PAPER

Moving Picture Coding by Lapped Transform and Edge Adaptive Deblocking Filter with Zero Pruning SPIHT

Nasharuddin ZAINAL[†a)], ***Nonmember***, **Toshihisa TANAKA**[††], ***and*** **Yukihiko YAMASHITA**[†], ***Members***

SUMMARY We propose a moving picture coding by lapped transform and an edge adaptive deblocking filter to reduce the blocking distortion. We apply subband coding (SBC) with lapped transform (LT) and zero pruning set partitioning in hierarchical trees (zpSPIHT) to encode the difference picture. Effective coding using zpSPIHT was achieved by quantizing and pruning the quantized zeros. The blocking distortion caused by block motion compensated prediction is reduced by an edge adaptive deblocking filter. Since the original edges can be detected precisely at the reference picture, an edge adaptive deblocking filter on the predicted picture is very effective. Experimental results show that blocking distortion has been visually reduced at very low bit rate coding and better PSNRs of about 1.0 dB was achieved.

key words: *lapped transform (LT), adaptive deblocking filter, zero pruning set partitioning in hierarchical trees (zpSPIHT), motion compensated prediction*

1. Introduction

The necessity of coding moving picture at low bit rate has become significant since the demand for applications such as mobile communications, video applications, and multimedia distance learning has increased rapidly. The limited infrastructure (wired or wireless communications) and high cost are the main reasons why many researchers concentrate on developments in effective video communication. For still picture compression, the Joint Photographic Experts Group (JPEG) standard has been established by International Organization for Standardization (ISO) and International Electro-Technical Commission (IEC). The performance of JPEG generally degrades at low bit rates mainly because of the underlying block-based discrete cosine transform (DCT) scheme [1].

Recently, the wavelet transform has emerged as an excellent technology within the field of picture compression. Wavelet-based coding provides substantial improvements in picture quality at low bit rates. An example of the wavelet-based still picture coding is the JPEG2000 standard [2]. The disadvantage of JPEG2000 is the increase in computational complexity of the coding scheme that is not justified the wide replacement of the DCT-based JPEG [3]–[5]. On the other hand, wavelet-based coders improve the picture quality at low bit rates because of overlapping basis functions and better energy compaction property of wavelet transform [6]. Wavelet coding can effectively eliminate the blocking distortion since basis in adjacent blocks overlap one another.

The fundamental concept behind subband coding (SBC) is to split up the frequency band of a signal and then code each subband using a coder. The bit rate must be matched accurately to the statistics of the band [7]. The SBC has been widely used in speech coding and later in picture coding because of its inherent advantages, namely variable bit assignment among the subbands as well as coding error confinement within the subbands [8]. Methods very similar and closely related to SBC have been proposed by many researchers as in [9]–[12] using filters. The filters are specifically designed to improve the filter banks and subband techniques. Orthogonality is another useful requirement since orthogonal filters, in addition to preservation of energy, implement a unitary transform between the input and the subbands [13] as in lapped orthogonal transform (LOT). In most cases, however, some residual artifacts are still visible [14] because the basis functions do not decay exactly to zero at their boundaries. One way to force the basis functions to decay to zero is by using biorthogonal transforms. Malvar suggested the lapped biorthogonal transform (LBT) by introducing a $\sqrt{2}$ scaling of the first antisymmetric basis function of the DCT as shown in [15].

For moving picture compression, Overlapped Block Motion Compensation (OBMC) is employed in almost all wavelet and subband moving picture coding to alleviate discontinuity between the motion compensated blocks [16]. Another basic approach to reduce the blocking distortion is by introducing an efficient deblocking filter. Various post-filtering schemes for reducing the blocking distortions have already been proposed to improve the visual quality of the block-based coded pictures either from the transform coding or the motion compensated prediction in the decoder [17]. We deploy an edge adaptive deblocking filter as the loop filters because post filter does not have any effect on the coding efficiency of the difference picture for SBC. The algorithm is simple and easy to use in existing methods compared to OBMC that increased the computational complexity [18]. In moving pictures, generally, the blocking distortions of the previous picture are propagated to the current picture and are located at any position in a block due to motion compensated prediction. In this paper, approaches to reducing

Manuscript received September 15, 2009.
Manuscript revised January 14, 2010.

[†]The authors are with the Department of International Development Engineering, Graduate School of Science and Engineering, Tokyo Institute of Technology, Tokyo, 152–8550 Japan.
[††]The author is with the Department of Electrical and Electronic Engineering, Tokyo University of Agriculture and Technology, Koganei-shi, 184–8558 Japan.
a) E-mail: nash@eng.ukm.my

DOI: 10.1587/transinf.E93.D.1608

the blocking distortions are justified for moving picture coding. We apply an edge adaptive deblocking filter to the predicted picture. Precise detection of edges is essential for effective application of the adaptive deblocking filter. Since the lapped transform (LT) and the DWT have no blocking distortion, they allow precise edge detection by using not predicted picture but reference picture which is free from blocking distortion. However, precise edge detection is difficult for the DCT since there are many edges due to the blocking distortion produced by the transform.

Embedded Zerotree Wavelet (EZW) coding introduced by Shapiro is known as a very effective and computationally simple technique for image compression [19]. EZW realized the coding using the character of tree structure of the wavelets coefficients [20]. Thereafter, Said et al. introduced Set Partitioning in Hierarchical Trees (SPIHT), which uses the special structure of the wavelet coefficients but has better efficiency than EZW. The quantity of coding by SPIHT is easy to control since the output is embedded completely. The SPIHT is, however, more efficient than the EZW because it is necessary to code the output bit stream of the EZW with arithmetic coding but not to the output bit stream of the SPIHT [21]. Coefficients of the difference picture are mostly small values. Efficient coding and low computational complexity could be achieved if we pruned the zero nodes after quantization. We proposed this method and called it zero pruning SPIHT (zpSPIHT).

In Sect. 2, we explain the detail of the proposed methods. In Sect. 3, we show its advantages by experimental results. In Sect. 4, we provide conclusions and future works.

