Motion-JPEG2000 Codec Compensated for Interlaced Scanning Videos

Takuma Ishida, *Student Member, IEEE*, Shogo Muramatsu, *Member, IEEE*, and Hisakazu Kikuchi, *Member, IEEE*

Abstract—This paper presents an implementation scheme of Motion-JPEG2000 (MJP2) integrated with invertible deinterlacing. In previous work, we developed an invertible deinterlacing technique that suppresses the comb-tooth artifacts which are caused by field interleaving for interlaced scanning videos, and affect the quality of scalable frame-based codecs, such as MJP2. Our technique has two features, where sampling density is preserved and image quality is recovered by an inverse process. When no codec is placed between the deinterlacer and inverse process, the original video is perfectly reconstructed. Otherwise, it is almost completely recovered. We suggest an application scenario of this invertible deinterlacer for enhancing the sophisticated signal-to-noise ratio scalability in the frame-based MJP2 coding. The proposed system suppresses the comb-tooth artifacts at low bitrates, while enabling the quality recovery through its inverse process at high bitrates within the standard bitstream format. The main purpose of this paper is to present a system that yields high quality recovery for an MJP2 codec. We demonstrate that our invertible deinterlacer can be embedded into the discrete wavelet transform employed in MJP2. As a result, the energy gain factor to control rate-distortion characteristics can be compensated for optimal compression. Simulation results show that the recovery of quality is improved by, for example, more than 2.0 dB in peak signal-to-noise ratio by applying our proposed gain compensation when decoding 8-bit grayscale Football sequence at 2.0 bpp.

Index Terms—Energy gain compensation, invertible deinterlacing, lifting implementation, motion-JPEG2000 (MJP2), signal-to-noise ratio (SNR) scalability.

I. INTRODUCTION

INTERLACED scanning and progressive scanning are known as recording and displaying formats for motion pictures [1]–[4]. In the television broadcasting community, the video standards, such as NTSC, PAL, and SECAM signals, are based on the interlaced scanning format, whereas currently, in the visual communication community, including mobile communication, videophone, and video on the Internet, video contents and video coding standards are developed in the progressive scanning format. As the border of these communities fades away, a particular demand increases for coming and going between the interlaced and progressive scanning formats. In this work, we deal with a video coding technique to meet this demand.

The authors are with Department of Electrical and Electronics Engineering, Faculty of Engineering, Niigata University, Niigata 950-2181, Japan (e-mail: takumaro@telecom0.eng.niigata-u.ac.jp; shogo@eng.niigata-u.ac.jp; kikuchi@eng.niigata-u.ac.jp).

Video coding techniques are classified into two categories according to the way of treatment in temporal correlation: intracoding and intercoding. The intercoding is in general superior to the intracoding in terms of coding efficiency and is well observed in widely accepted standards MPEG1/2/4 and H.26x [5]. These standards adopt a hybrid of the transform-based and motion-compensated coding techniques. On the other hand, the intracoding technique also receives strong supports in the field of content editing, visual information retrieval and communication with robust error resilience. No motion compensation is employed, and every frame is encoded independently so that handling of every frame is quite easy and errors in a frame never propagate to the others.

One of the representative intracoding standards is Motion-JPEG2000 (MJP2) as Part 3 of the JPEG2000 (JP2) image coding standard, which is based on the baseline JPEG2000 Part 1 [6]–[10]. Recent explosive growth requires various kinds of scalable functionality in audio and visual coding techniques as well as high compression performance at preferable quality. To cope with these requirements, JP2/MJP2 excellently covers the signal-to-noise ratio (SNR), spatial, frequency, and component dominant (color) scalabilities. The SNR scalability enables us to obtain a video content at an arbitrarily specified bitrate with moderate quality by partial decoding of a bit stream that has been encoded at a higher bitrate. This functionality is quite useful when a viewing user wants to control the balance between quantity and quality. This paper is concerned with this SNR scalability of MJP2 for interlaced scanning video contents.

With regard to interlaced sources, there are two classes in the way of managing field pictures: field-based coding and frame-based coding. The former encodes two successive fields of a frame separately and the latter encodes every frame as a single picture. Actually, MJP2 defines both of field- and frame-based coding modes. The article [11] describes that the relative rate-distortion (R-D) performance of these two coding modes highly depends on the properties of given sources. For a picture that shows strong spatial consistency across adjacent rows, the frame-based mode is preferable to the field-based mode. For example, it is the scene containing abundant detail where a camera has little action and objects in the sequence move slowly and occupy small areas. In contrast, the field-based mode is favorable for a video sequence including a fast camera action as well as fast moving objects covering large areas in a scene with little detail. From this observation, it is recommended to introduce picture-adaptive frame/field (PAFF)

coding, which adaptively switches the coding mode depending on the characteristics of a video sequence. The followings are also the cases where the frame-based mode is advantageous.

- A sequence contains small moving objects with a still dominant background or slowly moving dominant foreground objects.
- A simple encoding structure is preferred instead of introducing a mode selector.

The frame-based mode requires a field interleaving (or field-merging) process so as to generate a frame picture from two successive fields. Unfortunately, this process causes horizontal comb-tooth artifacts at edges of moving objects [12]. In other words, frames generated through field interleaving suffer from a tricky problem because of vertical-temporal aliasing or a vertical-temporal offset between two successive fields in the same frame [3], [4]. In the case of transform-based coding such as MJP2, if the vertically high-frequency components involved with comb-tooth artifacts are quantized, the quantization errors appear annoyingly and become perceivable to the naked eye, especially at low bitrates. This fact implies that some remedy for the comb-tooth artifacts is required for making the most of the SNR scalability in MJP2 for interlaced scanning video.

So far, two different techniques have been studied for solving the comb-tooth artifacts in motion picture coding. One of them is the motion-compensated (MC) deinterlacing approach [3], [13], [14], and the other is the linear prefiltering approach [3], [12]. The MC approach shown in the article [14] takes account of an optimal quantization gain with respect to the sensitivity of the visual perception. In particular, the MC technique seems more attractive for intercoding systems such as MPEG2 because computational resources for motion compensation can be shared. Since missing lines in every field are interpolated along the motion trajectory, the MC approach performs well for both of still and moving parts. On the other hand, this approach requires a huge amount of processing and the transmission of motion vectors to reconstruct interlaced pictures. Thus, it is not necessarily appropriate to apply the MC approach to intracoding such as MJP2. In addition, when the motion estimation fails, the error grows perceptually large.

Recently, a simple MJP2 system with a prefilter was proposed by Kuge [12] to suppress these artifacts. This technique was demonstrated to be effective for decoding at a targeted bitrate. It was especially effective for low bitrate decoding.[1] However, since the scalable functionality hides decoding bitrates from encoders, nobody knows how much the decoding bitrate is while encoding. When the decoding bitrate is higher than a fixed bitrate targeted by involved filtering, the quality of a picture is degraded by the effect of lowpass filtering. Such behavior is not desirable with respect to the SNR scalability.

To solve this problem, in a previous work, we developed an invertible deinterlacing technique[2] with sampling density preservation as a preprocess for scalable frame-based coding [15], [16]. Then, we designed several deinterlacing filters and suggested a scenario of using a deinterlacer prior to a frame-based coding, where the deinterlacer was used to enhance the scalability of MJP2 [17]–[19]. With this technique, we could suppress comb-tooth artifacts at low bitrates while guaranteeing the quality recovery through the inverse process at high bitrates. Such recovered pictures at high bitrates, however, have less peak signal-to-noise ratio (PSNR) than that obtained by the conventional process with field interleaving. For example, the difference is around 2 dB when decoding 8-bit grayscale sequence *Football* in the MJP2 bitstreams at 2.0 bpp.

The purpose of this paper is to present a system that improves the quality in decoding at high bitrates from the direct application of invertible deinterlacing to MJP2, while suppressing the comb-tooth artifacts at low bitrates. To achieve this purpose, we first analyze the influence of an invertible deinterlacer to the error energy distribution in the wavelet transform domain of JP2, and propose an energy gain compensation technique for giving optimal R-D control. The energy gain factor is computed from the L2 norm of the inverse filter coefficients for the relevant subband [7], [10]. Thus, the problem can be reduced to know how the inverse filter coefficients change due to the invertible deinterlacer. As a previous work, we have already shown that invertible deinterlacing can be embedded into the discrete wavelet transform (DWT) of JP2 in some special cases [20], where the computational costs were only concerned. In this paper, it will be shown that this fact can be utilized for analyzing the influence of our deinterlacing and that a mismatching energy gain factor can be compensated, while suppressing the effect of comb-tooth artifacts within the standard bitstream format. We demonstrate that this optimal recovery method is superior to the default gain approach through several simulation results for some video sequences of different characteristics.

This paper is organized as follows. Section II outlines the invertible deinterlacing technique with sampling-density preservation, describes the scalable frame-based coding system that accompanies with the invertible deinterlacing, and summarizes the performance. Section III proposes a gain compensation technique of the invertible deinterlacing for JP2. Section IV discusses the performance evaluation of image recovery with the proposed method, where several test sequences and filter characteristics are investigated. Last, the conclusion follows in Section V.

