noise reduction mechanism, we must control H.264 behavior to imitate MPEG-2 behavior as much as possible when the first encoder is MPEG-2 and the second encoder is H.264. The two encoders are similar but are not upper compatible. The differences between them are listed in Table I. The "ok" means that H.264 can recreate the same MPEG-2 function in the table. The "?" means that H.264 can process the function only in a similar but not identical way that MPEG-2 does. The "checked" means that these differences were analyzed in our previous work [2], [3]. The following two functions are not exactly the same between MPEG-2 and H.264, and are not investigated: (1) Specification unit of DCT type representing frame MB or field MB, (2) Specification unit of motion compensation type representing frame MB or field MB. In the next section, we describe these differences in detail and show a way to overcome them.

TABLE I
Major differences between MPEG-2 and H.264

	MPEG-2	H.264	H.264 re-enact
Block size	16x16	4x4, 4x8, 8x4, 8x8 8x16,16x8 16x16	OK
Vector resolution	0.5pix	0.25pix	OK
Orthogonal transformation	DCT	integer DCT	? checked
Size of orthogonal transformation	8x8	8x8(*1) 4x4	OK
Specification of quantization matrix	yes	yes(*1)	OK
Deadzone in intra quantization	no	no	OK
Deadzone in inter quantization	yes	no	? checked
MC specification unit of field/frameMB	each MB	pair MBs	?
DCT specification unit of field/frameMB	each MB	pair MBs	?

(*1): only High-Profile

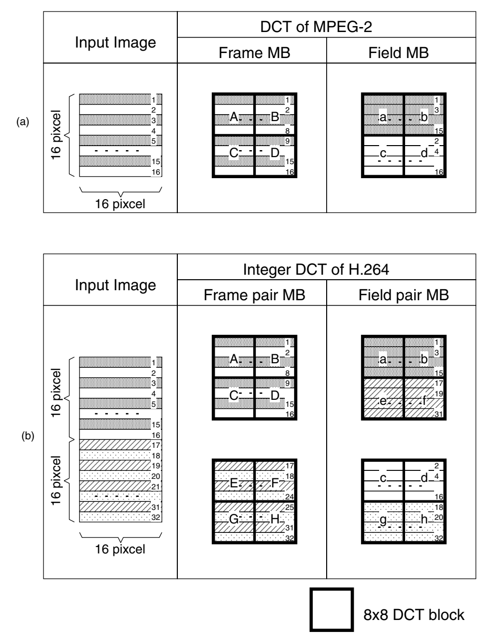

Fig. 2. DCT type specification in MPEG-2 and H.264

IV. Frame/Field macroblock specification

A. DCT type specification in MPEG-2 and H.264

The DCT type specification of MB in MPEG-2 and H.264 is shown in Fig. 2. Although a DCT type can be specified for each MB independently in MPEG-2, H.264 specifies the DCT type for each pair of MBs that adjoin each other above and below. When an upper MB is specified as the frame DCT and its lower MB is specified as the field DCT in MPEG-2, the DCT type of one MB is different from its DCT type specified in H.264, whether the DCT type of the pair of MBs is frame or field type in H.264.

In our previous trial [4], we compared the two methods that select frame MB pairs or field MB pairs when DCT types of upper and lower MB were different in MPEG-2. The simulation results obtained show that the selection of frame MB pairs is superior to that of field MB pairs in PSNR. In this paper, we use this MB pair selection method again for intra MBs, as shown in Table II.

TABLE II
RELATIONSHIP BETWEEN H.264 MB PAIR TYPE AND MPEG-2 DCT TYPE SUCCESSION

Input MPEG-2 DCT type		Output H.264				
		Frame pair MB selection		Field pair MB selection		Our selection type
upper MB	lower MB	upper MB	lower MB	upper MB	lower MB	
Frame	Frame	O	O	X	X	Frame
	Field	O	X	X	O	Frame
Field	Frame	X	O	O	X	Frame
	Field	X	X	O	O	Field