2. Proposed Method

We adopt the similar scheme with MPEG-2 coding [22] but with addition of several refinements. Figure 1 shows the proposed method. The main part of the proposal is the combination of LT which is LBT and SPIHT for I-picture or zpSPIHT for P-picture replacing DCT, and Huffman and run length codings as usually used in MPEG-2. Most of works for moving picture coding on LT deployed LOT such as in [16] but none on LBT. In order to improve the picture quality, an edge adaptive deblocking filter [23] for the predicted picture is also introduced to the scheme as shown in Fig. 1.

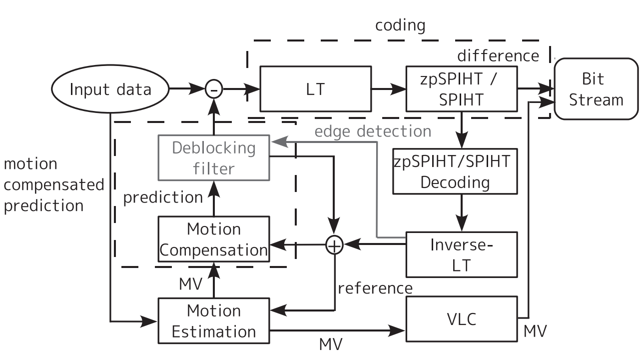


Fig. 1 Diagram of proposed encoding method.

First of all, the picture of the beginning of a video or the picture used as a starting point is coded into an I-picture, as also a basic procedure of MPEG-2. I-picture is reversely converted by the local decoder, and temporarily memorized by the picture memory as the reference picture. Simultaneously, the motion vectors and the difference picture between the target picture and the predicted picture obtained from the reference pictures and the motion vectors are coded. This method reduces temporal redundancy and increases the compression performance. After block matching and prediction by motion vector, a deblocking filter is introduced to reduce the blocking distortion from the difference pictures. We do not have to deblock the DCT blocking distortion and only concentrate on the blocking distortion from block matching at the motion compensate prediction [24], [25].

2.1 Lapped Transform

LT is adopted in the proposed method instead of DCT to gain better performance with an acceptable complexity of computation. The LT is used for compression coding in the case of mainly inclining information for every frequency band. The coding is considered efficient when energy inclines toward the low frequency. Subband transform divided a nonstationary signal into a 8×8 subband coefficients. Rearrangement of the frequencies into wavelet-like structure for the next process is performed. As the result, in the next stage of coding process, the coding is carried out easily.

2.2 zpSPIHT

We propose zpSPIHT to encode P-pictures. We introduce an arithmetic coder to both SPIHT and zpSPIHT. The pruning of quantized zeros is carried out in order to obtain efficient coding since there are many zeros in quantized coefficients of the difference picture. The fundamental of the SPIHT is a tree structure called spatial orientation tree as shown in Fig. 2. Each node of the tree corresponds to a pixel and

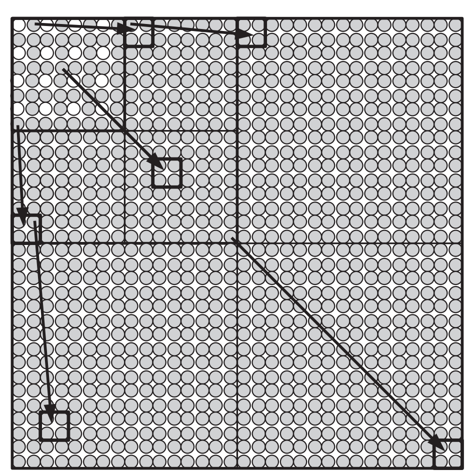


Fig. 2 Spatial orientation tree.

is identified by the pixel coordinate. The tree structure of the proposed method is as the same as the SPIHT. The following sets of coordinates are used to present the coding method:

- $\mathcal{O}(i, j)$: set of coordinates of all offspring of the node (i, j);
- $\mathcal{D}(i, j)$: set of coordinates of all descendants of the node (i, j);
- $\mathcal{H}$: set of coordinates of all spatial orientation tree roots (nodes in the highest pyramid level).

The parameters that are used for calculations are as the following:

- $f_{i,j}$: transform coefficient,
- $v_{i,j}$: quantized transform coefficient,
- q: quantization coefficient,
- z: threshold which is set to zero level,
- n: binary digit of bit planes in process,

and flags (2^{-1} is set to 0):

- $aliveN_{i,j}$:

$$aliveN_{i,j} = \begin{cases} \text{true} & v_{i,j} \geq 2^n \\ \text{false} & \text{else}, \end{cases}$$

- $aliveD_{i,j}$:

$$aliveD_{i,j} = \begin{cases} \text{true} & v_{k,l} \geq 2^n \text{ for } \exists(k, l) \in \mathcal{D}(i, j) \\ \text{false} & \text{else}, \end{cases}$$

- $pAliveN_{i,j}$:

$$pAliveN_{i,j} = \begin{cases} \text{true} & v_{i,j} \geq 2^{n-1} \\ \text{false} & \text{else}, \end{cases}$$

- $pAliveD_{i,j}$:

$$pAliveD_{i,j} = \begin{cases} \text{true} & v_{k,l} \geq 2^{n-1} \text{ for } \exists(k, l) \in \mathcal{D}(i, j) \\ \text{false} & \text{else}, \end{cases}$$

- $nzN_{i,j}$:

$$nzN_{i,j} = \begin{cases} \text{true} & v_{i,j} \neq 0 \\ \text{false} & \text{else}, \end{cases}$$

- $nzD_{i,j}$:

$$nzD_{i,j} = \begin{cases} \text{true} & v_{k,l} \neq 0 \text{ for } \exists(k, l) \in \mathcal{D}(i, j) \\ \text{false} & \text{else}, \end{cases}$$

- $youAlive_{i,j}$: True if we can know $aliveN_{i,j}$ = true or $aliveD_{i,j}$ = true at its parent node and false if else,
- $youAliveD_{i,j}$: True if we can know there is an offspring node (k, l) such that $aliveN_{k,l}$ = true or $aliveD_{k,l}$ = true at node (i, j) and false if else (That is $aliveD_{k,l}$ = true and for only one of offspring (k, l), $pAliveN_{k,l}$ = true or $pAliveD_{k,l}$ = true).

