PAPER Special Section on Information Theory and Its Applications

Parallel Processing of Distributed Video Coding to Reduce Decoding Time

Yoshihide TONOMURA[†a)], ***Student Member***, **Takayuki NAKACHI**[†], **Tatsuya FUJII**[†], ***Members***, ***and*** **Hitoshi KIYA**[††], ***Fellow***

SUMMARY This paper proposes a parallelized DVC framework that treats each bitplane independently to reduce the decoding time. Unfortunately, simple parallelization generates inaccurate bit probabilities because additional side information is not available for the decoding of subsequent bitplanes, which degrades encoding efficiency. Our solution is an effective estimation method that can calculate the bit probability as accurately as possible by index assignment without recourse to side information. Moreover, we improve the coding performance of Rate-Adaptive LDPC (RA-LDPC), which is used in the parallelized DVC framework. This proposal selects a fitting sparse matrix for each bitplane according to the syndrome rate estimation results at the encoder side. Simulations show that our parallelization method reduces the decoding time by up to 35[%] and achieves a bit rate reduction of about 10[%].

key words: *distributed video coding, LDPC codes, parallelization system, bitplane correlation, Gray code*

1. Introduction

High-performance image compression has recently become extremely important since the number and size of digital image data files continue to grow. One response, the international standard H.264/AVC [1], achieves the compression rate is twice as much as MPEG-2 [2]. A weakness of H.264/AVC is that the encoder needs a high level of computing power in order to support MC (Motion Compensation) and optimization of the macro block size. A new trend is the emergence of applications for low-complexity encoders. Examples of such applications include mobile camera phones and sensor network cameras. One approach to implementing these encoders is Distributed Video Coding (DVC) [3], [4].

One advantage of DVC systems is that they transfer the computation overhead, which is significant, to the decoder. One example is intraframe video coding with interframe decoding [5], [6]. In these systems, the encoder does not need to support MC because the decoder does. This means that the encoder is less complex. Another characteristic of DVC systems is robustness. The encoder does not handle motion vectors since it does not support MC, which means that DVC systems are drift-free.

For the above reasons, DVC is attracting researchers' attention as a new paradigm for video compression. However, the DVC decoder can be an order of magnitude more complex than a conventional video encoder such as H.264/AVC [7]. According to the DISCOVER project [8], the DVC decoder requires 20–60 times much more computation time than the H.264/AVC encoder. One reason is that the DVC decoder uses channel coding with a message-passing algorithm. The use of this algorithm means that DVC systems have excessive decoding times. In other words, to realize practical systems, we need a DVC framework that offers reasonable decoding times.

This paper proposes a parallel processing DVC framework that greatly reduces the decoding time. The recent development of good error correcting codes such as LDPC codes [9], [10] has boosted the feasibility of parallel processing; the remaining difficulty is bitplane processing. We, therefore, propose a parallelized DVC framework that treats each bitplane independently to overcome the remaining barrier to parallel processing. The simple application of parallelization generates inaccurate bit probabilities since the parallelized framework can't use additional side information when decoding subsequent bitplanes and so suffers from low encoding efficiency. Our solution is an information index assignment method that can estimate the bit probability as accurately as possible without any additional side information.

Note that [11]–[16] proposed index assignment methods that provide more accurate bit probabilities for DVC framework. However, the method in [11] requires knowledge of bitplane correlation and the method in [16] introduces a Gray code [17] for joint bitplane coder, so these methods can't be used in the parallelized DVC framework. Papers [12]–[15] also introduce a Gray code for the DVC framework, but these methods simply change the Euclidean distances between the source and side information coefficients. In other words, these methods fail to consider the "error" distribution. On the other hand, we propose 3 new contributions in this paper. The first is we propose a bit probability estimation method for the parallelized DVC framework using a Gray code that considers the error distribution. A part of its contents was discussed in Ref. [18]. The second is we give theoretically grounds for the proposed method. The third is we improve the coding performance uses a fitting sparse matrix for each bitplane according to the syndrome rate estimation results.

Manuscript received January 20, 2009.
Manuscript revised April 25, 2009.

[†]The authors are with NTT Network Innovation Laboratories, NTT Corporation, Yokosuka-shi, 239-0847 Japan.
[††]The author is with the Department of Information and Communication Systems Engineering, Faculty of System Design, Tokyo Metropolitan University, Hino-shi, 191-0065 Japan.
a) E-mail: tonomura.yoshihide@lab.ntt.co.jp

DOI: 10.1587/transfun.E92.A.2463

This paper is organized as follows. Section 2 introduces the pixel domain DVC architecture. Section 3 introduce the parallelized DVC framework and describes methods for improving the coding performance of the DVC framework. Section 4 evaluates the performance of the proposed DVC framework. Section 5 concludes this work.

2. Basic Distributed Video Coding Framework

In this paper, we consider the basic DVC framework called the pixel domain Wyner-Ziv codec [6] and we treat the case of GOP (group of pictures) = 2 for ease of explanation without any loss of generality.