O: Succession, X: No succession

B. Motion compensation type specification in MPEG-2 and H.264

The relationship between the pixel group and the motion vector in MPEG-2's MB is shown in Fig. 3. A single frame motion vector MV_{MPEG} represents the motion of the MB including 16x16 pixels when its motion type is specified as frame. When its motion type is specified as field, a top field motion vector mvt_{MPEG} represents the motion of 16x8 pixels included in the top field MB; a bottom field motion vector mvb_{MPEG} represents the motion of 16x8 pixels included in the bottom field MB. H.264 specifies the motion type for each pair of MBs that adjoin each other above and below. When the MB pair is specified as a frame, a frame vector $MV0_{H264}$ represents the motion of the above MB, and another frame vector $MV1_{H264}$ represents the motion of the below MB. When the MB pair is specified as a field, the four motion vectors $mvt0_{H264}$, $mvt1_{H264}$, $mvb0_{H264}$, and $mvb1_{H264}$ are used. The vectors $mvt0_{H264}$ and $mvb0_{H264}$ represent the motion of top and bottom field pixels in the upper MB. The vectors $mvt1_{H264}$ and $mvb1_{H264}$ represent the motion of top and bottom field pixels in the lower MB. This is similar to a DCT type specification in which H.264 specifies the motion type for each pair of MBs that adjoin each other above and below. The MPEG-2 motion type succeeds to H.264 only when both the upper and lower MBs have the same motion type in MPEG-2.

C. Conversion method from frame vector to field vector

We introduce a method of converting from frame vector to field vector to increase the number of H.264 MBs that succeed to the MPEG-2 motion type. Figure 4shows examples of conversion from frame vectors to field vectors. Figure 4(a) shows the conversion when the vertical vector value equals -2 pixels and Fig. 4(b) shows it when the value equals -1 pixels. In Fig. 4(a), the top field vector points to the top field pixels in the reference picture, and the bottom field vector points to the bottom field pixels. In contrast, in Fig. 4(b), the top field vector points to the bottom field pixels, and the bottom field vector points to the top field pixels. The top field pixels are located one pixel position higher than that of bottom field pixels. Therefore, a frame vector that indicates -1 pixel movement is replaced by a top field vector that indicates -2 movement and a bottom field vector that indicates zero movement. The rules

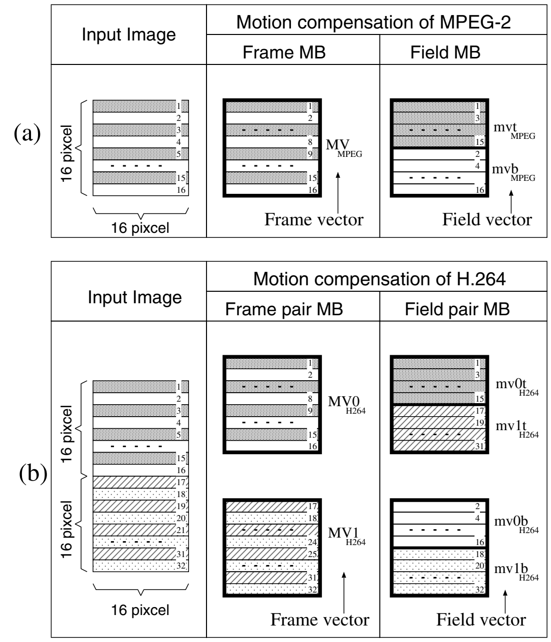

Fig. 3. Motion compensation type specification in MPEG-2 and H.264

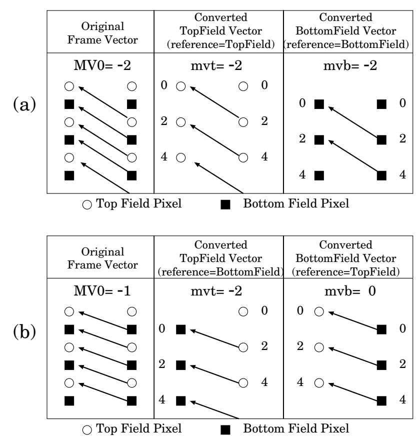

Fig. 4. Conversion from frame vector to field vector
(a)Y vector= –2 pix, (b)Y vector = –1 pix

for converting a frame vector to two field vectors are shown in the following equations.