Several simulations during coder refinements were done and the proposed algorithm of zpSPIHT are:

1. Calculate $|f_{i,j}/q|$ for each node:

$$v_{i,j} = \begin{cases} 0 & |f_{i,j}/q| \leq z \\ \lfloor |f_{i,j}/q| + 0.5 \rfloor & \text{else}, \end{cases}$$

   where $\lfloor x \rfloor$ is the floor function of x,
2. $n = 0$,
3. Execute NodeEnc$(i, j, 0)$ for all $(i, j) \in \mathcal{H}$,
4. Stop encoding if $aliveN_{i,j}$ = false and $aliveD_{i,j}$ = false for all $(i, j) \in \mathcal{H}$,
5. $n = n + 1$,
6. Go to Step 3,

where the algorithm of NodeEnc$(i, j, youAlive_{i,j})$ is as follows:

1. Return if $pAliveN_{i,j}$ = false and $pAliveD_{i,j}$ = false,

(Processes to cut value of node (i, j))

2. When $pAliveN_{i,j}$ = true and $aliveN_{i,j}$ = true,
   a. Output 1 if not $youAlive_{i,j}$ = true and $pAliveD_{i,j}$ = false,
   b. and

$$Output \begin{cases} \text{sign } f_{i,j} & n = 0 \\ \text{residual of } v_{i,j}/2^{n-1} \text{ divided by 2} & n > 1, \end{cases}$$

3. Output 0 if $pAliveN_{i,j}$ = true and $aliveN_{i,j}$ = false,

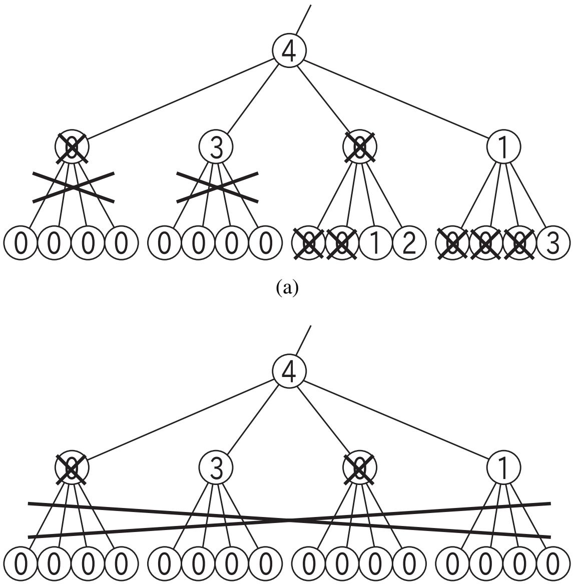
(Processes to cut the descendants of node (i, j))

4. Output 1 if $pAliveD_{i,j}$ = true and $aliveD_{i,j}$ = true; and not $youAlive_{i,j}$ = true and $aliveN_{i,j}$ = false,
5. Output 0 if $pAliveN_{i,j}$ = true and $aliveD_{i,j}$ = false,

(Processes to transfer the offspring of node (i, j) as the object to execute NodeEnc)

6. If the number of nodes in $\mathcal{O}(i, j)$ such that $pAliveN_{k,l}$ = true or $pAliveD_{k,l}$ = true is one, $youAliveD_{i,j}$ is true, else false,
7. Execute NodeEnc$(k, l, youAliveD_{i,j})$ for all $(k, l) \in \mathcal{O}(i, j)$,
8. Return.

Figure 3 (a) and (b) illustrate the examples of pruning of quantized zeros and number (0, 1, 2, 4) in the figures represent their transform coefficients. In Fig. 3 (a), the descendants of the leftest and the second leftest nodes at the second layer are pruned at the stage $n = 0$ since all of them are zeros. The other zeros are also pruned individually. Since at the stage $n = 0$ three children of the rightest node at the second layer have been pruned, we can know the remained child is alive at the stage $n = 1$ from the information that a descendant of the node at the second layer is alive at the stage $n = 1$. Then, we can omit to output the information whether the remained child is alive or not. The flag $youAlive_{i,j}$ or $youAliveD_{i,j}$ is used for the purpose.



(a)

(b)

Fig. 3 Pruning of quantized zeros.

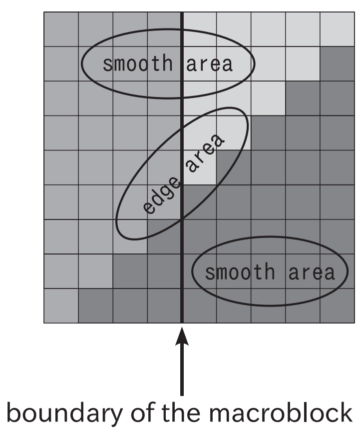


Fig. 4 Concept of deblocking filter.

We tried to prune $\mathcal{L}(i, j) = \mathcal{D}(i, j) - \mathcal{O}(i, j)$ that corresponds to type B node of list of insignificant sets (LIS) in SPIHT in Fig. 3 (b) and to prune (i, j) and $\mathcal{D}(i, j)$ simultaneously. However, they did not improve the coding efficiency. The computational complexity of zpSPIHT and SPIHT are similar since the scanned node are not changed.

2.3 Edge Adaptive Deblocking Filter

We deploy the edge adaptive deblocking filter to the predicted picture which is after the motion compensated predictions, and not after the inverse transform as has been widely used in many studies [26], [27] since LT does not produce blocking distortion. The concept of the edge adaptive deblocking filter is shown in Fig. 4. In the figure, we assume a vertical boundary of the macroblock on the center. Therefore, the area is divided into a smooth area in the upper and the lower parts and an inclined edge in the center area. The deblocking filter could detect the area which has edges around the macroblock boundaries. In such case, at first, deblocking filter works in a smooth area. Then, the deblocking filter detects the direction of the edges and the pixels near the edges are filtered by only the orthogonal direction to the edges. The filter can detect horizontal, vertical, and inclined (45° or −45°) edges at the macroblock boundaries. The algorithm and details of the edge adaptive deblocking filter is in [23]. In contrast, if we deploy OBMC instead of the edge adaptive deblocking filter, the area of blocks for motion compensation will be increased four times that also increased the computational complexity. Furthermore, it is possible to apply many advanced motion compensation techniques to this proposed method easily.