2.1 Pixel Domain Wyner-Ziv Codec

Figure 1 illustrates a pixel domain Wyner-Ziv codec [6]; note that we replace Turbo codes with RA-LDPC codes because that latter are more suitable for parallel processing than the former and offer better coding performance [19].

In the architecture considered, every odd-frame of the video sequence is mapped as a key frame $X_{2i\pm1}$ which is compressed losslessly; these frames are assumed to be available to the decoder. Every even-frame of the video sequence is mapped as a Wyner-Ziv frame X_{2n}. These frames are compressed by a scalar quantizer and the Slepian-Wolf codec based on LDPC codes. First, Wyner-Ziv frames are quantized by a uniform scalar quantizer with 2^M, and each Wyner-Ziv frame is divided into $(B-M)$ bitplanes (where B is the total number of image bitplanes). These bitplanes are fed one by one into the RA-LDPC encoder [19]. In the RA-LDPC encoder, a sequence of input bits, $\mathcal{K}^l$, is mapped into the corresponding C^l syndrome bits. The encoder sends syndrome bits until the decoder can correctly decode the sequence.

At the decoder side, for each Wyner-Ziv frame, the decoder generates side information Y_{2i} from the key frames $X_{2i\pm1}$ by motion compensated interpolation. A block of side information $Y_{2i,b}$ is computed by

$$Y_{2i,b} = \frac{1}{2}\left(X_{2i-1,b'}\{MV/2\} + X_{2i+1,b'}\{MV/2\}\right) \tag{1}$$

where MV is the estimated motion vector between key frames $X_{2i\pm1}$. For a more efficient approach to generating the side information, see Ref. [21].

Next, we compute bit probabilities using the "*error*" correlation model between the side information Y_{2i} and the original Wyner-Ziv frame X_{2i}. The residual between Wyner-Ziv frames and side information is modeled as a Laplacian distribution. Details of the bit probability computation method are described in the next subsection. The rate-adaptive LDPC decoder recovers the Wyner-Ziv frames using the bit probabilities and received syndrome bits.

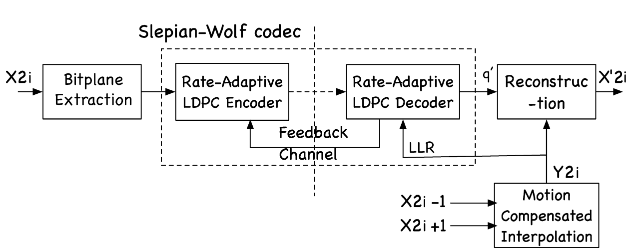


Fig. 1 Schematic diagram of pixel domain Wyner-Ziv codec.

Finally, the decoded symbol is reconstructed using the side information. The reconstruction operation for each pixel is given by

$$X'_{2i,n} = E(X_{2i,n}|q'_{2i,n}, Y_{2i,n}). \tag{2}$$

The operation of reconstruction falls into 2 cases. Case 1: if the side information $Y_{2i,n}$ is within decoded quantized bin $q'_{2i,n}$, the reconstructed pixel value $X'_{2i,n}$ is made equal to the side information. Case 2: if the side information $Y_{2i,n}$ is outside bin $q'_{2i,n}$, the reconstructed pixel value takes the high (or low) boundary of the bin's closest value.

2.2 Estimating Bit Probability of Basic DVC Framework

In the Wyner-Ziv codec, the decoder needs to accurately estimate the bit probabilities using "error" distribution and side information. The "error" distribution between Wyner-Ziv frames and side information is modeled as a Laplacian distribution and the conditional probability of the original Wyner-Ziv frame pixel$X_{2i,n}$ being equal to x is evaluated as

$$Pr\{X_{2i,n} = x|Y_{2i,n}\} = \frac{\alpha}{2}e^{-\alpha|x-Y_{2i,n}|} \tag{3}$$

where α is the Laplacian distribution parameter, which can be estimated at the decoder.

Next, bit conditional probability for each bitplane is computed. Assume that $X^l_{2i,n}$ is the l-th bitplane value of $X_{2i,n}$ and Z^l is the set of x values whose l-th bit equals 1. The conditional probability of each bit $X^l_{2i,n}$ equaling 1 is evaluated as

$$\begin{aligned} &Pr\{X^l_{2i,n} = 1|X^{j-1}_{2i,n}, Y_{2i,n}\} \\ &= \frac{\displaystyle\sum_{x\in Z^l} Pr\{X_{2i,n} = x|X^{j-1}_{2i,n}, Y_{2i,n}\}}{1 - \displaystyle\sum_{x\notin\{0,1\ldots2^B-1\}} Pr\{X_{2i,n} = x|X^{j-1}_{2i,n}, Y_{2i,n}\}} \end{aligned} \tag{4}$$

where $X^{j-1}_{2i,n}$ denotes the previous (all those decoded) bitplanes of $X_{2i,n}$ and $\sum_{x\notin\{0,1\ldots2^B-1\}} Pr\{X_{2i,n} = x|Y_{2i,n}\}$ is the probability that x exceeds the image range. Z^l is given by

$$\begin{cases} Z^0 & = \{2^{B-1}, \ldots, 2^B - 1\} \\ Z^1 & = \{2^{B-2}, \ldots, 2^{B-1} - 1\}\ \underline{or}\ \{3\cdot 2^{B-2}, \ldots, 2^B - 1\} \\ \vdots & = \quad \vdots \end{cases}. \tag{5}$$

These ranges of each Z^l depend on decoded bitplanes $X^{j-1}_{2i,n}$.