II. REVIEW OF INVERTIBLE DEINTERLACING

In previous work, we proposed a deinterlacing technique that preserves sampling density and has invertibility [15], [16]. Here,

---

[1]In this paper, based on [10, Table 1.1], we refer to bitrates higher than 1 bpp as high bitrates, and those lower than 0.25 bpp as low bitrates.

[2]Term "invertible" is used to indicate the capability of a perfect reconstruction where an inverse system analytically exists, and to distinguish our technique from the existing deinterlacing popularly used in advanced TV receivers [3]. In fact, the design problem of deinterlacing filters can be reduced to a design problem of multidimensional two-channel maximally-decimated filter banks [16]. The lossless or reversible property is not necessarily assumed.
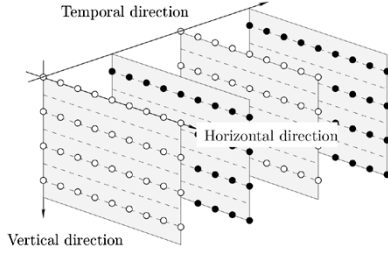
Fig. 1. Interlaced scanning, where open and closed circles are sample points in top and bottom fields, respectively.
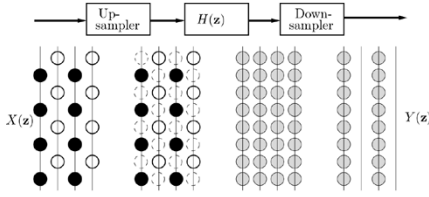

Fig. 2. Basic structure of deinterlacing with preservation of sampling density, where only the vertical-temporal plane is shown. The dotted and gray circles indicate sample points of inserted zero values and deinterlaced lines, respectively.
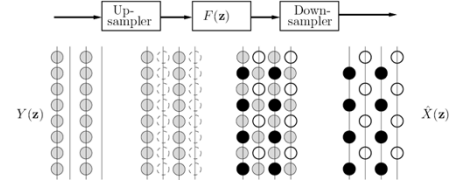

Fig. 3. Basic structure of reinterlacing, where only the vertical-temporal plane is shown. The open and closed, dotted, and gray circles indicate sample points of top line, bottom line, inserted zero value, and the other, respectively.
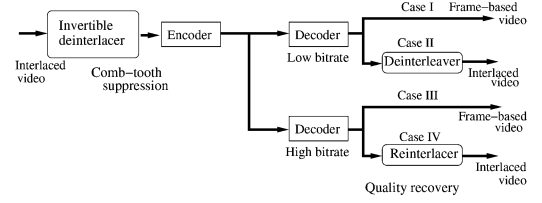

Fig. 4. Frame-based coding/decoding system with an invertible deinterlacer.

we will briefly review this technique as a preliminary to the more detailed discussion that follows. Assume that input array $X(\mathbf{z})$ is given as shown in Fig. 1.

A. Deinterlacing That Preserves Sampling Density

Fig. 2 outlines the basic structure of deinterlacing that preserves sampling density, where $\mathbf{z}$ is a three-dimensional (3-D) vector that consists of 3-D z-domain variables. The upsampler converts interlaced video signal array $X(\mathbf{z})$ into another array that is noninterlaced as shown in Fig. 2. Filter $H(\mathbf{z})$ that follows is a 3-D spatiotemporal filter, which suppresses comb-tooth artifacts. To preserve input array density, the processed video signal array is finally downsampled in the temporal direction (Fig. 2) to obtain deinterlaced array $Y(\mathbf{z})$. This process can be regarded as a generalization of field interleaving.

B. Reinterlacing (Inverse Process)

The deinterlaced array $Y(\mathbf{z})$ is encoded, transmitted and then decoded frame by frame in a frame-based codec system. The interlaced video source is expected to be reconstructed from frame-based pictures in some applications. To achieve this, we introduced an inverse converter, i.e., *reinterlacer*. Fig. 3 illustrates the basic structure. Array $Y(\mathbf{z})$ is first upsampled in the temporal direction, filtered with $F(\mathbf{z})$ and then downsampled (Fig. 3). We have shown how input array $X(\mathbf{z})$ is perfectly reconstructed from deinterlaced array $Y(\mathbf{z})$ when $H(\mathbf{z})$ and $F(\mathbf{z})$ satisfy a set of perfect reconstruction conditions in the articles [15] and [16]. If the conversion system does not suffer from any rounding error, it is theoretically possible to recover the original picture in perfect.

C. Application Scenario

An application scenario can be developed on our suggested codec system [17], [18] outlined in Fig. 4. It offers the compatibility and coexistence of interlaced and progressive scanning video formats and also provides an extension of decoding options at front-end receivers. An encoded single bit-stream is delivered to a decoder, and front-end users select the most desirable and optimum decoding among various decoding options depending on their displaying terminals and available transmission bitrates. It can be referred to as single-source encoding/multiple-display decoding. A difference from our previous work is that the encoder in this work is optimally designed for the reinterlacer. Its design affects just a few parametric values in MJP2 and gives no critical changes in the encoding procedure. It should be noted that nobody is assumed to know exact decoding conditions during the process of encoding.

Fig. 4 shows the block diagram of the process flow for the frame-based mode. For the field-based mode, we assume no special process. As illustrated in Fig. 4, decoding can be roughly divided into four cases. Note that there is no clear border of low and high bitrate decoding. Thus, an important point here is that both of interlaced and frame-based video contents are obtained from the same compressed bitstream on demand. The invertible deinterlacer ensures better low bitrate decoding because comb-tooth artifacts are suppressed. The decoded signals are displayed on a monitor directly or after passing through a reinterlacer depending on whether the displaying monitor is progressive or interlaced, respectively. The reinterlacing for low bitrate decoding may be replaced by deinterleaving that is the simplest reinterlacing to keep the comb-tooth suppression. For high bitrate decoding of interlaced video, the reinterlacer guarantees a quality recovery in terms of spatiotemporal resolution. In some cases, for example on advanced TV receivers, the reinterlaced video would further be processed to yield progressive video through another line-doubling deinterlacer [3], where the choice of the technique is beyond the scope of this paper.
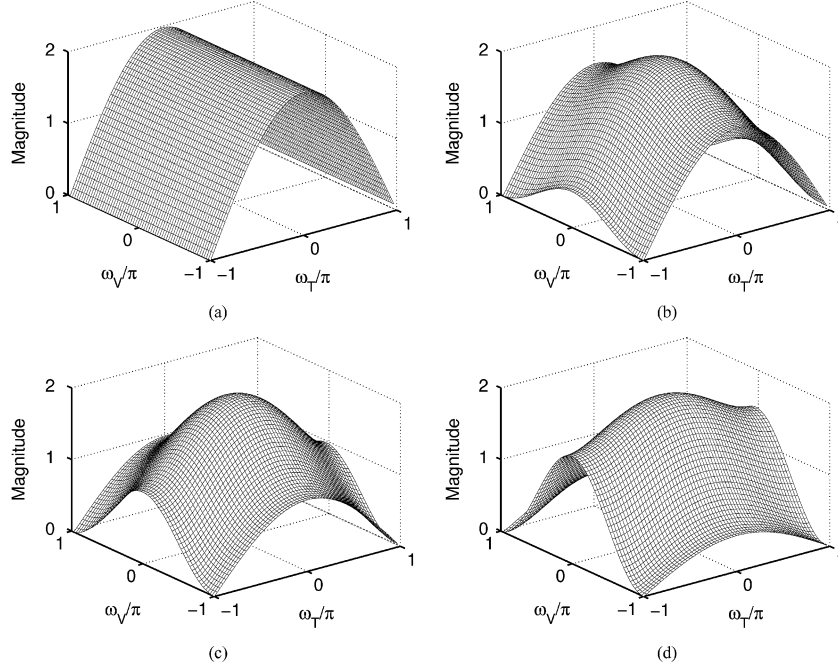
Fig. 5.   Frequency magnitude responses of deinterlacing filter $H(\mathbf{z})$. (a) $\theta = 1.0$. (b) $\theta = 0.75$. (c) $\theta = 0.5$. (d) $\theta = 0.25$.

For our system, deinterlacing and reinterlacing do not affect any encoding/decoding functionalities except for a few gain parameters, and all scalabilities equipped with MJP2 are still effective.

D. $(3 + 1)$-Tap Deinterlacing Filter

A $(3 + 1)$-tap filter is an example of a deinterlacing filter presented in the articles [15] and [16]. The transfer functions of the filter and its dual are given as follows:

$$H(\mathbf{z}) = 1 + \theta z_T^{-1} + \frac{1 - \theta}{2}\left(z_V + z_V^{-1}\right) \qquad (1)$$

$$F(\mathbf{z}) = \frac{1}{\theta} z_T^{-1}\left\{1 + \theta z_T^{-1} - \frac{1 - \theta}{2}\left(z_V + z_V^{-1}\right)\right\} \qquad (2)$$

where θ is a design parameter in range $0 < \theta \leq 1$. These filters have the following properties: 1) normalized amplitude to maintain brightness, 2) regularity to avoid the checkerboard effect [21], [22], and 3) vertical symmetry to enable symmetric border extension [21], [23]. Additionally, all samples on their original position preserve their sample values.