We evaluated the computational complexity of the proposed edge adaptive deblocking filter compared to the H.264 deblocking filter [28]. The calculations for each deblocking filter are:

- Edge adaptive deblocking filter: $\left(\frac{WH}{8} - W - H\right)\times$(4 times bit shift, 40 times addition & subtraction and 2 times clipping process)
- The H.264 deblocking filter ($bS = 1$): $\left(\frac{WH}{8} - W - H\right)\times$(6 times bit shift, 12 times addition & subtraction and 5 times clipping process)
- The H.264 deblocking filter ($bS = 4$): $\left(\frac{WH}{8} - W - H\right)\times$(16 times bit shift, 32 times addition & subtraction)

where $W = 720$ and $H = 480$ are the width and height for the SDTV, respectively. We found that the proposed method has slightly higher computational cost than the H.264 deblocking filter ($bS = 1$). However, since it is comparable to the H.264 deblocking filter ($bS = 4$), it can be realized practically.

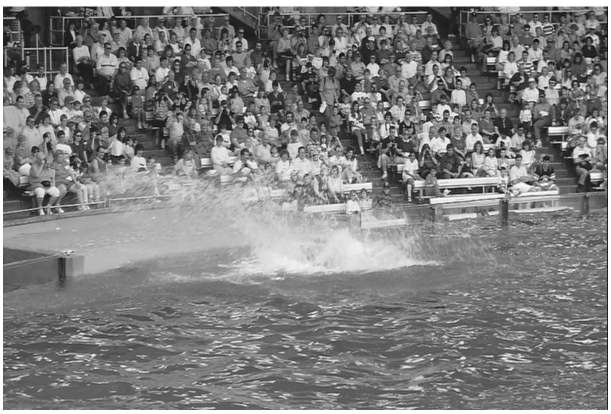
3. Experimental Results

We conducted some coding experiments to verify the performance of the LT. We used three ITE standard pictures in video sequences as test samples. The characteristic of each standard picture is presented in Table 1. Figure 5 shows the first original picture out of 10 pictures in sequences for respective standard picture. Only luminance component in the standard picture was used. The first picture was encoded as I-picture and the other nine pictures were encoded as P-pictures. The quality-of-picture degradation of a decoding

Table 1 Characteristic of the standard pictures.

Standard pictures	Characteristics
Intersection	Vehicles moving at the junction
Whale show	Jumping whale and audience during show
Rotating disk	Rotating disk with still picture attached to it

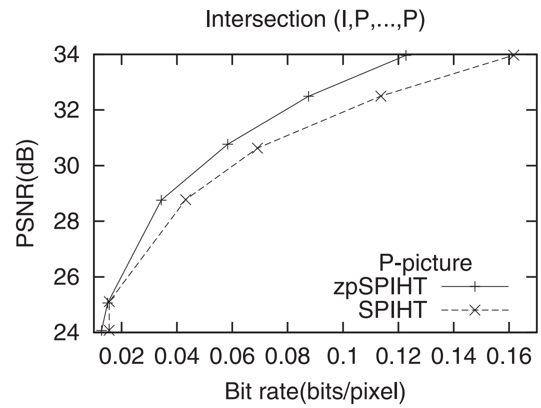
(a) *Intersection*



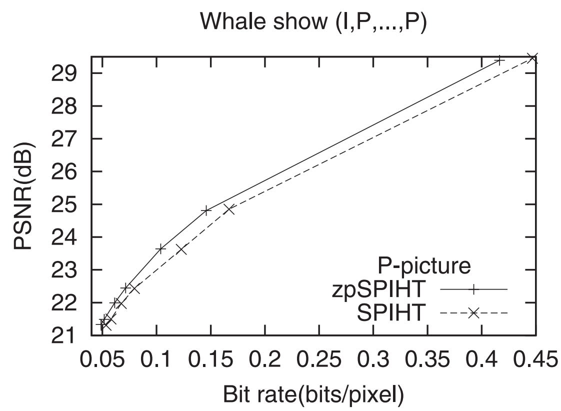
(b) *Whale show*

(c) *Rotating disk*

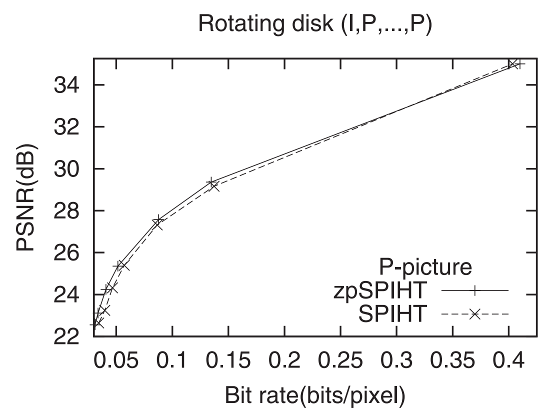
Fig. 5 Standard pictures.



(a) *Intersection*



(b) *Whale show*



(c) *Rotating disk*

Fig. 6 Rate-distortion curves (P-picture) with zpSPIHT and SPIHT.

picture was evaluated numerically. Peak-to-peak Signal-to-Noise Ratio (PSNR) is used for numerical assessment. PSNR is given by the following formulas:

$$\mathrm{PSNR(dB)} = 20\log_{10}\frac{255}{\sqrt{\mathrm{MSE}}}, \tag{1}$$

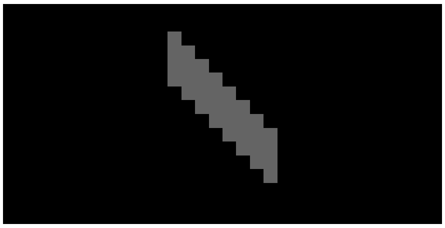
and

$$\mathrm{MSE} = \frac{1}{WH}\sum_{x=1}^{W}\sum_{y=1}^{H}\left|f(x,y) - f'(x,y)\right|^2, \tag{2}$$

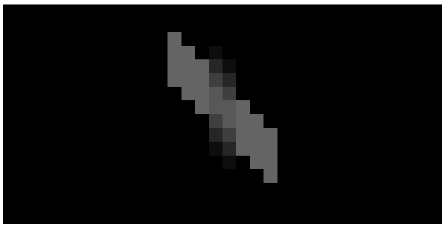
where $f(x,y)$ and $f'(x,y)$ are the pixel values of the original and the decoded pictures. In this experiments the size of the standard pictures is $W = 720$ and $H = 480$.