The input to the RA-LDPC decoder is the log-likelihood ratio (LLR), which is defined by

$$L_n^l = \log \frac{Pr\{X_{2i,n}^l = 0 | X_{2i,n}^{j-1}, Y_{2i,n}\}}{Pr\{X_{2i,n}^l = 1 | X_{2i,n}^{j-1}, Y_{2i,n}\}} = \log \frac{1 - Pr\{X_{2i,n}^l = 1 | X_{2i,n}^{j-1}, Y_{2i,n}\}}{Pr\{X_{2i,n}^l = 1 | X_{2i,n}^{j-1}, Y_{2i,n}\}}. \quad (6)$$

These LLRs are fed one by one into the rate-adaptive LDPC decoder and the q-ary sources are decompressed. This multilevel coding method for q-ary sources is basically similar to the method in [22].

2.3 Syndrome Rate Estimation for Basic DVC Framework

To avoid feedback channel complexity, syndrome rate estimation methods are proposed in [14], [20], [23]. In this paper, we amend the work in Refs. [20], [23], the mismatch probability of each bitplane between $X_{2i,n}^l$ and $Y_{2i,n}^l$ is computed from

$$P_e^l = \frac{1}{N} \sum_n Pr\{X_{2i,n}^l \neq Y_{2i,n}^l | Y_{2i,n}, X_{2i,n}^{j-1}\} \quad (7)$$

$$= \frac{1}{N} \sum_n Pr\{X_{2i,n}^l = 1 | Y_{2i,n}^l = 0, Y_{2i,n}, X_{2i,n}^{j-1}\} + \frac{1}{N} \sum_n Pr\{X_{2i,n}^l = 0 | Y_{2i,n}^l = 1, Y_{2i,n}, X_{2i,n}^{j-1}\}. \quad (8)$$

Moreover, the l-th bitplane's syndrome rate is estimated as follows

$$C^l \approx P_e^l \times \log_2 \left[\frac{1}{P_e^l} \right] + [1 - P_e^l] \times \log_2 \left[\frac{1}{1 - P_e^l} \right]. \quad (9)$$

The estimated rate given by Eq. (7) and Eq. (9) would appear to be completely consistent with the channel capacity but this is not so in reality. One reason is the estimated rate is the entropy of the crossover-like probability assuming a binary symmetric channel (BSC). Thus, the estimated rate has an inherent level of error.

3. Parallelized Distributed Video Coding Framework

The basic DVC framework fails to treat each bitplane independently, which hinders parallelization of the RA-LDPC codes. In this section, we introduce a parallelized DVC framework that computes each bit probability independently. Moreover, we propose coding performance enhancement methods for the parallelized DVC framework.

3.1 Estimating Bit Probabilities in Parallelized DVC Framework

In this subsection, we describe two methods for estimating the bit probabilities in the parallelized DVC framework.

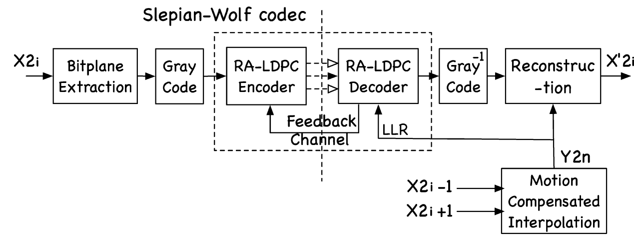


Fig. 2 Schematic diagram of Parallel Processing Framework for Wyner-Ziv codec.

In the parallelized DVC framework, the bitplanes are processed independently; we can not use the estimation method described in Sect. 2.2 because the decoded bitplanes $X_{2i,n}^{j-1}$ are not available at the decoder side. First, we extend the estimation method described above to suit the parallelized DVC, we call this the "simple method." Next, we propose a more effective estimation method with index assignment called "Gray Code" for the parallelized DVC.