$$
\begin{aligned}
mvt0x &= MV0x \\
mvt0y &= MV0y + \alpha
\end{aligned}
$$

$$
where\ \ \alpha = \begin{cases} -1\,(MV0y = \pm 1, \pm 3, \pm 5, ..) \\ 0 \quad (MV0y = otherwise) \end{cases} \tag{1}
$$

$$
\begin{aligned}
mvb0x &= MV0x \\
mvb0y &= MV0y + \beta
\end{aligned}
$$

$$
where\ \ \beta = \begin{cases} +1\,(MV0y = \pm 1, \pm 3, \pm 5, ..) \\ 0 \quad (MV0y = otherwise) \end{cases} \tag{2}
$$

$$t0ref = \begin{cases} bottom & (MV0y = \pm 1, \pm 3, \pm 5, ..) \\ top & (MV0y = otherwise) \end{cases} \quad (3)$$

$$b0ref = \begin{cases} top & (MV0y = \pm 1, \pm 3, \pm 5, ..) \\ bottom & (MV0y = otherwise) \end{cases} \quad (4)$$

In these equations, the MPEG-2 frame vector is $MV0$ and its X and Y direction values are $MV0x$ and $MV0y$. The H.264 top and bottom field vectors for the upper MB are $mvt0$ and $mvb0$. Their X and Y direction values are $mvt0x$, $mvt0y$, $mvb0x$ and $mvb0y$. The reference fields of $mvt0$ and $mvb0$ are $t0ref$ and $b0ref$. These conversions are only approximations, but are sufficiently accurate that the noise reduction effect mentioned in section II remains when using them. The relationship between noise reduction effect and the vector conversion will be described in Section V-C. Table III shows the succession of DCT and motion compensation from MPEG-2 to H.264 using frame to field vector conversion. There are 16 cases of the combination of DCT type and motion compensation type for a MB pair in MPEG-2. The right side of the table shows whether the H.264 MB succeeds or does not succeed to the DCT type and the motion compensation type in frame pair selection or in field pair selection. An "o" means that H.264 MB succeeds without vector conversion. A "#" means that H.264 MB succeeds with vector conversion. An "x" means that H.264 MB does not succeed. We measured the DCT type and motion compensation type of MPEG-2 bitsteams generated by five commercial hardware encoders, as shown in Table IV. The succeeding ratio is 57.2% without vector conversion and increases to 72.1% using vector conversion.

TABLE III
THE RELATIONSHIP BETWEEN THE SUCCESSION OF DCT AND MOTION COMPENSATION FROM MPEG-2 TO H.264 AND THE PAIR TYPE SELECTION IN H.264.

Group No.	Input MPEG-2				Output H.264			
	DCT type		Motion compensation type		Frame pair MB selection		Field pair MB selection	
	up MB	low MB	up MB	low MB	up MB	low MB	up MB	low MB
0	Fr	Fr	Fr	Fr	o	o	x	x
1				Fi	o	x	x	x
2			Fi	Fr	x	o	x	x
3				Fi	x	x	x	x
4		Fi	Fr	Fr	o	x	x	#
5				Fi	o	x	x	o
6			Fi	Fr	x	x	x	#
7				Fi	x	x	x	o
8	Fi	Fr	Fr	Fr	x	o	#	x
9				Fi	x	x	#	x
10			Fi	Fr	x	o	o	x
11				Fi	x	x	o	x
12		Fi	Fr	Fr	x	x	#	#
13				Fi	x	x	#	o
14			Fi	Fr	x	x	o	#
15				Fi	x	x	o	o
Number of succeeded MB					4	4	8(4)	8(4)

Fr: Frame, Fi: Field
o: Succeeded MB without vector conversion
#: Succeeded MB with vector conversion
x: Non-succeeded MB
( ): Number of succeeded MBs without vector conversion

V. Proposed Method and Evaluation

A. Proposed Method

The inter-MB conversion rule of the proposed method for P-pictures is shown in Table.V. Our method uses the intra-MB conversion rule that has already been reported in [4]. For B-pictures, our method uses a conventional method that does not use first encoding information.