3.1 Comparison between zpSPIHT and SPIHT

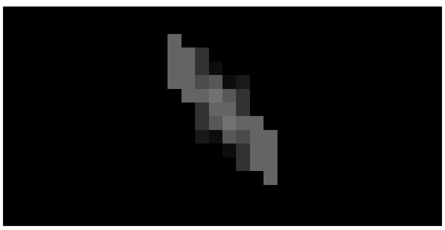
Coding and decoding of the difference pictures were performed by zpSPIHT and SPIHT. Figure 6 shows the PSNR for the decoded P-picture against bit rate (bpp) to compare the performance of zpSPIHT and SPIHT for coding. The value of bitrate of zpSPIHT was lower than SPIHT when



(a) Original



(b) Proposed method deblocking filter



(c) H.264 deblocking filter

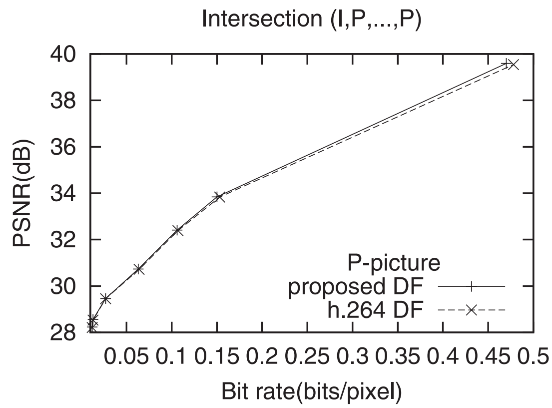
Fig. 7 Comparison of deblocking filters.

PSNR values were fixed. It means that zpSPIHT performs better than SPIHT for all standard pictures. Especially for *Intersection*, the zpSPIHT achieved efficient coding since the difference picture contains many quantized zeros.

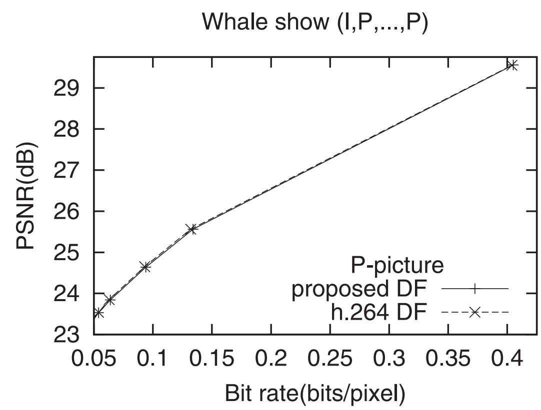
3.2 Comparison between Edge Adaptive Deblocking Filter and H.264 Deblocking Filter

We compared the proposed edge adaptive deblocking filter to the H.264 deblocking filter [28]. Since H.264 deals with block based transform, a higher degree of deblocking filter is necessary. In this experiments, we apply the weakest H.264 deblocking filter ($bS = 1$) when the two motion vectors are different between adjacent macroblocks. First, we carried out experiments on a picture of 45° perpendicular line with both deblocking filters. Figure 7 (a) shows the original picture of 45° perpendicular line. Figure 7 (b) and (c) show its pictures filtered by the proposed edge adaptive deblocking filter and by the H.264 deblocking filter, respectively. From the results, we clearly can conclude that the perpendicular edges can be preserved nicely by the proposed edge adaptive deblocking filter.

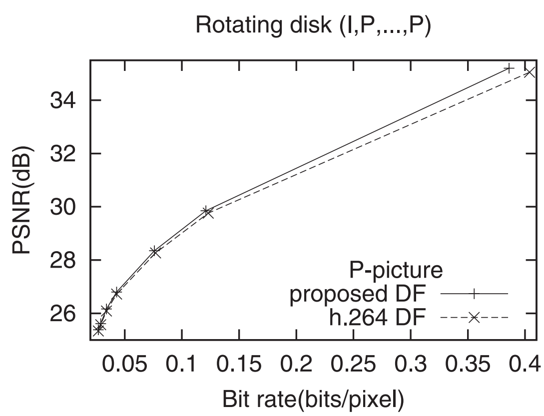
Figure 8 shows the PSNRs of the P-pictures decoded with the proposed edge adaptive deblocking filter or the H.264 deblocking filter using standard SPIHT coder. The PSNRs of the proposed method are slightly higher than those of the H.264 deblocking filter for all standard pictures. The proposed edge adaptive deblocking filter could produce



(a) *Intersection*



(b) *Whale show*



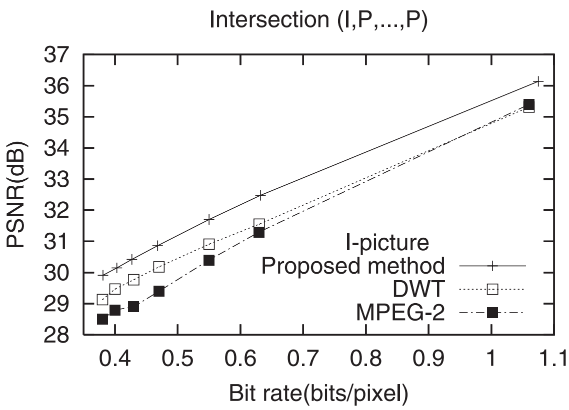
(c) *Rotating disk*

Fig. 8 Rate-distortion curves (P-picture) with different deblocking filters.