3.1.1 Simple Method

Simply, the probabilities are estimated without using information about any other bitplane $X_{2i,n}^{j-1}$, as

$$Pr\{X_{2i,n}^l = 1 | Y_{2i,n}\} = \frac{\sum_{x \in Z^l} Pr\{X_{2i,n} = x | Y_{2i,n}\}}{1 - \sum_{x \notin \{0,1...2^B-1\}} Pr\{X_{2i,n} = x | Y_{2i,n}\}}, \quad (10)$$

where each Z^l is given by

$$\begin{cases} Z^0 &= \{2^{B-1}, ..., 2^B - 1\} \\ Z^1 &= \{2^{B-2}, ..., 2^{B-1} - 1\} \ \underline{and} \ \{3 \cdot 2^{B-2}, ..., 2^B - 1\} \\ \vdots &= \quad \vdots \end{cases} . \quad (11)$$

Since these ranges of each Z^l do not depend on decoded bitplanes $X_{2i,n}^{j-1}$, Eq. (10) supports parallel decoding for each bitplane independently.

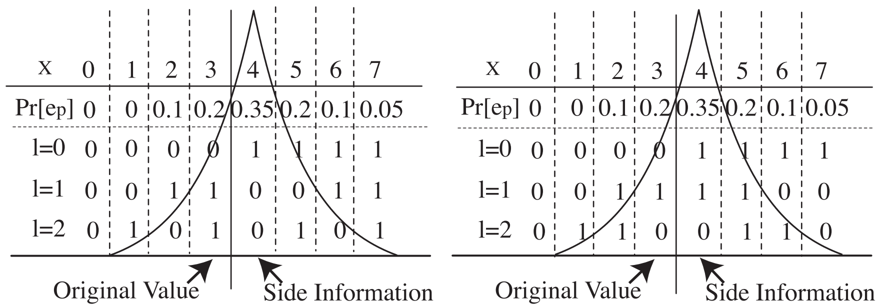
3.1.2 Proposed Method

The *simple method* can not provide accurate conditional probabilities because, from Eq. (10), the parallelized framework does not use decoded bitplane information $X_{2i,n}^{j-1}$. The difference between the $non-parallelized\ framework$ and the *simple method* is seen in the range of each Z^l. From Eq. (5), the range of each Z^l is limited to half that of the upper level bitplane. For the *simple method*, however, Eq. (11) shows that the range of each Z^l has no limit.

For the above reasons, the *simple method* generates inaccurate bit probabilities. To solve this problem, we introduce the index assignment method called the "Gray Code" [17] to change the range of each Z^l. Figure 2 illustrates the

Table 1 Example of error probabilities for 3-bits.

e_p	−4	−3	−2	−1	0	1	2	3
$Pr\{e_p\}$	0	0	0.1	0.2	0.35	0.2	0.1	0.05



(a) Binary code (b) Gray code

Fig. 3 Example of Binary code and Gray code for 3-bits.

Table 2 Log-likelihood ratio in each case (Original value).

l	0	1	2
L^l(Non-parallelized)	−0.9(0)	− inf(1)	−0.7(1)
L^l(Simple method)	−0.9(0)	0.2(1)	0.2(1)
L^l(Proposed method)	−0.9(0)	−1.7(1)	0.4(0)

proposed parallelized framework. In the proposed method, binary codewords X_{2i} are transformed into Gray codewords $\hat{X}_{2i}$ by

$$\hat{X}_{2i} = X_{2i} \oplus X_{2i} \gg 1. \tag{12}$$

$\oplus$ and $\gg$ represent eXclusive OR and bit shift operations, respectively, and binary codewords X_{2i} are given by

$$X_{2i} = [X^0_{2i}(MSB), X^1_{2i}, ..., X^B_{2i}(LSB)] \tag{13}$$

$$X^l_{2i} \in \{0, 1\}. \tag{14}$$

The conditional probabilities of Gray codewords $\hat{X^l}_{2i}$ are derived in the same way as Eq. (10) by replacing X with $\hat{X}$, where Z^l is given by

$$\begin{cases} Z^0 & = \{2^{B-1}, ..., 2^B - 1\} \\ Z^1 & = \{2^{B-2}, ..., 3 \cdot 2^{B-2} - 1\} \\ Z^2 & = \{2^{B-3}, ..., 3 \cdot 2^{B-3} - 1\}... \\ & \qquad \underline{and}\ \{5 \cdot 2^{B-3}, ..., 7 \cdot 2^{B-3} - 1\} \\ \vdots & = \quad \vdots \end{cases} . \tag{15}$$

These ranges of each Z^l fit the error model given by Eq. (3).

We consider the example of 3-bits ($B = 3$) to clarify why the bit exchange of "Gray code" yields accurate bit probabilities in the parallelized DVC framework. When the parameters are given as $X = 3$, $Y = 4$ and $Pr\{e_p\}$, as shown in Table 1, Fig. 3 illustrates the relationships. From Eqs. (4), (5) and substituting Eqs. (10), (11), (12) into Eq. (6), the LLR of each case is given in Table 2. From Eq. (6), if $L^l_n < 0$, the LDPC decoder assumes that $X^l_{2i} = 1$. From Table 2, we can verify that the proposed method yields a bit probability close to that provided by the *non − parallelized method*. If the *simple method* is used, the LDPC decoder is misled by all bits, and a lot of syndrome bits are required to correct these "errors."