The frequency characteristics of the filter can be changed between temporal, vertical-temporal (V-T), and vertical lowpass filters by selecting parameter θ as shown in Fig. 5. Particularly, when $\theta = 0.5$, all of the filter coefficients are given as powers of two. When $\theta = 1.0$, simple field interleaving is obtained. We should note that this temporal filter ($\theta = 1.0$) has no capability of removing vertical high frequency components.

It has been verified that the filters can be computed through the in-place implementation as shown in Fig. 6, [20], where symmetric border extension is applied vertically. The open, closed, and gray circles indicate pixels on the top line, the bottom line of $X(\mathbf{z})$, and the odd line of deinterlaced frame
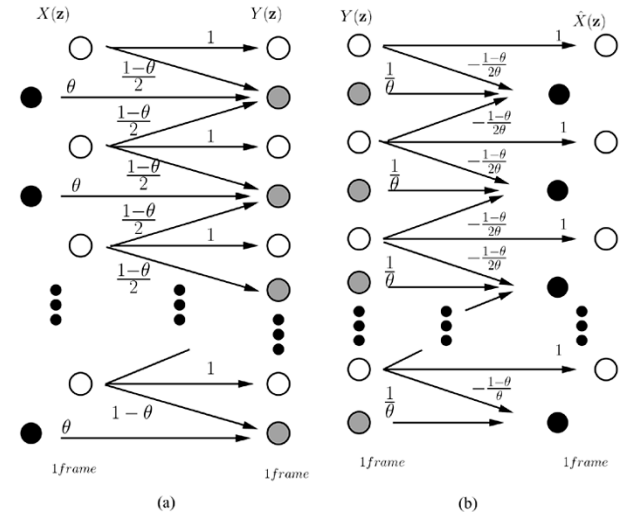

Fig. 6.   In-place computation of $(3 + 1)$-tap filters, where the symmetric extension is applied. The open, closed, and gray circles denote samples of top, bottom, and deinterlaced line, respectively, where only the vertical-temporal plane is illustrated. (a) Deinterlacing. (b) Reinterlacing.
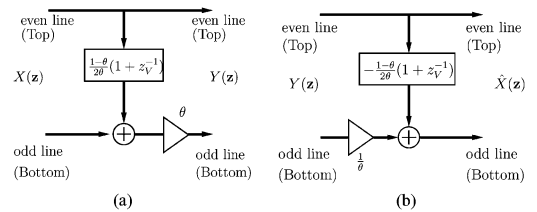

Fig. 7.   Lifting implementation of invertible scan format converters. (a) Deinterlacing. (b) Reinterlacing.

$Y(\mathbf{z})$, respectively. An odd line of deinterlaced frame pictures is obtained by the weighted sum of three lines with the values
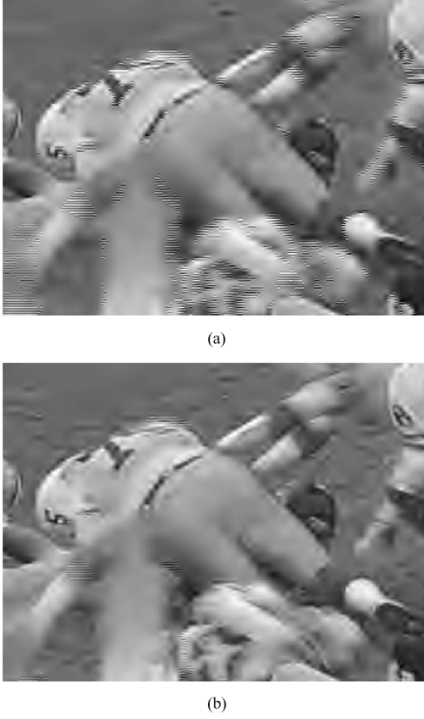
(a)

(b)

Fig. 8. Magnified views of each picture decoded at 0.1 bpp. (a) Conventional field interleaving. (b) Invertible deinterlacing $H(\mathbf{z})$ with $\theta = 0.5$.

alongside the arrows. Note that this implementation is simple and efficient because redundant computations due to downsampling and upsampling are removed from the basic structures in Figs. 2 and 3.

Fig. 7 shows the corresponding lifting structures. Each structure is composed of one prediction with simple Haar type two-tap filters. These lifting structures are equivalently applied to interleaved frame pictures in the vertical direction. We verified that comb-tooth artifacts can be avoided for low bitrate decoding with this $(3 + 1)$-tap filter $H(\mathbf{z})$ of $\theta = 0.5$. Fig. 8(a) and (b) shows pictures obtained by the field interleaving and invertible deinterlacing, respectively, where successive field pictures of a *Football* sequence (720×486 pixels, 8-bit grayscale) are used and the lossy mode of JP2 Part 1 (JJ2000 ver.4.1 [8], [25])[3] is applied to each frame. We can recognize that spot-like comb-tooth artifacts around the edges of moving objects have been suppressed by our deinterlacing. Note that the field interleaving, on the other hand, has no capability of suppressing the comb-tooth artifacts.

The filtering effect of the deinterlacer sometimes gives us blurred pictures. Fortunately, we now know that it can be smartly removed by the reinterlacer if needed. However, without any remedy, the recovered pictures show less PSNR than those obtained by field interleaving. The purpose of this work is to improve the quality of recovered images when MJP2 is used between the deinterlacer and the reinterlacer, while keeping the comb-tooth artifact suppression capability.

[3]We have fixed a bug of value δ in the forward 9/7 lifting transform module.

III. DWT GAIN COMPENSATION

The quality of recovered pictures was not superior to that obtained by field interleaving. To improve this behavior, we consider compensating the energy gain factor by embedding the invertible deinterlacer into DWT.

A. Energy Gain Factor

To minimize the distortion of a decoded picture in terms of the mean squared error (MSE) for a given bitrate, an energy gain factor is used in JP2. The energy gain factor is computed from the squared norm of the relevant subband's wavelet synthesis filter coefficients. According to the corresponding subband's energy gain factor, quantization errors of transform coefficients are weighted and the distortion of the decoded picture is estimated [7], [10]. Our scenario assumes that deinterlaced images will be reinterlaced for the quality recovery after decoding at high bitrates. However, the original weight is not necessarily optimal for controlling the R-D

$$
\begin{aligned}
\mathbf{E}(z_V) &= \begin{pmatrix} E_{00}(z_V) & E_{01}(z_V) \\ E_{10}(z_V) & E_{11}(z_V) \end{pmatrix} \\
&= \begin{pmatrix} \frac{1}{K} & 0 \\ 0 & K \end{pmatrix} \begin{pmatrix} 1 & \delta\left(1 + z_V^{-1}\right) \\ 0 & 1 \end{pmatrix} \\
&\quad \times \begin{pmatrix} z_V^{-1} & 0 \\ \gamma\left(1 + z_V^{-1}\right) & 1 \end{pmatrix} \begin{pmatrix} 1 & \beta\left(1 + z_V^{-1}\right) \\ 0 & z_V^{-1} \end{pmatrix} \\
&\quad \times \begin{pmatrix} z_V^{-1} & 0 \\ \alpha\left(1 + z_V^{-1}\right) & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \frac{1-\theta}{2}\left(1 + z_V^{-1}\right) & \theta \end{pmatrix} \\
&= \begin{pmatrix} \frac{1}{K} & 0 \\ 0 & K \end{pmatrix} \begin{pmatrix} 1 & \delta\left(1 + z_V^{-1}\right) \\ 0 & 1 \end{pmatrix} \\
&\quad \times \begin{pmatrix} z_V^{-1} & 0 \\ \gamma\left(1 + z_V^{-1}\right) & 1 \end{pmatrix} \begin{pmatrix} 1 & \beta\left(1 + z_V^{-1}\right) \\ 0 & z_V^{-1} \end{pmatrix} \\
&\quad \times \begin{pmatrix} z_V^{-1} & 0 \\ \left(\alpha + \frac{1-\theta}{2}\right)\left(1 + z_V^{-1}\right) & \theta \end{pmatrix}
\end{aligned}
$$