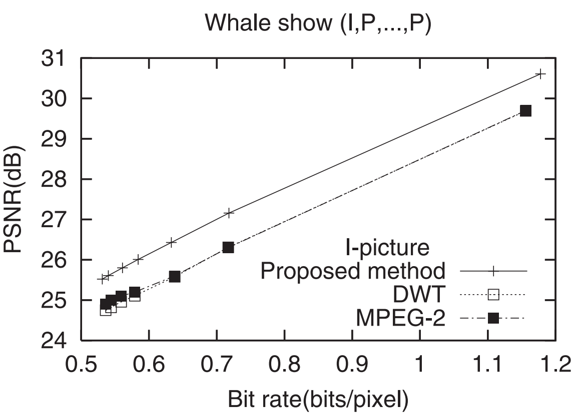
better quality pictures compared to H.264. This is because the H.264 deblocking filter ($bS = 1$) is still strong for this purpose.

3.3 Objective Performance

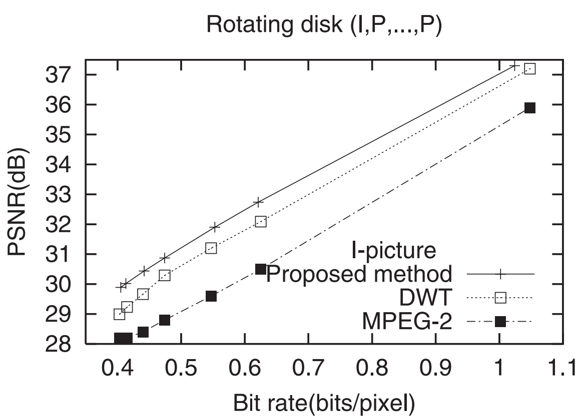
Figure 9 shows the PSNR for the decoded I-pictures against bit rate (bpp) to compare the performance of proposed method, DWT, and MPEG-2. DWT deployed the same coding scheme as LT and used the Daubechies 9/7 filter pair. The PSNR results of proposed method were the highest for



(a) *Intersection*

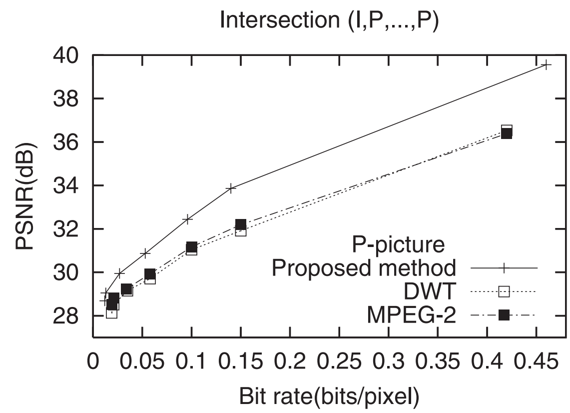


(b) *Whale show*

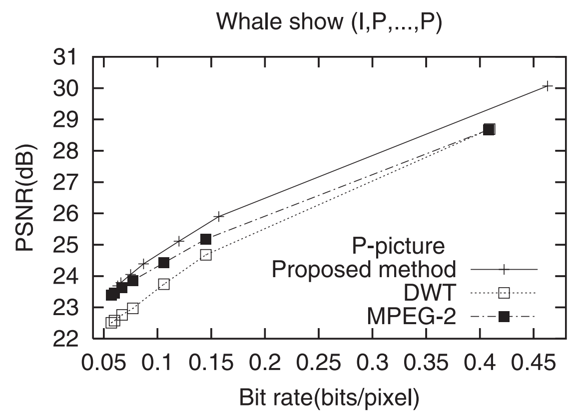


(c) *Rotating disk*

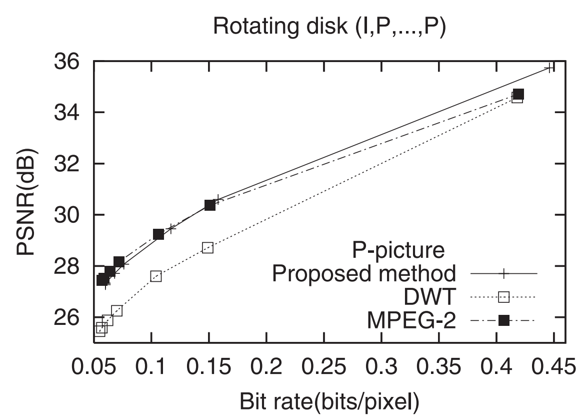
Fig. 9 Rate-distortion curves (I-picture) with different coders.



(a) *Intersection*



(b) *Whale show*



(c) *Rotating disk*

Fig. 10 Rate-distortion curves (P-picture) with different coders.

all standard pictures. The difference of around 1.0 dB in average from those of MPEG-2 was achieved. It was possible that in coding of I-picture, the proposed method did not produce the blocking distortions as usually occur in MPEG-2. Figure 10 shows the PSNRs of decoded P-picture of proposed method, DWT and MPEG-2. Again, the proposed method gained the best PSNRs compared to the MPEG-2 and the DWT for all standard pictures. In all conditions, the PSNR of proposed method was higher than those of DWT and MPEG-2. Thus, better picture quality could be expected from proposed method compared to DWT and MPEG-2.

3.4 Subjective Performance

We qualitatively investigated the presence of the blocking distortion. All of the standard pictures were decoded by their respective methods, and the quality of the pictures was compared. Figure 11 shows the fifth picture in time sequence of decoded P-picture of *Intersection*. The pictures at the bit rate of 0.012 bpp were magnified four times. Figure 11 (a) shows the original picture. By proposed method, the edges of the zebra crossing were pre-

(a) Original

(b) Proposed method

(c) DWT

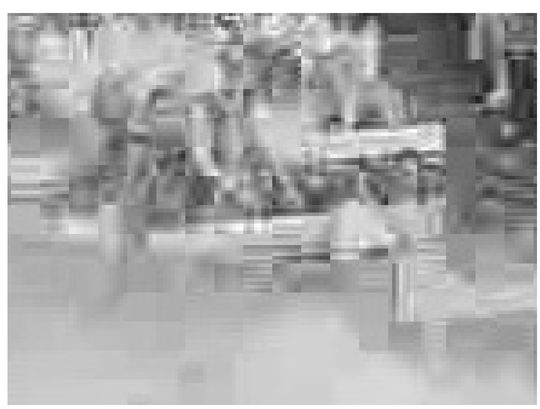
(d) MPEG-2

Fig. 11 *Intersection* (P-picture).