Next, we generalize the previous example. The memoryless sources X is independent and identically distributed and the side information Y is given as the memoryless sources X to which is added random Laplacian additional noise given by Eq. (3). We evaluate each $H(P^l_e)$ using Eq. (7) and Eq. (16) as described in the next subsection. The result is shown in Fig. 4. As shown in Fig. 4, we confirm that our proposed method very closely approaches non-parallelized performance.

3.2 Syndrome Rate Estimation for Parallelized DVC Framework

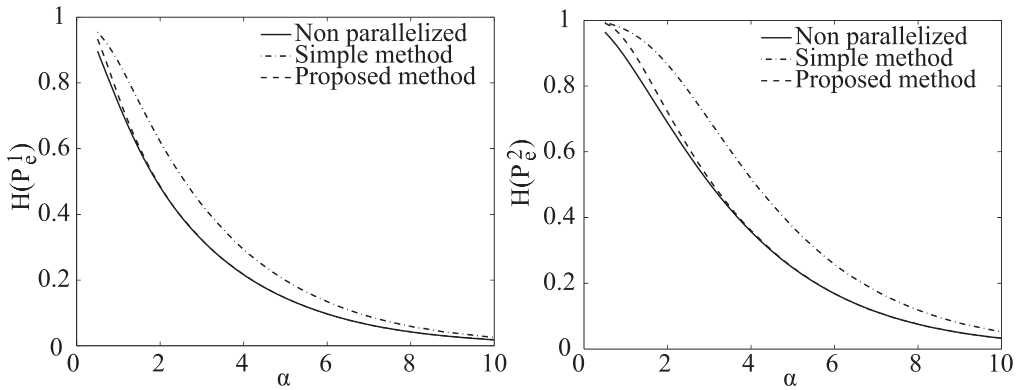
In this subsection, we introduce a syndrome rate estimation method to improve the coding performance of the parallelized DVC. The improvement in coding performance is described in the next subsection.

In the parallelized DVC system, syndrome rate estimation is easier to perform than is true with the non-parallelized DVC framework. The mismatch probability of each bitplane between $X^l_{2i,n}$ and $Y^l_{2i,n}$ is computed from

$$P^l_e = \frac{1}{N}\sum_n Pr\{X^l_{2i,n} \neq Y^l_{2i,n} | Y_{2i,n}\} \tag{16}$$

$$= \frac{1}{N}\sum_n Pr\{X^l_{2i,n} = 1 | Y^l_{2i,n} = 0, Y_{2i,n}\} + \frac{1}{N}\sum_n Pr\{X^l_{2i,n} = 0 | Y^l_{2i,n} = 1, Y_{2i,n}\}. \tag{17}$$

We estimate the l-th bitplane's syndrome rate in the parallelized framework by substituting Eq. (16) for Eq. (9).



(1) l=1 (2) l=2 (LSB)

Fig. 4 Theoretical relationship between $H(P_e)$ and α.

Table 3 Optimized LDPC degree distributions for BSC [25].

degree	Rate=0.15	Rate=0.50	Rate=0.85
λ_2	0.0903419	0.1527930	0.3151270
λ_3	0.1760760	0.2823500	0.1902840
λ_4		0.0062193	
λ_5			0.0449124
λ_7	0.3044350		0.1705930
λ_{16}	0.1356970		
λ_{17}	0.0127703		
λ_{18}			0.1405970
λ_{20}		0.5586370	
λ_{23}	0.0764734		
λ_{28}	0.1680910		
λ_{38}			0.0081261
λ_{42}			0.0440236
λ_{48}	0.0361156		
λ_{67}			0.0863369
Avgrd	40	10	4.5

3.3 Improving Coding Performance of Parallelized DVC Framework

In this subsection, we propose a coding performance enhancement method that uses rate estimation. In general, LDPC codes have good distributions for each target rate as do RA-LDPC codes. Table 3 shows optimized degree distributions for the BSC channel [25]. In the table, "Rate" represents the ratio of information bits to syndrome bits. Note that these degree distributions are not optimal for the RA-LDPC codes. The reason is that RA-LDPC codes, such as [19], are a class of Irregular Repeat-Accumulate (IRA) Codes [24]. Basically, the optimal sparse matrix of IRA codes is not the same as that of LDPC codes. However, we propose to use these degree distributions as a suboptimal way.

We evaluated the impact of LDPC degree distribution on coding performance. The experiment's model is shown in Fig. 5. The simulation considered a memoryless random binary signal, X, all sources have length of 6336 bits, and correlated signal Y follows i.i.d. BSC. At the encoder side, X is compressed using four RA-LDPC codes of differing

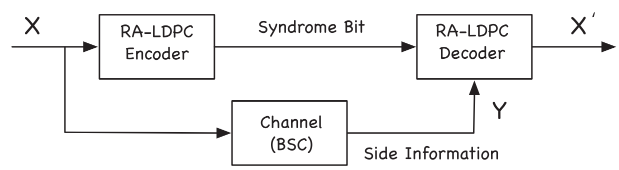


Fig. 5 Experiment's model for evaluation of compression with side information.