$$(3)$$

$$
\begin{aligned}
\mathbf{R}(z_V) &= \begin{pmatrix} R_{00}(z_V) & R_{01}(z_V) \\ R_{10}(z_V) & R_{11}(z_V) \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 \\ -\frac{1-\theta}{2\theta}\left(1 + z_V^{-1}\right) & \frac{1}{\theta} \end{pmatrix} \\
&\quad \times \begin{pmatrix} 1 & 0 \\ -\alpha\left(1 + z_V^{-1}\right) & 1 \end{pmatrix} \\
&\quad \times \begin{pmatrix} 1 & -\beta\left(1 + z_V^{-1}\right) \\ 0 & z_V^{-1} \end{pmatrix} \\
&\quad \times \begin{pmatrix} z_V^{-1} & 0 \\ -\gamma\left(1 + z_V^{-1}\right) & 1 \end{pmatrix} \\
&\quad \times \begin{pmatrix} 1 & -\delta\left(1 + z_V^{-1}\right) \\ 0 & z_V^{-1} \end{pmatrix} \begin{pmatrix} K & 0 \\ 0 & \frac{1}{K} \end{pmatrix} \\
&= \begin{pmatrix} 1 & 0 \\ -\frac{1}{\theta}\left(\alpha - \frac{1-\theta}{2\theta}\right)\left(1 + z_V^{-1}\right) & \frac{1}{\theta} \end{pmatrix} \\
&\quad \times \begin{pmatrix} 1 & -\beta\left(1 + z_V^{-1}\right) \\ 0 & z_V^{-1} \end{pmatrix} \begin{pmatrix} z_V^{-1} & 0 \\ -\gamma\left(1 + z_V^{-1}\right) & 1 \end{pmatrix} \\
&\quad \times \begin{pmatrix} 1 & -\delta\left(1 + z_V^{-1}\right) \\ 0 & z_V^{-1} \end{pmatrix} \begin{pmatrix} K & 0 \\ 0 & \frac{1}{K} \end{pmatrix}
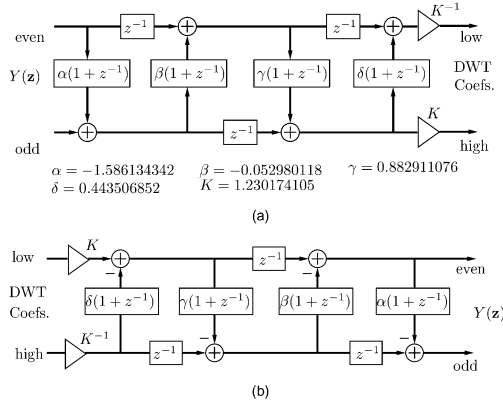\end{aligned}
$$

$$(4)$$

Fig. 9. Lifting implementation of DWT in JPEG2000 of lossy compression. (a) Forward $9/7$ transform. (b) Inverse $9/7$ transform.

performance because the synthesis process includes the reinterlacer and the synthesis filter coefficients virtually change. In order to optimize the quality recovery, the energy gain factors have to be recalculated by taking the reinterlacer into account.

B. Discrete Wavelet Transforms in JPEG2000

Let us now briefly discuss the lifting implementation of wavelet transforms employed in JP2. Two types of wavelet transforms are standardized in JP2 Part 1 [7], [10]. One is the reversible $5/3$ transform of lossless compression and the other is the irreversible $9/7$ transform of lossy compression. These transforms are implemented through lifting structures composed of two-tap lifting filters. Fig. 9 illustrates the structure of the forward $9/7$ transform and its inverse. As will be shown, our invertible deinterlacer in Fig. 7 is followed by the forward transform and the reinterlacer follows the inverse. In this work, we only consider the irreversible $9/7$ transform of the lossy compression, where a similar situation holds for any other lifting DWT including the reversible $5/3$ transform.

Note that, for the lossless compression mode in JP2, the rounding operations are inserted after each lifting step to make transform coefficients integers [10]. As for the invertible deinterlacer shown in Fig. 7(a), however, a scaling process with θ on the bottom line is required and it brings fractional bits to the transform coefficients. As a result, the application of our deinterlacers to the lossless mode requires some other special treatment for the fractional bits. Currently, this issue is under investigation.

C. Derivation of DWT Integrated With Invertible Deinterlacer

In [20], we demonstrated that it is possible to integrate the lifting DWT with the (3+1)-tap deinterlacer and the inverse with the reinterlacer, where we were concerned only with the computational issues.

All of prediction and update filters of the $9/7$ wavelet transform, are simple Haar-type two-tap filters with some different scaling factors (Fig. 9). As well, the (3+1)-tap deinterlacing filter is also composed of a Haar-type two-tap filter for the vertical direction (Fig. 7). Hence, the first lifting stage of the vertical wavelet transform can be combined with the deinterlacer.

TABLE I
SYNTHESIS FILTER COEFS. INTEGRATED WITH REINTERLACER FOR $\theta = 0.5$

n	$f_0(n)$	$f_1(n)$
0	1.11508705384879	1.47276214857289
± 1	0.65377176960586	-0.26686411943681
± 2	-0.05754352557561	-0.03144653669661
± 3	-0.15377176136265	0.01686411824750
± 4		0.04506545507608

This paper derives the compensated gain factor by using this fact.

The resulting polyphase matrix of the forward DWT integrated with the deinterlacer is given in (3), and that of the inverse transform is given in (4). Equations (3) and (4) are expressed as a Type-I polyphase matrix and a Type-II polyphase matrix, respectively [24]. From (3), one can verify that the first prediction of the wavelet transform and the deinterlacer can be combined. A similar situation holds for the inverse wavelet transform and the reinterlacer. Consequently, the synthesis filters are obtained as follows [24]:

$$(F_0(z_V)F_1(z_V)) = \left(z_V^{-1}1\right) \mathbf{R}\left(z_V^2\right). \qquad (5)$$

Table I lists the synthesis filter coefficients obtained from (4) and (5) for $\theta = 0.5$. The recalculated synthesis filter coefficients are used only for the level-one vertical wavelet transform. Additionally, Table II lists the original and the compensated energy gains for the level-one subbands, where the gain of each subband is normalized by each 1LLs gain. In each subband, the relevant gain as energy weight is just the squared norm of low- and high-pass synthesis filter coefficients. For a deeper decomposition stage, the original gain is applied in the case of multilevel DWTs.

Our proposed gain compensation relies only on the MSE measure and is calculated only from the L2 norm, i.e., the squared energy, of the synthesis filters. Unfortunately, MSE is known to be poor model for the perceptual significance of the distortion. For overcoming this problem, one can apply the so-called contrast sensitivity function (CSF) to frequency weighting in JP2, provided a final viewing condition [7, Annex J.12], [9, Annex B]. Note that CSF weights can additionally be applied to our derived gain factors for further subjective optimization [7, Annex J.14.4.1]. The following experimental results are obtained without applying CSF weights.

IV. PERFORMANCE EVALUATION

To see the improvements of our proposed method numerically, we evaluate the coding efficiency and filter performance by selecting several values of parameter θ. The results are also compared with those obtained by the conventional field interleaving ($\theta = 1.0$) and the field-based coding. In this evaluation, several video test sequences are used. The characteristics of these sequences are summarized in Table III.

A. Coding Efficiency

To demonstrate the improvement through our derived gain compensation, let us evaluate the coding efficiency in terms of

TABLE II
SYNTHESIS ENERGY GAINS OF LEVEL ONE SUBBANDS

	Original for $\theta = 1.0$	Proposal for $\theta = 0.75$	Proposal for $\theta = 0.5$	Proposal for $\theta = 0.25$
LL	1.00000000	1.00000000	1.00000000	1.00000000
HL	0.51441208	0.51441208	0.51441208	0.51441208
LH	0.51441208	0.68510677	1.03782740	2.00093023
HH	0.26461979	0.35242720	0.53387095	1.02930268

TABLE III
PROPERTIES OF TEST VIDEO SEQUENCES

Sequence	Format	Characteristics
Football	486×720i @ 60Hz 240 frames	Fast camera panning, fast moving objects
Susie	486×720i @ 60Hz 240 frames	Fixed camera view, a large moving object
Tempete	486×720i @ 60Hz 240 frames	Slow zooming out, lots of tiny moving objects
Canoa Valsesia	576×720i @ 50Hz 220 frames	Fast camera panning, a fast moving object
Rugby	576×720i @ 50Hz 220 frames	Fast camera panning, fast moving objects
Mobile & Calendar	576×720i @ 50Hz 220 frames	Slow moving camera and objects, abundant detail

PSNR for luminance components. Here, we use the sequences listed in Table III. Every frame of the sequences is encoded at 2.0 bpp and then decoded at $\{0.1, 0.25, 0.5, 1.0, 2.0\}$ bpp by the lossy mode of the JP2 Part 1 codec (JJ2000 ver.4.1 [8], [25]). Fig. 10, as objective performance evaluation, shows the R-D curves of the conventional field interleaving, our deinterlacing with default gain and our deinterlacing with the compensated gain. Table IV numerically shows the average PSNR of sequences "*Football*" and "*Mobile and Calendar*" when decoded at 0.5, 1.0, and 2.0 bpp, where the parameter is chosen as $\theta = 0.5$. For reference, the result of the field-based coding is also given. In addition, two examples of decoded and reinterlaced pictures of sequence "*Mobile and Calendar*" are shown in Fig. 11, where the default and compensated gain factors are applied for $\theta = 0.5$. Furthermore, their magnified views are shown in Fig. 12 as well as those of the original picture, pictures obtained by the field-based coding and the frame-based coding with field interleaving.