(a) Original

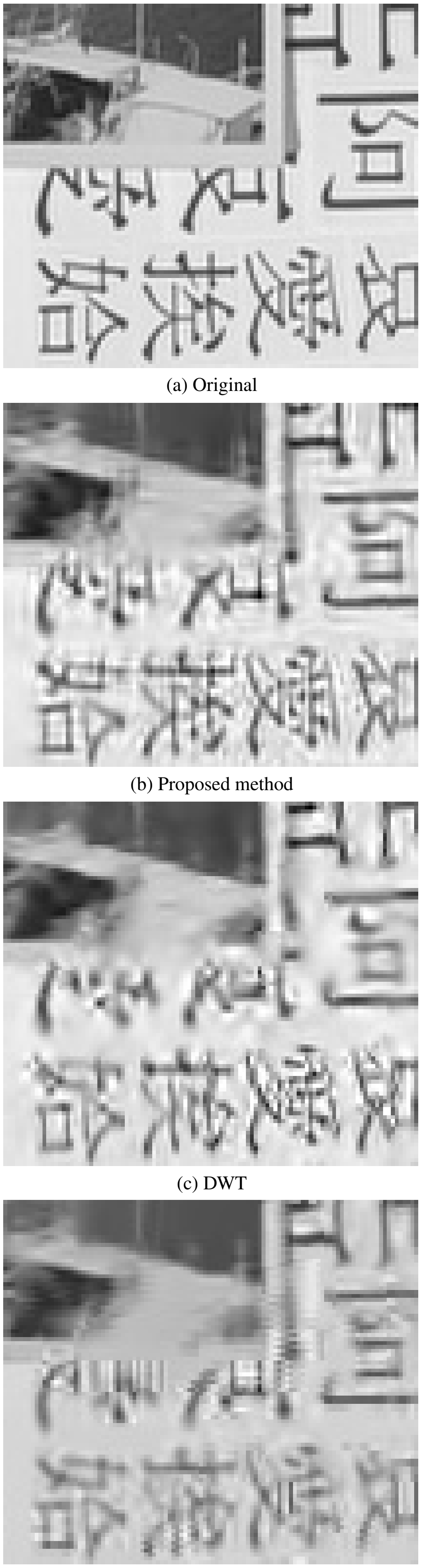
(b) Proposed method

(c) DWT



(d) MPEG-2

Fig. 12 *Whale show* (P-picture).



(a) Original

(b) Proposed method

(c) DWT

(d) MPEG-2

Fig. 13 *Rotating disk* (P-picture).

served and the blocking distortions were less as shown in Fig. 11 (b). Though the blocking distortions are less in Fig. 11 (c), the edges of the zebra crossing lines are not as sharp as in Fig. 11 (b). There are many blocking distortions could be noticed in Fig. 11 (d) coded by MPEG-2. The road lane in picture decoded by proposed method can be seen more clearly compared to the picture decoded by DWT and MPEG-2.

Figure 12 shows the fifth picture in time sequence of the decoded pictures of *Whale show* which was zoomed by four times. The bit rate was 0.063 bpp. Figure 12 (a) shows the original picture. The faces of the audience in Fig. 12 (c) could not be seen as clearly as Fig. 12 (b). The audience could be seen clearly without blocking distortions in picture decoded by proposed method as shown in Fig. 12 (b) compared to those by MPEG-2 in Fig. 12 (d). The details of the picture encoded by proposed method are clearly shown compared to those encoded by DWT and MPEG-2.

Figure 13 shows the fifth picture in time sequence of the decoded pictures of *Rotating disk* which was enlarged by four times at 0.060 bpp bit rate. Figure 13 (a) shows the original picture. In Fig. 13 (b), the edges of the photo frame and the kanji characters could be seen clearly and sharply without any blocking distortion. The blocking distortions were noticeable in Fig. 13 (c) and the edges of the photo frame were not sharp in Fig. 13 (d).

Overall, the P-picture decoded by the proposed method was less distorted than those decoded by the MPEG-2 and the DWT. The deblocking filter reduced the unnecessary high frequency elements included in the difference picture. In the picture decoded by MPEG-2, distortion was produced during coding of the I-picture and the distortion possibly propagated to P-picture. The better quality of P-pictures was produced by the proposed method and supported by the good results of the objective performance (PSNR value).

4. Conclusions

In this paper, a moving picture coding using the lapped transform with zpSPIHT and edge adaptive deblocking filter was proposed. The computer simulations using three standard pictures were conducted. zpSPIHT showed better performance than the conventional SPIHT. The proposed method exhibited the highest PSNR results in about 1.0 dB compared to the DWT and MPEG-2 especially in the low bit rate condition. The pleasant picture of a better quality was produced from coding by using the proposed method compared to DWT and MPEG-2. Thus, the proposed method is a reliable and a promising method for coding moving picture, especially at low bit rates.