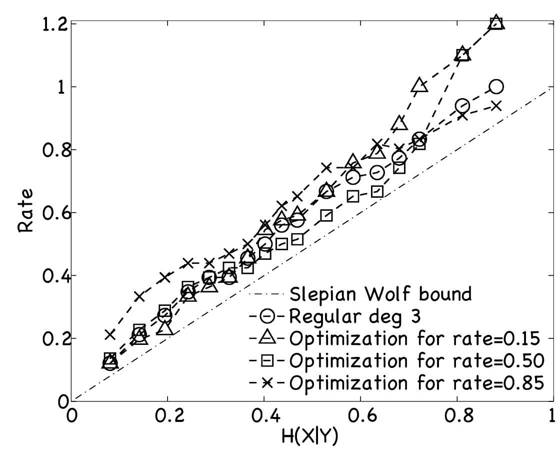


Fig. 6 Performance of RA-LDPC codes of lengths 6336 bits over i.i.d. BSC.

degree distribution, and X is recovered at the joint decoder using the reception syndrome and LLR (L_n). LLR is given by

$$L_n^l = \begin{cases} \log\dfrac{1-p}{p}, & Y_n = 0 \\ \log\dfrac{p}{1-p}, & Y_n = 1 \end{cases}. \tag{18}$$

We changed the crossover probability p in BSC and measured the syndrome bits required. The result is shown in Fig. 6.

As shown in Fig. 6, for the optimized rate of 0.15, the best performance is achieved at high compression ratios (range of 0.00 to 0.34), for the optimized rate of 0.50, the best performance is achieved at medium compression ratios (range of 0.34 to 0.72), and for the optimized rate of 0.85, the best performance is achieved at low compression ratios (range of 0.72 to 1.00).

Given the above simulation results, we propose to select the degree distribution according to the estimated syndrome rate. In our proposed method, if the estimated syndrome rate is small (high), we select a degree distribution that is optimum at low (high) syndrome rates. Our approach ensures the use of the near-optimal distribution and so improves coding performance if syndrome rate estimation is effective. We evaluate in the next section the extent to which coding performance can be improved if we apply this method in the DVC framework.

4. Simulation Results

This section discusses the simulations conducted to compare the performance of the proposed parallelized DVC framework to that of the existing non-parallelized DVC framework.

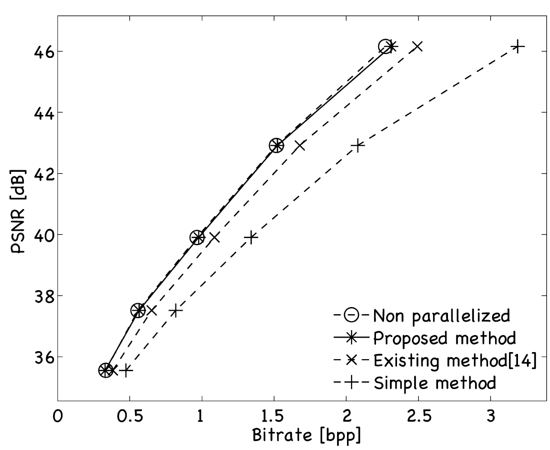


Fig. 7 Compression performance of parallel DVC framework (*Foreman sequence*, $\alpha = 0.24$).

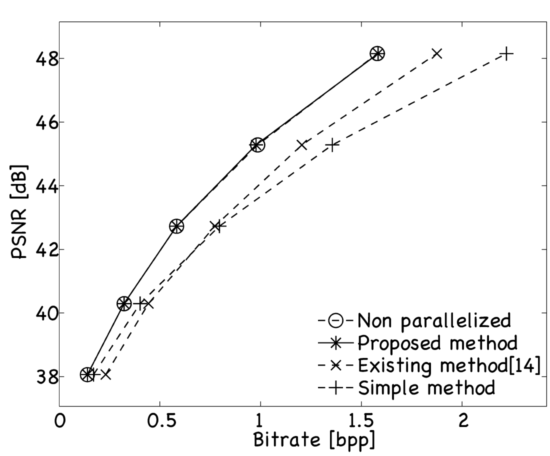


Fig. 8 Compression performance of parallel DVC framework (*News sequence*, $\alpha = 0.34$).

First, we evaluate coding performances of the parallelized and non-parallelized DVC frameworks. The results for 30 frames of the *Foreman* and *News* Gray-scale QCIF sequences are shown in Fig. 7 and Fig. 8. In this simulation, the Wyner-Ziv frames were divided into four tiles; each tile was compressed using the regular RA-LDPC ($deg = 3, k = 6336$) codec [19] and optimal α was computed using the original version of the Wyner-Ziv frame. Side information was made by the method of [21]. The four sequences had average α values of 0.24 and 0.34, respectively. Figure 7 and Fig. 8 show that the proposed method equals the compression performance of the non-parallelized DVC framework and yields about 3 [dB] better compression than the “simple method” at the same bitrate of 1.5 [bpp]. Our proposed method works well if α is large because the distribution of the“Gray code” is excellent with a Laplacian distribution.