First, we discuss the quality recovery performance of reinterlacing. For sequences "*Canoa Valsesia*," "*Rugby*," and "*Football*," including fast moving objects with fast camera action, one can verify that the quality recovery with the compensated gain at higher bitrates than 1.0 bpp achieves the improvement of more than 8 dB in PSNR from the result before reinterlacing, while the results with the default gain shows worse recovery. As well, for sequences "*Mobile and Calendar*," "*Susie*," and "*Tempete*," including slow or small moving objects with little camera action, the quality with the compensated gain is recovered more than 7 dB from the result before reinterlacing at 2.0 bpp, and is also superior to the result obtained with the default gain. In summary, compared with the default gain, our proposal always improves the quality more than 0.5 dB in decoding at more than 1.0 bpp for all test sequences as shown in Fig. 10, where sequences "*Canoa Valsesia*," "*Rugby*," and "*Football*" show the PSNR improvement more than 1.0 dB. From Fig. 12(d) and (e), it is perceptually verified that our gain compensation recovered the picture better than the default gain.

Next, let us discuss a comparison among the field-based coding, the frame-based coding with the field interleaving

and our proposal. In the experimental results shown in Fig. 10(a)–(c), the sequences, which include fast camera panning as well as fast moving objects, show that both of the field-based coding and the frame-based coding with the simple field interleaving has better PSNR gain around 0.5 dB than our proposed method. In contrast, the results in Fig. 10(d)–(f), for sequences including inactive camera motion with slow or tiny moving objects, show that the proposal achieved PSNR gains in the range of 0.5–1.5 dB relative to the field-based coding over the bitrates range of 0.1–2.0 bpp.

From the above experiments, it is observed that the proposed method for $\theta = 0.5$ behaves similarly to the conventional frame-based mode with field interleaving and shows performance stable to change in the characteristics of a given sequence at the price of around 0.5 dB loss in PSNR. Remember that, although the field interleaving suffers from the comb-tooth artifacts, our proposed method can suppress those disagreeable effects in the same frame-based coding mode. The detailed discussion of the deinterlacing filter performance is made in the following subsection.

B. Filter Performance

To demonstrate the effect of deinterlacing filters with the proposed gain compensation, we evaluate both of the comb-tooth artifact suppression capability and the coding efficiency for several choices of parameter θ.

1) Performance of Comb-Tooth Suppression: To see how the parameter value influences to the comb-tooth suppression capability, we evaluate decoding results with several deinterlacing filters by selecting θ among the value 1.0, 0.75, 0.5, and 0.25. In the first experiment, a frame of successive two fields in sequence "*Football*" is used for showing the deinterlacing effect as the case of pictures including fast moving objects with fast camera action. The frame is encoded at 2.0 bpp and then decoded at 0.25 bpp.

Fig. 13 shows the whole decoded frame pictures for $\theta = 1.0$ and 0.25, and Fig. 14 shows the magnified views of the decoded pictures at 0.25 bpp obtained by using the field-based mode and
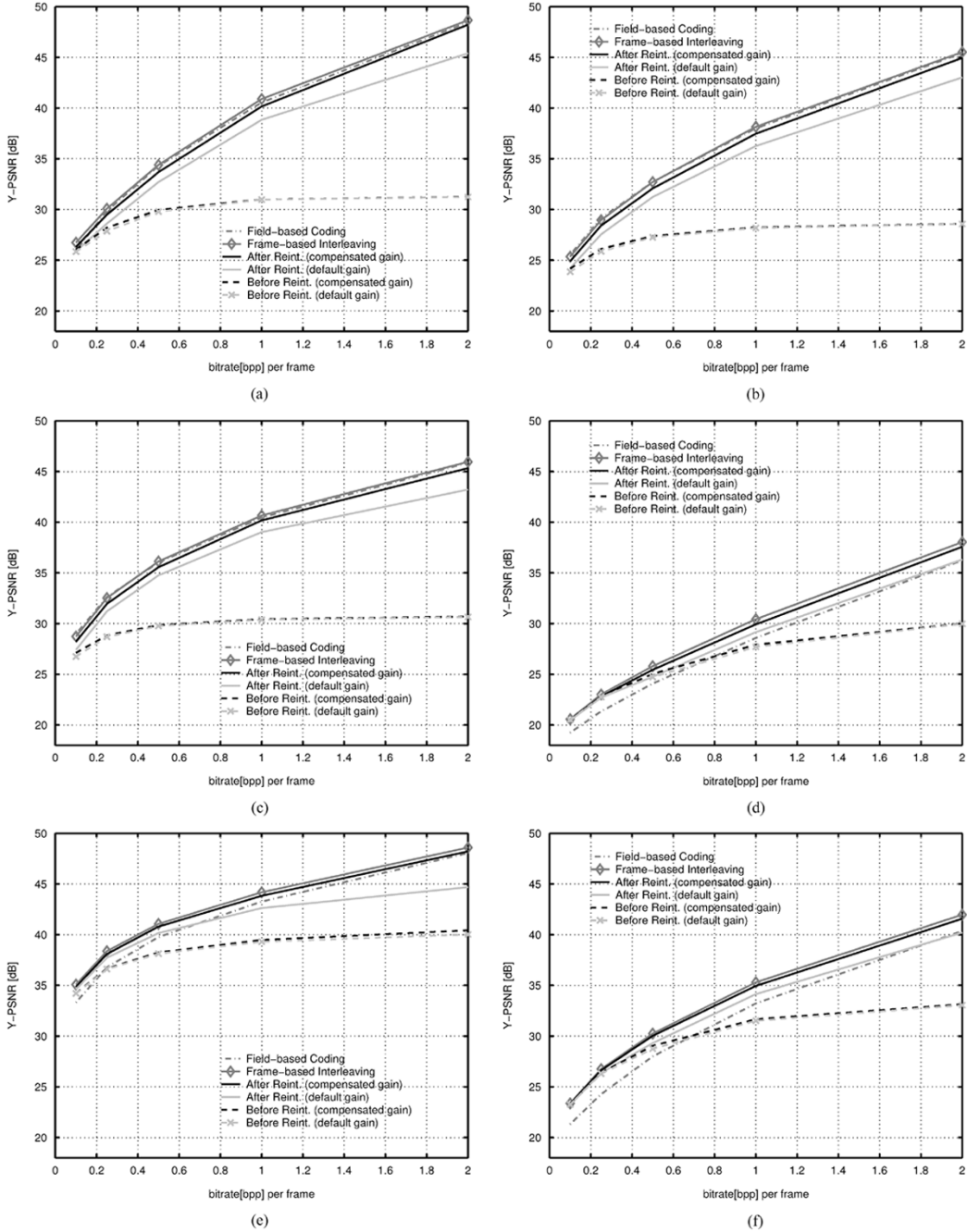
Fig. 10. Coding efficiency for $\theta = 0.5$, where the R-D curves are obtained by taking the average of PSNR only for the luminance component (Y) over whole frames of each sequence. (a) *Canoa Valsesia* (576 × 720) luminance. (b) *Rugby* (576 × 720) luminance. (c) *Football* (486 × 720) luminance. (d) *Mobile and Calendar* (576 × 720) luminance. (e) *Susie* (486 × 720) luminance. (f) *Tempete* (486 × 720) luminance.

the frame-based mode with the deinterlacing filters, where Fig. 14(a) shows, for reference, the result of the field-based mode, and Fig. 14(b)–(e) shows the decoded frame pictures of the frame-based mode with deinterlacing filter of θ in $\{1.0, 0.75, 0.5, 0.25\}$, respectively. From Fig. 13, one can verify that the capability of the comb-tooth suppression around the moving objects is still maintained by our proposed gain compensation with θ smaller than one. It is also observed from Fig. 14 that, as the deinterlacing filter characteristic reaches to the temporal lowpass one ($\theta = 1.0$), the capability to remove

the vertical high frequencies decreases so that comb-tooth artifacts remain more. In contrast, the vertical lowpass filter used in Fig. 14(e) for $\theta = 0.25$ significantly removes the comb-tooth artifacts compared with the others.