References

[1] K.K. Kwon, S.H. Im, and D.S. Lim, "Deblocking algorithm in MPEG-4 Video coding using block boundary characteristics and adaptive filtering," Proc. 2005 IEEE International Conference on Image Processing (ICIP 2005), pp.541–544, Italy, Genoa, Sept. 2005.
[2] The JPEG-2000 Still Image Compression Standard, ISO/IEC JTC

/SC 29/WG 1 (ITU-T SG 16), Dec. 2005.
[3] B.E. Usevitch, "A tutorial on modern lossy wavelet image compression: Foundations of JPEG 2000," IEEE Signal Process. Mag., pp.22–35, Sept. 2001.
[4] M. Rabbani and R. Joshi, "An overview of the JPEG 2000 still image compression standard," Elsevier Signal Processing: Image Communication, vol.17, pp.3–48, 2002.
[5] D. Santa-Cruz, R. Grosbois, and T. Ebrahimi, "JPEG 2000 performance evaluation and assessment," Elsevier Signal Processing: Image Communication, vol.17, pp.113–130, 2002.
[6] C. Kwan, B. Li, R. Xu, X. Li, T. Tran, and T. Nguyen, "A complete image compression scheme based on overlapped block transform with post-processing," EURASIP Journal on Applied Signal Processing, vol.2006, pp.1–15, Jan. 2006.
[7] N. Zainal and Y. Yamashita, "Moving picture coding by subband transform and edge adaptive deblocking filter," Proc. 2009 RISP International Workshop on Nonlinear Circuits and Signal Processing (NCSP 09), pp.451–454, Hawaii, USA, March 2009.
[8] H.S. Malvar, "Lapped transforms for efficient transform/subband coding," IEEE Trans. Acoust. Speech Signal Process., vol.38, no.6, pp.969–978, June 1990.
[9] A.S. Lewis and G. Knowles, "Image compression using the 2-D wavelet transform," IEEE Trans. Image Process., vol.1, no.2, pp.244–250, April 1992.
[10] M. Ohta and S. Nogaki, "Hybrid picture coding with wavelet transform and overlapped motion-compensated interframe prediction coding," IEEE Trans. Signal Process., vol.41, no.12, pp.3416–3424, Dec. 1993.
[11] T. Nagai, M. Ikehara, M. Kaneko, and A. Kurematsu, "Generalized unequal length lapped orthogonal transform for subband image coding," IEEE Trans. Signal Process., vol.48, no.12, pp.3365–3378, Dec. 2000.
[12] Y. Tanaka, M. Ikehara, and T.Q. Nguyen, "Unequal length first-order linear-phase filter banks for efficient image coding," Proc. 2007 IEEE International Conference on Image Processing (ICIP 2007), pp.173–176, San Antonio, USA, Sept. 2007.
[13] H.S. Malvar, "Extended lapped transform: Properties, applications, and fast algorithms," IEEE Trans. Signal Process., vol.40, no.11, pp.2703–2714, Nov. 1992.
[14] H.S. Malvar and D.H. Staelin, "The LOT: Transform coding without blocking effects," IEEE Trans. Acoust. Speech Signal Process., vol.37, no.4, pp.553–559, April 1989.
[15] H.S. Malvar, "Lapped biorthogonal transforms for transform coding with reduced blocking and ringing artifacts," Proc. 1997 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 1997), April 1997.
[16] Y.C. Li, T.H. Wu, and Y.C. Chen, "A scene adaptive hybrid video coding scheme based on the LOT," IEEE Trans. Circuits Syst. Video Technol., vol.8, no.1, pp.92–102, Feb. 1998.
[17] B. Deknuydt, L. Van Eycken, and A. Oosterlinck, "A deblocking scheme suitable for use inside the codec loop," Proc. 7th Mediterranean Electrotechnical Conference, vol.1, pp.9–12, April 1994.
[18] L. Hsu and K.R. Rao, "Wavelet and lapped orthogonal transforms with overlapped motion-compensation for multiresolution coding of HDTV," IEEE Trans. Circuits Syst. II: Analog Digit. Signal Process., vol.45, no.8, pp.1002–1014, Aug. 1998.
[19] J.M. Shapiro, "Embedded image coding using zerotrees of wavelets coefficients," IEEE Trans. Signal Process., vol.41, no.12, pp.3445–3462, Dec. 1993.
[20] T. Uto, M. Ikehara, and M. Okuda, "Wavelet packet algorithm for quadtree-based embedded image coding," Proc. 2002 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2002), pp.3517–3520, May 2002.
[21] A. Said and W.A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," IEEE Trans. Circuits Syst. Video Technol., vol.6, no.3, pp.243–250, June 1996.
[22] U. Reimers, Digital Video Broadcasting, Springer, 2001.
[23] N. Zainal, T. Ohta, and Y. Yamashita, "Moving picture coding by wavelet transform and edge adaptive deblocking filter," Proc. 2007 International conference on Image Processing, Computer Vision, & Pattern Recognition, pp.320–326, June 2007.
[24] K.K. Pang and T.K. Tan, "Optimum loop filter in hybrid coders," IEEE Trans. Circuits Syst. Video Technol., vol.4, no.2, pp.158–167, April 1994.
[25] K. Martin, R. Lukac, and K.N. Plataniotis, "SPIHT-based coding of the shape and texture of arbitrarily shaped visual objects," IEEE Trans. Circuits Syst. Video Technol., vol.16, no.10, pp.1196–1208, Oct. 2006.
[26] M. Sima, Y. Zhou, and W. Zhang, "An efficient architecture for adaptive deblocking filter of H.264/AVC video coding," IEEE Trans. Consum. Electron., vol.50, no.1, pp.292–296, Feb. 2004.
[27] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz, "Adaptive deblocking filter," IEEE Trans. Circuit Syst. Video Technol., vol.13, no.7, pp.614–619, July 2003.
[28] "H.264 Series H: Audiovisual and multimedia systems," Telecommunication Standardization Sector of ITU, International Telecommunication Union, Nov. 2007.

Nasharuddin Zainal was born in Kuala Lumpur, Malaysia, on September 12, 1974. He received the B.E. degree from Tokyo Institute of Technology in 1998, and M.E. degree from Malaysian National University in 2003 and currently working toward the Ph.D. degree at Tokyo Institute of Technology.

Toshihisa Tanaka was born in Nagano, Japan, on 1974. He received the B.E., the M.E., and the Ph.D. degrees from Tokyo Institute of Technology in 1997, 2000, and 2002 respectively. In Apr. 2004, he joined Department of Electrical and Electronic Engineering, Tokyo University of Agriculture and Technology, and he is currently an Associate Professor in the same university.

Yukihiko Yamashita was born in Tokyo, Japan, on November 21, 1960. He received the B.E., the M.E., and the Ph.D. degrees from Tokyo Institute of Technology. He joined Department of International Development Engineering, Tokyo Institute of Technology Engineering, and he is currently an Associate Professor in the same university.