Next, we measured the decoding time of each method for the *Foreman* sequence with PSNR values of 40 [dB] and 43 [dB] on a Mac Pro (2×3 GHz Dual Core Intel Xeon). The result, shown in Table 4, confirms that the parallelized DVC scheme reduces the decoding times by up to 35[%]. We also measured the encoding time of each method; all methods were run under parallel processing. The results are shown in Table 5. The encoding methods of each method are completely same. Therefore, the encoding times were the same. Note that the encoding/decoding time includes only RA-LDPC decoding time. However, the proposed parallelized DVC framework also reduces complexity. To calculate each LLR, look-up table (LUT) access is frequent; our proposed parallelized DVC framework, however, reduces the cost of making the LUT compared to the non-parallelized DVC framework. For example, in the case of 8-bits, the LUT needs only 2048 elements in our framework to calculate all LLR whereas 65280 elements are needed in the non-parallelized framework. This suggests that our proposal may reduce the overall decoding time, but this is an issue for a later paper.

We evaluated the estimation method proposed in 3.2 and the result, shown in Fig. 9 and Fig. 10, indicates that the difference between the estimated rate and the measured

Table 4 Average decoding time per frame.

Foreman sequence (PSNR = 40 [dB])	
Existing Method (non-parallelized)	1.55 [sec]
Simple Method (parallelized)	0.59 [sec]
Proposed method (parallelized)	0.45 [sec]
Foreman sequence (PSNR = 43 [dB])	
Existing Method (non-parallelized)	1.79 [sec]
Simple Method (parallelized)	0.69 [sec]
Proposed method (parallelized)	0.68 [sec]

Table 5 Average encoding time per frame.

Foreman sequence (all bitplanes)	
Existing Method (non-parallelized dec.)	12 [msec]
Simple Method (parallelized dec.)	12 [msec]
Proposed method (parallelized dec.)	12 [msec]

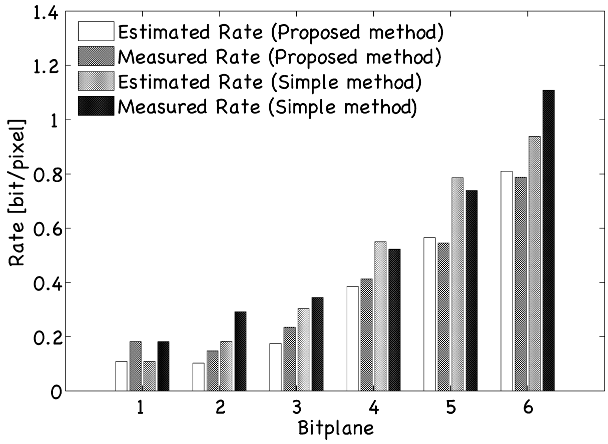


Fig. 9 Evaluation of proposed syndrome rate estimation method (*Foreman sequence*, $\alpha = 0.24$).

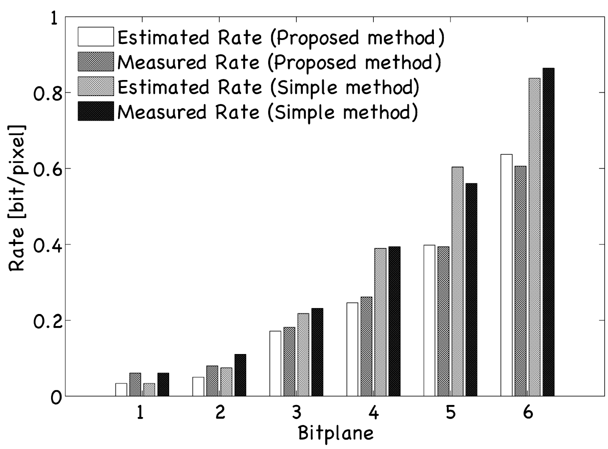


Fig. 10 Evaluation of proposed syndrome rate estimation method (*News sequence*, $\alpha = 0.34$).

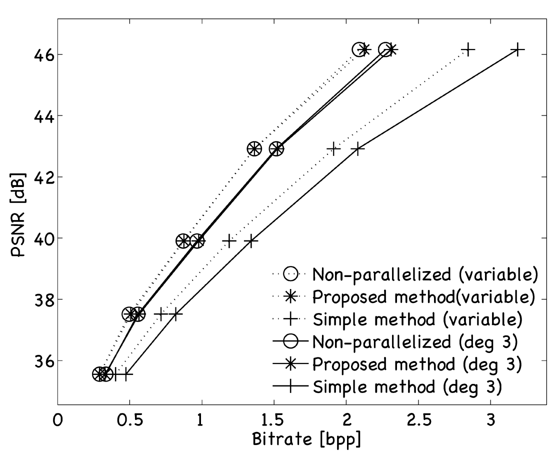


Fig. 11 Compression performance of parallel DVC framework (use of variable degree, *Foreman sequence*).

rate is small. The maximum difference was 10[%], which confirms the validity of our estimation method.