To evaluate the deinterlacing effect for a sequence including slow moving objects with slow camera action, the second experiment shows some decoded results of a frame of successive two fields in sequence "*Mobile and Calendar*" for different values of parameter θ, where the frame is encoded at 2.0 bpp and then decoded at 0.25 bpp. Fig. 15 shows the whole decoded pictures

TABLE IV
AVERAGE PSNR (IN DECIBELS) OF FRAMES DECODED AT MIDDLE- AND HIGH-BITRATES. FOR DEINTERLACING, $\theta = 0.5$ WAS USED

| *Football*	0.5 bpp	1.0 bpp	2.0 bpp	*Mobile & Calendar*	0.5 bpp	1.0 bpp	2.0 bpp
Field-based	36.04	40.45	45.86	Field-based	24.09	28.59	36.24
Interleaving	36.13	40.66	45.95	Interleaving	25.80	30.44	38.04
After Reint. (proposal)	35.57	40.18	45.34	After Reint. (proposal)	25.45	29.93	37.58
After Reint. (default)	34.76	39.03	43.22	After Reint. (default)	24.77	29.16	36.32
Before Reint. (proposal)	29.84	30.44	30.70	Before Reint. (proposal)	25.06	27.93	30.02
Before Reint. (default)	29.77	30.39	30.66	Before Reint. (default)	24.74	27.71	29.99
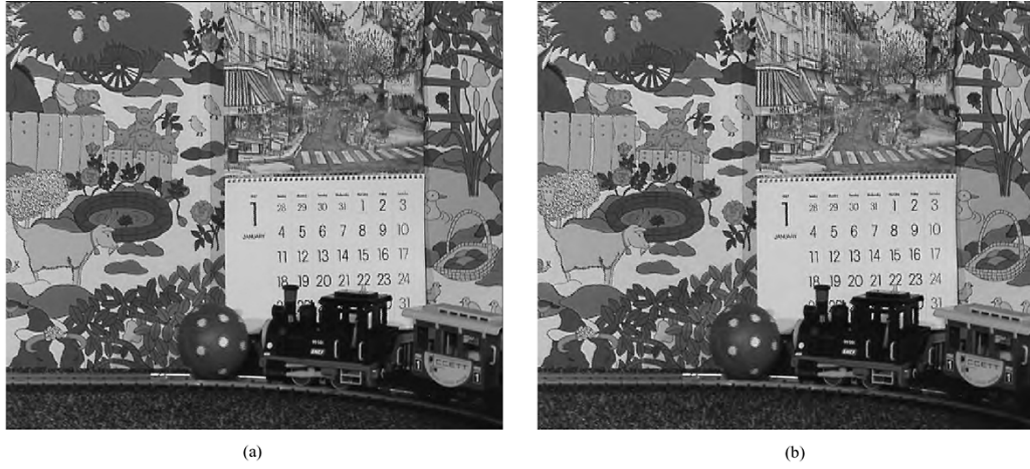
Fig. 11. Decoded pictures at 1.0 bpp for *Mobile and Calendar*. (a) After reinterlacing, where $\theta = 0.5$ (default gain). (b) After reinterlacing, where $\theta = 0.5$ (compensated gain).
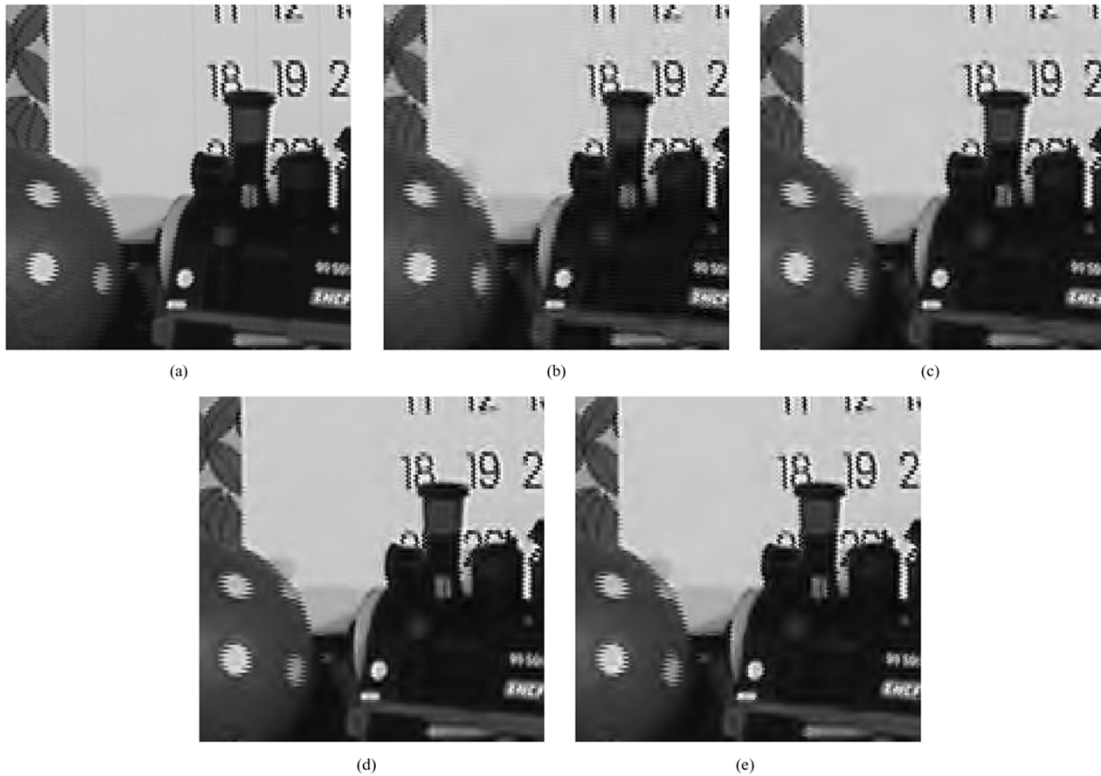

Fig. 12. Magnified views of decoded pictures at 1.0 bpp for *Mobile and Calendar*. (a) Original (no compression). (b) Field-based coding. (c) $\theta = 1.0$ (field interleaving). (d) After reinterlacing, where $\theta = 0.5$ (default gain). (e) After reinterlacing, where $\theta = 0.5$ (compensated gain).

(a)   (b)

Fig. 13.   Simulation results with the gain compensation decoded at 0.25 bpp. (a) $\theta = 1.0$ (field interleaving). (b) $\theta = 0.25$.
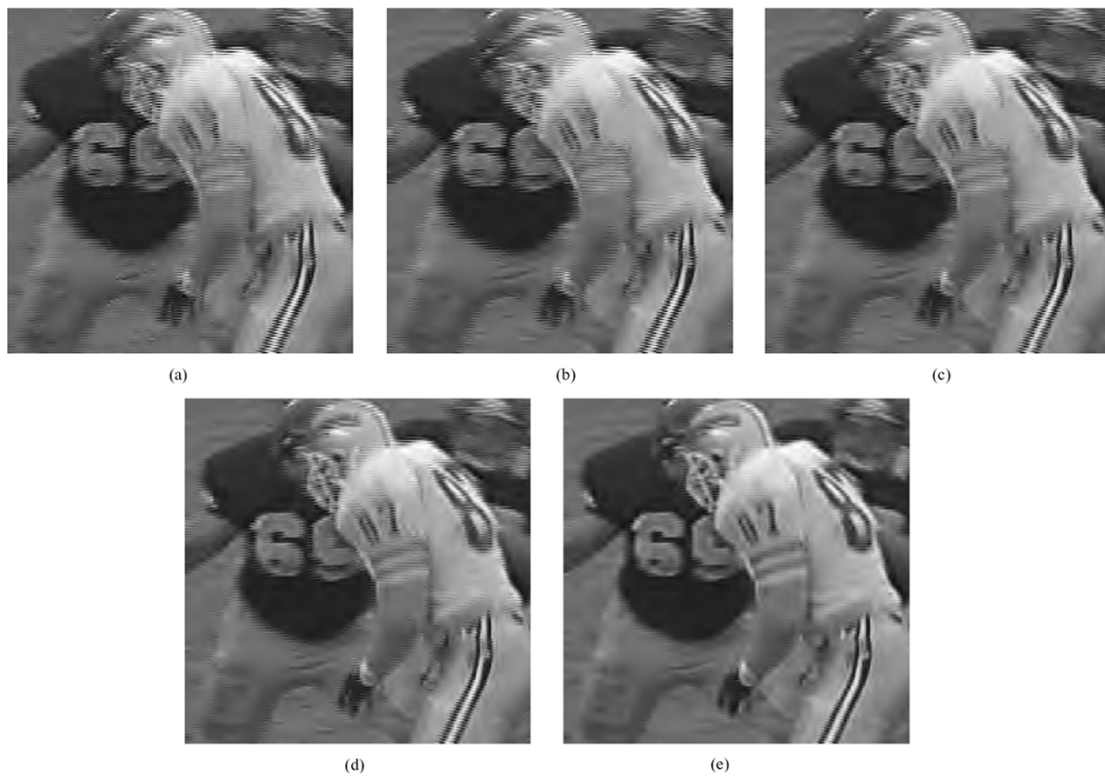

(a)   (b)   (c)

(d)   (e)

Fig. 14.   Magnified views decoded at 0.25 bpp for several choices of parameter θ. (a) Field-based coding. (b) $\theta = 1.0$ (field interleaving). (c) $\theta = 0.75$. (d) $\theta = 0.5$. (e) $\theta = 0.25$.