The method selects pairs of MB type, MB type, motion vector, and $Transform\_size\_8x8\_flag$ according to the information of the DCT type and the motion compensation type in MPEG-2. The MB pair type selection of H.264 is set up to maximize the number of MBs that succeed to the DCT type and motion compensation type from MPEG-2 to H.264. The method does not select whether to use vector conversion when the number of MBs succeeding MPEG-2's type does not change whether or not vector conversion is used. This is because the conversion from a single frame vector to two field vectors increases the bitstream size. For example, when the group number is four in TableIII, a single MB succeeds the MPEG-2's type whether or not vector conversion is used. TableV shows that our method selects a frame pair that does not require the conversion in such cases.

For succeeding MBs, the MB type is P_L0_16x16 in frame pair selection, or P_L0_L0_16x8 in field pair selection. The motion vectors are succeeded directly, or converted by the vector conversion mentioned in SectionIV-C. There is no need to search a motion vector that requires a large CPU time. The $Transform\_size\_8x8\_flag$ is set to one to equalize the DCT size in H.264 and MPEG-2. The quantization step is selected to be the value nearest to that of MPEG-2's quantization step. For non-succeeding MBs, the MB type, $Transform\_size\_8x8\_flag$, and motion vectors are re-judged or re-searched using the same way that the conventional method does.

The following common settings are declared in the H.264's picture header or slice header. The H.264 profile is a high profile one that makes it possible to use the same size of MPEG-2 DCT. The coefficient of the H.264 quantization matrix is set to the same as that of the MPEG-2's matrix. Because the H.264 encoder can refer to the same picture

TABLE IV
THE POSSIBLE SUCCEEDING RATIO OF BOTH DCT TYPE AND MOTION COMPENSATION TYPE FROM MPEG-2 TO H.264 MEASURED IN FIVE COMMERCIAL MPEG-2 ENCODER(%)

	Without vector conv.	With vector conv.	Difference
Encoder A	53.9	78.4	22.4
Encoder B	45.3	77.8	32.5
Encoder C	59.4	65.6	6.2
Encoder D	64.8	70.5	5.7
Encoder E	62.6	68.0	5.4
Ave.	57.2	72.1	14.9

TABLE V
INTER MB CONVERSION RULE FROM MPEG-2 TO H.264 IN P-PICTURE

Group no.	Input MPEG-2				Output H.264						
	DCT type		Motion comp. type		Pair MB type	MB type		Vector		$Transform\_size\_8x8\_flag$	
	up MB	low MB	up MB	low MB		up MB	low MB	up MB	low MB	up MB	low MB
0	Fr	Fr	Fr	Fr	Fr	P_L0_16x16	P_L0_16x16	succession	succession	1	1
1				Fi	Fr	P_L0_16x16	re-judge	succession	re-search	1	re-judge
2			Fi	Fr	Fr	re-judge	P_L0_16x16	re-search	succession	re-judge	1
3				Fi	re-judge	re-judge	re-judge	re-search	re-search	re-judge	re-judge
4		Fi	Fr	Fr	Fr	P_L0_16x16	re-judge	succession	re-search	1	re-judge
5				Fi	Fr	P_L0_16x16	re-judge	succession	re-search	1	re-judge
6			Fi	Fr	Fi	re-judge	P_L0_L0_16x8	re-search	conversion	re-judge	1
7				Fi	Fi	re-judge	P_L0_L0_16x8	re-search	succession	re-judge	1
8	Fi	Fr	Fr	Fr	Fr	re-judge	P_L0_16x16	re-search	succession	re-judge	1
9				Fi	Fi	P_L0_L0_16x8	re-judge	conversion	re-search	1	re-judge
10			Fi	Fr	Fr	re-judge	P_L0_16x16	re-search	succession	re-judge	1
11				Fi	Fi	P_L0_L0_16x8	re-judge	succession	re-search	1	re-judge
12		Fi	Fr	Fr	Fi	P_L0_L0_16x8	P_L0_L0_16x8	conversion	conversion	1	1
13				Fi	Fi	P_L0_L0_16x8	P_L0_L0_16x8	conversion	succession	1	1
14			Fi	Fr	Fi	P_L0_L0_16x8	P_L0_L0_16x8	succession	conversion	1	1
15				Fi	Fi	P_L0_L0_16x8	P_L0_L0_16x8	succession	succession	1	1

Fr: frame, Fi: field, conversion: convert from frame vector to field vector

to which the MPEG-2 encoder refers, the reference picture number is one for P-pictures and two for B-pictures.

B. Simulation conditions

The interlace video sequences, which size is 720 x 480 pixcels, were compressed first by MPEG-2 encoder (TM5 [8]) to generate bitstreams to input the proposed and conventional transcoder. We used an original jm12.1 [9] program for the conventional transcoding. It does not use the encoding information except picture type, and does an ordinary motion search and a MB type selection. The conventional transcoding had no restrictions with regard to selecting MB type and motion vector type, so its output bit-streams included the all-MB mode and all motion type, and also included all available DCT sizes. We used the modified jm12.1, which can input the first encoding information, for our proposed transcoding method. The quantization step Δ was fixed. The frame rate is 30 fps. The GOP size is 15 and its structure is IBBP(M=3, N=15) for the simulations of both the conventional and proposed methods.

C. Noise reduction generated by the type succession

Using the video sequence ”Cheerleader”, we measured the first P-picture's PSNRs of our method and a conventional method through simulation. Every MB of our method was classified by the combination of their DCT type and motion compensation type in MPEG-2. Each MB compressed with our method was compared to the MBs compressed by the conventional method located on the same image position. The relationship between the MB group and the PSNR is shown in Table VI. The meanings of MB group having “o”, “#”, and ”x” are the same as in Table III. In the “o” MB group, which succeeds the MPEG-2's DCT and motion compensation types without vector conversion, the PSNR of the proposed method is 0.67–2.04 dB higher than that of the conventional method. In the “#” MB group, which succeeds these types with vector conversion, the PSNR of the proposed method is also 0.21–1.51 dB higher. However, the PSNR difference is less than 0.18 dB in the “x” MB group in which there is no succession. These results indicate that transcoding noise is reduced in succeeding MBs whether or not the vector conversion is used.

TABLE VI
SUCCESSION MANNER AND PSNR IMPROVEMENT OF THE FIRST P-PICTURE USING PROPOSED METHOD

Group #	Num of pair MB	Proposed				Conventional		Difference	
		Succession manner		PSNR (dB)		PSNR (dB)		PSNR (dB)	
		up MB	low MB	up MB (1)	low MB (2)	up MB (3)	low MB (4)	up MB (1-3)	low MB (2-4)
0	98	o	o	43.02	43.00	41.51	41.54	1.51	1.45
1	41	o	x	40.22	38.04	38.22	37.94	1.99	0.09
2	53	x	o	37.62	40.13	37.44	38.16	0.18	1.97
3	79	x	x	37.65	37.72	37.69	37.67	-0.03	0.05
4	57	o	x	41.33	39.92	39.28	40.47	2.04	-0.54
5	34	o	x	40.62	38.24	39.42	38.50	1.19	-0.25
6	12	x	#	38.38	41.66	38.46	40.40	-0.08	1.26
7	39	x	o	37.49	38.61	37.58	37.86	-0.08	0.75
8	31	x	o	39.95	40.72	39.94	38.76	0.01	1.96
9	8	#	x	39.27	37.71	39.05	37.70	0.21	0.00
10	33	x	o	38.36	40.50	38.43	39.25	-0.07	1.25
11	36	o	x	38.66	37.69	37.98	37.69	0.67	-0.00
12	60	#	#	41.26	41.60	40.21	40.08	1.04	1.51
13	19	#	o	40.89	40.27	39.88	39.11	1.01	1.16
14	14	o	#	40.76	40.99	38.88	39.77	1.87	1.22
15	61	o	o	39.15	39.69	38.04	38.22	1.11	1.47

o: succeeded without vector conversion
#: succeeded with vector conversion
x: non-succeeded