(a)   (b)
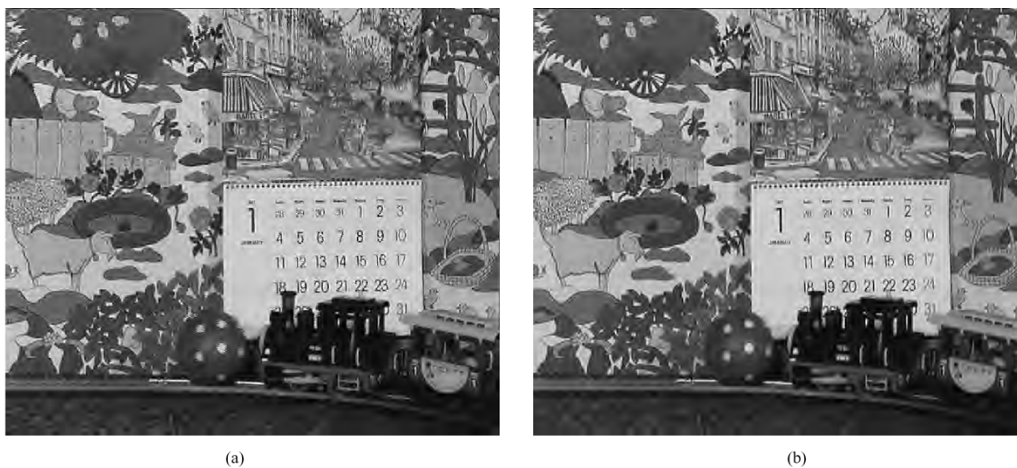
Fig. 15.   Simulation results with the gain compensation decoded at 0.25 bpp. (a) $\theta = 1.0$ (field interleaving). (b) $\theta = 0.25$.
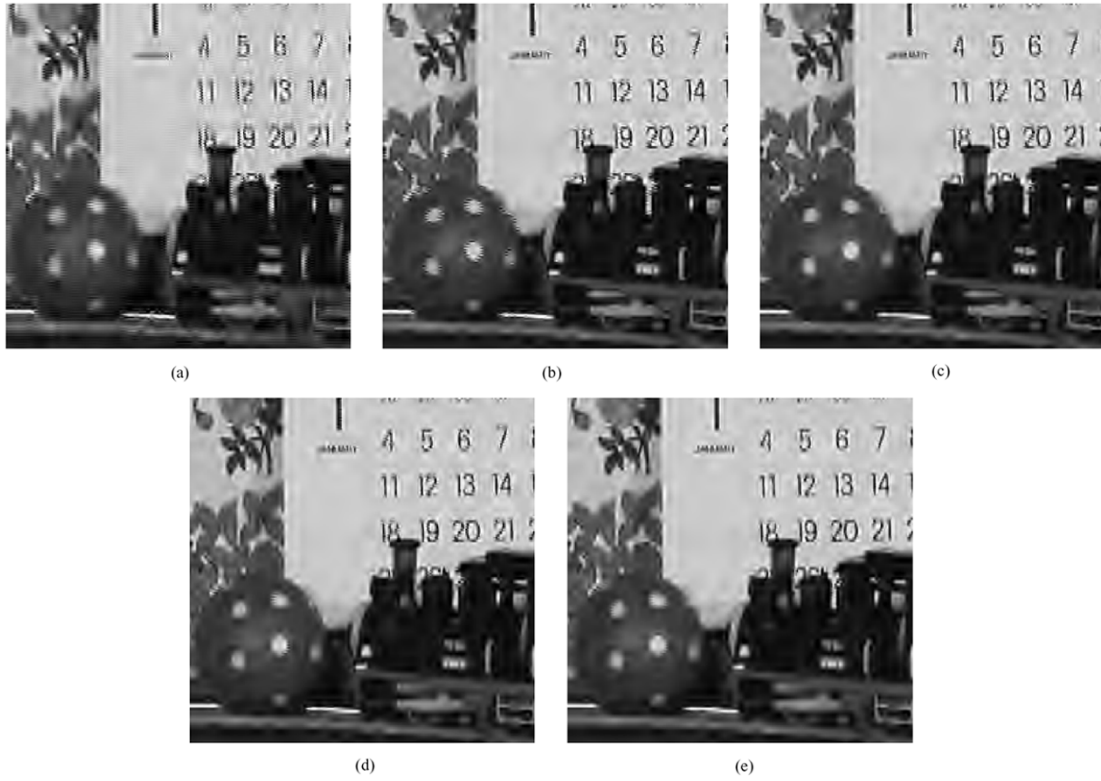
Fig. 16. Magnified views decoded at 0.25 bpp for several choices of parameter θ. (a) Field-based coding. (b) $\theta = 1.0$ (field interleaving). (c) $\theta = 0.75$. (d) $\theta = 0.5$. (e) $\theta = 0.25$.
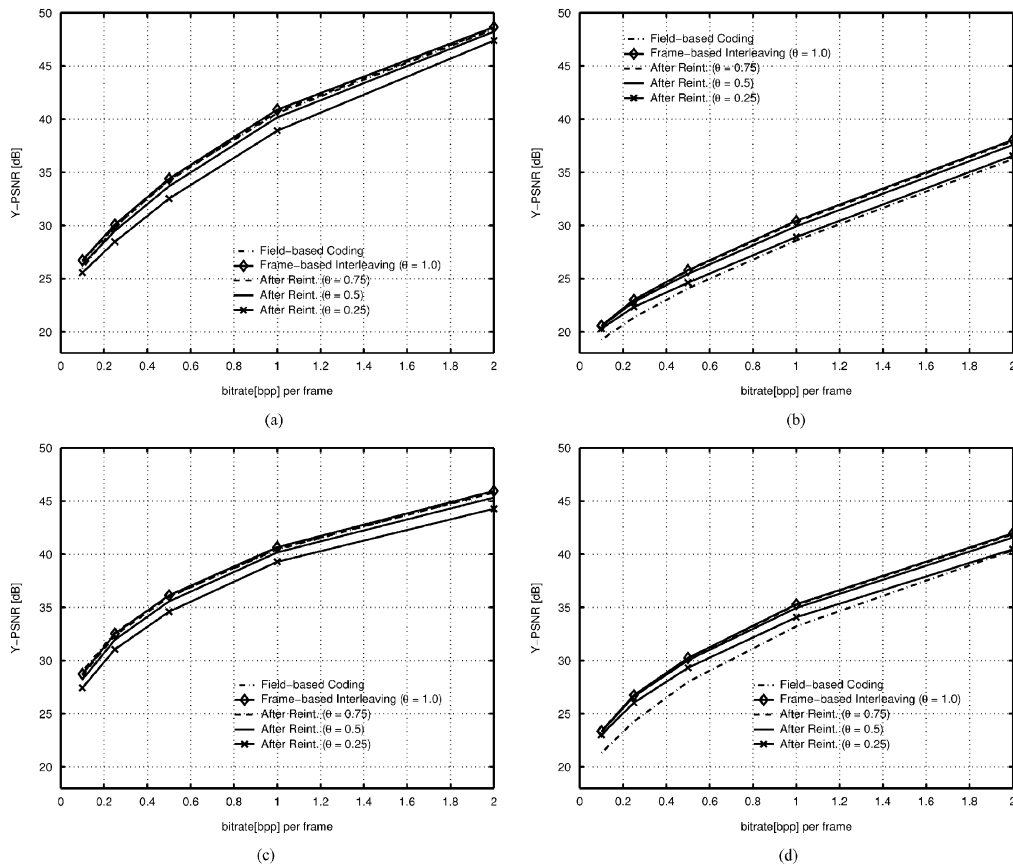

Fig. 17. Coding efficiency for several choices of parameter θ, where the R-D curves are obtained by taking the average of PSNR only for the luminance component (Y) over whole frames of each sequence. (a) R-D curve (*Canoa Valsesia*), (b) R-D curve (*Mobile and Calendar*). (c) R-D curve (*Football*). (d) R-D curve (*Tempete*).

for $\theta = 1.0$ and 0.25 decoded at 0.25 bpp, and Fig. 16 shows the magnified views of the results obtained by using the field-based mode and the frame-based mode with the deinterlacing filters, where Fig. 16(a) shows the result of the field-based mode for reference, and pictures in Fig. 14(b)–(e) shows the results of our proposed method for $\theta = 1.0, 0.75, 0.5$, and 0.25, respectively. One can verify that the comb-tooth artifacts causing around and inside of the rolling ball are significantly suppressed for a small θ.

2) Coding Efficiency: Last, we again give several R-D curves in terms of PSNR in Fig. 17 for evaluating the coding efficiency of different values of parameter θ.

Here, we use sequences "*Canoa Valsesia*," "*Mobile and Calendar*," "*Football*," and "*Tempete*." Every frame is encoded at 2.0 bpp, and then decoded at 0.1, 0.25, 0.5, 1.0, and 2.0 bpp. Although a filter of a small θ suppresses comb-tooth artifacts, PSNR of decoded frames is inferior to that of a filter with a large θ. This is because, as θ approaches to zero, the reinterlacing filter coefficients become infinity and the recovery performance becomes sensitive to errors. We finally find a trade-off between the comb-tooth-artifact suppression and the quality of recovery.

V. CONCLUSION

We presented a MJP2 codec system with invertible deinterlacing in this paper. To improve the quality of recovery, we proposed an energy gain compensation technique. We verified that the quality recovery can be improved through the compensation for the energy gain factors of subbands by taking the influence of reinterlacing into account. We have maintained the SNR scalability of the frame-based MJP2 coding system for interlaced scanning videos and shown better quality recovery than the default gain, which still keeps the filtering effect for suppressing the comb-tooth artifacts within the standard MJP2 bitstream format. We also found that, as a (3+1)-tap deinterlacing filter becomes close to a vertical filter ($\theta \to 0$), the comb-tooth suppression capability increases. On the other hand, as a filter becomes close to a temporal filter ($\theta \to 1.0$), the quality recovery is gained much more.

As future work, we will investigate a lossless mode of our deinterlacing technique integrated to MJP2 so as to realize a reversible frame-based coding system equipped with a similar scalable functionality for interlaced scanning videos. Currently, however, we suffer from a few problems, for instance, the bit increasing due to the scaling process with θ for odd (bottom) line shown in Fig. 7(a). As another interest, we would also like to consider applying the gain compensation technique to our adaptive invertible deinterlacing with variable coefficients [18], [19]. In regard to this, we are required to adaptively manage the gain compensation; that is, we have to choose a region of comb-tooth areas and compensate the region with a suitable gain by taking each boundary processes into account.

REFERENCES

[1] A. M. Tekalp, *Digital Video Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
[2] Y. Wang, J. Ostermann, and Y. Zhang, *Video Processing and Communications*. Englewood Cliffs, NJ: Prentice-Hall, 2002.
[3] G. de Haan and E. B. Bellers, "Deinterlacing-an overview," *Proc. IEEE*, vol. 86, no. 9, pp. 1837–1857, Sep. 1998.
[4] L. Vandendorpe and L. Cuvelier, "Statistical properties of coded interlaced and progressive image sequences," *IEEE Trans. Image Process.*, vol. 8, no. 6, pp. 749–761, Jun. 1999.
[5] V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standard Algorithms and Architectures*. Norwell, MA: Kluwer, 1997.
[6] T. Fukuhara, K. Katoh, S. Kimura, K. Hosakam, and A. Leung, "Motion-JPEG2000 standardization and target market," in *IEEE Proc. Int. Conf. Image Processing*, 2000, pp. II57–II60. TA02.08.
[7] *Information Technology—JPEG2000 Image Coding System: Core Coding System*, Oct. 2002. ITU-T T.800 Recommendation.
[8] *Information Technology—JPEG2000 Images Coding System: Reference Software*, 2001. ISO/IEC FDIS 15 444-5 Final Draft International Standard.
[9] *Motion JPEG2000 Final Committee Draft 1.0*, Mar. 2001. ISO/IEC JTC1/SC 29/WG1 N2117.
[10] D. Taubman and M. W. Marcellin, *JPEG2000 Image Compression Fundamentals, Standards and Practice*. Norwell, MA: Kluwer, 2002.
[11] D. Marpe, V. George, H. L. Cycon, and K. U. Barthel, "Performance evaluation of motion-JPEG2000 in comparison with H.264/AVC operated in pure intra coding mode," presented at the SPIE Wavelet Applications in Industrial Processing, Oct. 2003.
[12] T. Kuge, "Wavelet picture coding and its several problems of the application to the interlace HDTV and the ultra-high definition images," presented at the IEEE Int. Conf. Image Processing, 2002. WA-P2.1.
[13] L. Vandendorpe, L. Cuvelier, B. Maison, and P. Delogne, "Coding of deinterlaced image sequences," in *Proc. IEEE Int. Conf. Image Processing*, Nov. 1994, pp. 943–946.
[14] L. Vandendorpe, L. Cuvelier, and B. Maison, "human visual weighted quantization for transform/subband image coding revisited for interlaced pictures," *IEEE Trans. Image Process.*, vol. 7, no. 2, pp. 222–225, Feb. 1998.
[15] S. Muramatsu, T. Ishida, and H. Kikuchi, "A design method of invertible de-interlacer with sampling density preservation," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 4, May 2002, pp. 3277–3280.
[16] ——, "Invertible deinterlacer with sampling-density preservation: theory and design," *IEEE Trans. Signal Process.*, vol. 51, no. 9, pp. 2343–2356, Sep. 2003.
[17] H. Kikuchi, S. Muramatsu, T. Ishida, and T. Kuge, "Reversible conversion between interlaced and progressive scan formats and its efficient implementation," in *Proc. EUSIPCO*, vol. 3, 2002, pp. 275–278.
[18] T. Ishida, S. Muramatsu, H. Kikuchi, and T. Kuge, "Invertible deinterlacing with variable coefficients and its lifting implementation," presented at the IEEE Int. Conf. Acoustics, Speech, Signal Processing, Apr. 2003.
[19] T. Ishida, T. Soyama, S. Muramatsu, H. Kikuchi, and T. Kuge, "A lifting implementation of variable-coefficient invertible deinterlacer with embedded motion detector," *IEICE Trans. Fundamentals*, vol. E86-A, pp. 1942–1948, Aug. 2003.
[20] T. Soyama, T. Ishida, S. Muramatsu, H. Kikuchi, and T. Kuge, "Lifting implementation of invertible deinterlacing," *IEICE Trans. Fundamentals*, vol. E86-A, no. 4, pp. 779–786, Apr. 2003.
[21] G. Strang and T. Nguyen, *Wavelets and Filter Banks*. Cambridge, MA: Wellesley-Cambridge, 1996.
[22] Y. Harada, S. Muramatsu, and H. Kiya, "Multidimensional multirate filter and filter bank without checkerboard effect," in *Proc. EUSIPCO*, Sep. 1998, pp. 1881–1884.
[23] H. Kiya, K. Nishikawa, and M. Iwahashi, "A development of symmetric extension method for subband image coding," *IEEE Trans. Image Process.*, vol. 3, no. 1, pp. 78–81, Jan. 1994.
[24] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
[25] Cannon, EPFL, and Ericsson [Online]. Available: http://jj2000.epfl.ch
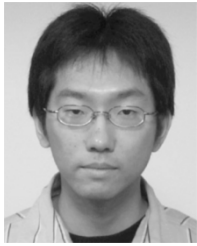
Takuma Ishida (S'02) was born in Niigata, Japan, in 1978. He received the B.E. and M.E. degrees from Niigata University, Niigata, in 2001 and 2003, respectively, where he is currently pursuing the Dr.Eng. degree.

Since 2004, he has been a Research Fellow of the Japan Society for the Promotion of Science (JSPS Research Fellow). His research interests include image and video processing, image coding, and VLSI architecture.

Mr. Ishida is a member of the Institute of Electronics, Information and Communication Engineers.

Shogo Muramatsu (M'99) received the B.E., M.E., and Dr.Eng. degrees in electrical engineering from the Tokyo Metropolitan University, Tokyo, Japan, in 1993, 1995, and 1998, respectively.

From 1997 to 1999, he was with Tokyo Metropolitan University. In 1999, he joined Niigata University, Niigata, Japan, where he is currently an Associate Professor of electrical and electric engineering with the Faculty of Engineering. From 2003 to 2004, he was a Visiting Scientist with the Department of System and Information Engineering, University of Florence, Florence, Italy. His research interests are in digital signal processing, multirate systems, image and video processing, and VLSI architecture.

Dr. Muramatsu is a member of the Institute of Electronics, Information and Communication Engineers, the Information Processing Society of Japan, and the Institute of Image Information and Television Engineers.

Hisakazu Kikuchi (M'84) received the B.E. and M.E. degrees from Niigata University, Niigata, Japan, in 1974 and 1976, respectively, and the Dr.Eng. degree in electrical and electronic engineering from the Tokyo Institute of Technology, Tokyo, Japan, in 1988.

From 1976 to 1979, he was with the Information Processing Systems Laboratory, Fujitsu, Japan. Since 1979, he has been with Niigata University, where he is a Professor of electrical engineering. He was a Visiting Scientist with the Electrical Engineering Department, University of California, Los Angeles, from 1992 to 1993. He has held a visiting professorship at Chongqing University of Posts and Telecommunications, China, since 2002. He served as the Chair of the Circuits and Systems Society, IEICE, in 2000, the General Chair of the Digital Signal Processing Symposium, IEICE, in 1988, and the General Chair of the Karuizawa Workshop on Circuits and Systems, IEICE, in 1996. His research interests are in digital signal processing and image/video processing, as well as ultrawideband systems.

Dr. Kikuchi is a member of Institute of Electronics, Information and Communication Engineers (IEICE), the Institute of Image Information and Television Engineers of Japan, the Institute of Image Electronics Engineers of Japan, the Japan Society for Industrial and Applied Mathematics, the Research Institute of Signal Processing, and SPIE.