D. Proposes method and distance from I-pictures

The relationship between the distance from an I-picture and the PSNR of a P-picture is shown in Table VII. The conventional PSNR column and the PSNR difference column are the interpolation values when the conventional method generates the same bitstream in terms of size. These inter-

TABLE VII
DISTANCE FROM I-PICTURES AND PSNR OF P-PICTURES

Seq.	Input MPEG-2			Proposed method			Conventional method	
	Dist	PSNR (dB)	Picture bits(k)	Picture bits(k)	Comp ratio.	PSNR (dB)	PSNR (dB)	Diff. (dB)
Boat	3	36.69	122.8	102.0	0.83	35.75	35.29	0.46
	6	36.40	130.7	112.7	0.86	35.19	35.11	0.08
	9	36.30	132.7	112.7	0.85	34.92	34.90	0.01
	12	36.19	140.1	122.1	0.87	34.64	34.79	-0.15
Bus	3	34.01	213.8	185.9	0.87	32.80	29.84	2.95
	6	34.07	219.2	193.9	0.88	32.39	30.09	2.29
	9	34.08	223.2	203.6	0.91	32.14	30.29	1.85
	12	34.05	208.1	198.8	0.96	31.97	30.06	1.91
Cheer Leader	3	34.14	373.8	346.9	0.93	32.31	31.88	0.43
	6	34.10	338.2	314.2	0.93	32.10	31.82	0.28
	9	34.09	348.9	320.1	0.92	32.01	31.91	0.09
	12	34.11	351.4	328.0	0.93	31.91	31.87	0.03
Plane	3	37.82	80.2	68.5	0.85	36.82	36.05	0.77
	6	37.70	80.3	70.8	0.88	36.32	35.93	0.40
	9	37.63	88.3	82.1	0.93	35.96	35.97	-0.01
	12	37.26	103.2	94.1	0.91	35.58	35.85	-0.27
Race	3	35.07	105.2	81.4	0.77	34.30	33.98	0.32
	6	35.21	105.0	82.5	0.79	34.24	34.20	0.04
	9	35.09	119.0	98.2	0.83	33.89	34.06	-0.17
	12	35.20	129.5	115.0	0.89	33.70	34.12	-0.43
Susie	3	37.56	51.0	27.7	0.54	36.80	36.39	0.40
	6	37.56	46.5	23.5	0.51	36.55	36.43	0.11
	9	37.58	44.0	23.1	0.53	36.35	36.31	0.04
	12	37.43	55.7	34.2	0.61	35.98	36.06	-0.07

TABLE VIII
COMPRESSION RATIO AND PSNR OF PROPOSED METHOD (QUANTIZATIONSTEP=8)

Seq.	Input MPEG-2		Proposed method			Conventional method	
	PSNR (dB)	Bitrate (kbps)	Bitrate (kbps)	Comp ratio.	PSNR (dB)	PSNR (dB)	Diff. (dB)
Boat	39.21	3279.9	2536.3	0.77	38.02	38.00	0.02
Bus	37.96	4796.0	4077.2	0.85	36.36	35.48	0.88
Cheer	37.89	7128.6	6146.9	0.86	36.06	35.98	0.08
Plane	40.43	2339.6	1789.5	0.76	39.21	39.19	0.02
Race	38.46	3419.5	2653.1	0.78	37.01	37.04	-0.03
Susie	40.17	1555.7	936.8	0.60	39.10	38.97	0.13
ave.	39.02	3753.2	3023.3	0.77	37.63	37.44	0.19

TABLE IX
COMPRESSION RATIO AND PSNR OF PROPOSED METHOD (QUANTIZATIONSTEP=16)

Seq.	Input MPEG-2		Proposed method			Conventional method	
	PSNR (dB)	Bitrate (kbps)	Bitrate (kbps)	Comp. ratio	PSNR (dB)	PSNR (dB)	Diff. (dB)
Boat	35.90	1666.2	1080.9	0.65	34.54	34.49	0.05
Bus	33.54	2315.0	1805.7	0.78	31.98	30.74	1.24
Cheer	33.23	3423.2	2934.3	0.86	31.57	31.39	0.18
Plane	37.25	1307.2	821.5	0.63	35.92	35.81	0.11
Race	34.69	1421.7	854.9	0.60	33.57	33.54	0.03
Susie	37.76	898.7	363.0	0.40	36.69	36.50	0.18
ave.	35.40	1838.7	1310.1	0.65	34.05	33.74	0.31

polation values are calculated using the simulation results of the conventional method. The decrease in PSNR difference between the proposed method and the conventional method is about 0.32-2.95 dB for the first P-picture when the distance is equal to three. This indicates that the proposed method is superior to the conventional method for the P-pictures that are located near I-pictures. The average PSNR improvement of the proposed method is (0.86 dB, 0.53 dB, 0.30 dB, and 0.17 dB) when the distances are (3, 6, 9, and 12) for six video sequences. Although the effect of the proposed method becomes smaller as the distance between I-picture and P-picture becomes larger, the PSNR of the proposed method is higher than that of the conventional method.

E. PSNR comparison at the same bitrate

The PSNRs of the conventional method and the proposed method are listed in Tables VIII and IX. The quantization step was set to 8 in Table VIII and to 16 in Table IX. The conventional PSNR column is the interpolation value when the conventional method generates the same bitrate. However, the PSNR of the proposed method is slightly lower than that of the conventional method only for the “Race” in Table VIII; the remaining nine simulation results show that the PSNR of the proposed method is higher than that of the conventional method. In other words, the PSNR improvement is 0.19 – 0.31 dB. These results indicate that the proposed method is superior to the conventional method.

VI. CONCLUSION

We proposed a MPEG-2 to H.264 transcoding method for interlace bit-streams intermingled with frame and field macroblocks. This method uses encoding information from MPEG-2 streams and keeps as many DCT coefficients of the original MPEG-2 bitstream as possible. The frame to field vector conversion increases the amount of MPEG-2 information succeeding MBs, which makes it possible to reduce re-quantization noise. Experimental results show that the proposed method improves PSNR by about 0.19-0.31 dB compared with a conventional method. The advantage of the proposed method is not only high transcoded video quality but also reduced CPU time in searching motion vectors.

REFERENCES

[1] P. Guilotel, et al ”Adaptive Encoders: The New Generation of MPEG-2 Encoders” SMPTE journal (April 2000)

[2] T. Yoshitome, et al ”A study of MPEG-2 to H.264 Intra Transcoding for Progressive Contents” ITE Journal, **62**,11,pp1819-1824,2008 (in Japanese)

[3] T. Yoshitome, et al ”An MPEG-2 to H.264 Transcoding Method Preserving DCT Information for Progressive Contents” ITE Journal, **63**,6,pp837-846,2009 (in Japanese)

[4] Takeshi Yoshitome, et al ”A Re-quantization Noise Reduction Method in MPEG-2 to H.264 Intra Transcoding”, IEEE Symposium on Industrial Electronics and Application (ISIEA 2009), Kuala Lumpur, Malaysia, pp.94-99, Oct 4-6, 2009

[5] Xiaoan Lu, et. al ”Fast mode decision and motion estimation for H.264 with a focus on MPEG-2/H.264 transcoding” in Proc. of ISCAS 2005, 23-26 May 2005, pp1246-1249 Vol.2

[6] Gao Chen, et al ”Efficient block size selection for MPEG-2 to H.264 transcoding” Proc. of the 12th annual ACM international conference on Multimedia

[7] Joo-Kyong Lee, et al ”Quantization/DCT Conversion Scheme for DCT-domain MPEG-2 to H.264/AVC Transcoding” IEICE Trans. Commun., **E87-B**, 7 (July 2004)

[8] MPEG-2, Test Model 5 (TM5), Doc ISO/IEC JTC/SC29/WG11/N0400, Test Model Editing Committe, April 1993.

[9] Joint Video Team(JCT), “Reference Software JM12.1”, http://iphome.hhi.de/suehring/tml/