Finally, we evaluated the coding performance enhancement method proposed in 3.3. For this simulation, we used 3 distributions, see Table 3. Distribution selection proceeded as follows: if the estimated rate was between 0 and 0.34, we selected the optimized distribution for Rate = 0.15, if the estimated rate was between 0.34 and 0.72, we selected the optimized distribution for Rate = 0.50 and if the estimated rate was between 0.72 and 1.00, we selected the optimized distribution for Rate = 0.85. We plot the characteristics of the PSNR-Bitrate of Wyner-Ziv frames in Fig. 11 and Fig. 12. As shown, the proposed method reduces the bit rate by about 10[%] for all sequences.

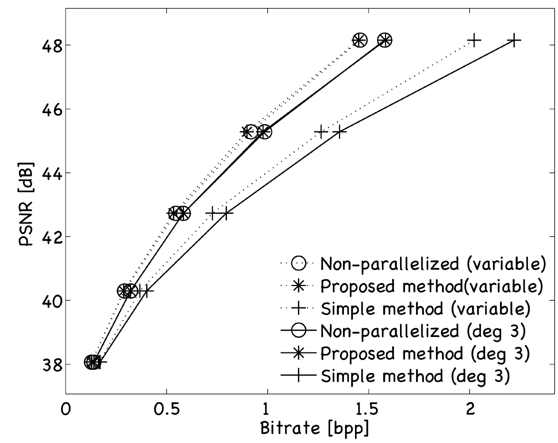


Fig. 12 Compression performance of parallel DVC framework (use of variable degree, *News sequence*)).

5. Conclusion

In this paper, we proposed a parallelized DVC framework. To realize a practical DVC system, we proposed an accurate bit probability estimation method with an index assignment method that we call the "Gray Code." We also gave the theoretical grounds for the proposed method. Moreover, we also proposed the approach in which the rate estimation method uses the derived bit probabilities to improve the coding performance. Simulation results show that the proposed framework can reduce the decode time by up to 35[%] with minimal parallelization loss and our proposal to increase the coding performance of the parallelized DVC framework reduces the bit rate by about 10[%].

Evaluating the circuit size of the proposed parallelized DVC system is future work.

References

[1] Information Technology Coding of Audio-Visual Objects Part 10: Advanced Video Coding (H.264), ISO/IEC JTC 1/SC 29 14496-10, 2005.

[2] Information Technology Generic Coding of Moving Pictures and Associated Audio Information: Video (MPEG-2), 2nd ed., ISO/IEC JTC 1/SC 29 13818-2, 2000.

[3] B. Girod, A.M. Aaron, S. Rane, and D.A.R.-M.D. Rebollo-Monedero, "Distributed video coding," Proc. IEEE, vol.93, pp.71–83, 2005.

[4] R. Puri, A. Majumdar, and K. Ramchandran, "PRISM: A video cod-

ing paradigm with motion estimation at the decoder," IEEE Trans. Image Process., vol.16, no.10, pp.2436–2448, 2007.
[5] R. Puri and K. Ramchandran, "PRISM: A new robust video coding architecture based on distributed compression principles," Proc. 40th Annual Allerton Conference on Communication, Control, and Computing, Allerton, Ill, USA, Oct. 2002.
[6] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," Proc. Asilomar Conference on Signals and Systems, Pacific Grove, CA, Nov. 2002.
[7] Z.M. Belkoura and T. Sikora, "Towards rate-decoder complexity optimisation in Turbo-coder based distributed video coding," PCS2006, April 2006.
[8] DISCOVER project: http://www.discoverdvc.org/
[9] R.G. Gallager, Low Density Parity Check Codes, Monograph, M.I.T. Press, 1963.
[10] D.J.C. MacKay, "Good error-correcting codes based on very sparse matrices," IEEE Trans. Inf. Theory, vol.45, no.2, pp.399–431, 1999.
[11] N.M. Cheung, H. Wang, and A. Ortega, "Correlation estimation for distributed source coding under information exchange constraints," IEEE International Conference on Image Processing (ICIP2005), Sept. 2005.
[12] L. Wei and Z. Wenjun, "Scalable non-binary distributed source coding using gray codes," in Multimedia Signal Processing, 2005 IEEE 7th Workshop on, pp.1–4, 2005.
[13] Y. Tonomura and T. Nakachi, "Performance evaluation of error resilient for distributed video coding using JPEG 2000," 20th SIP symposium, Japan, Nov. 2005.
[14] Y. Tonomura, T. Nakachi, and T. Fujii, "Distributed video coding using JPEG 2000 coding scheme," IEICE Trans. Fundamentals, vol.E90-A no.3, pp.581–589, March 2007.
[15] Z. He, L. Cao, and H. Cheng, "Correlation estimation and performance optimization for distributed image compression," Visual Communications and Image Processing, 2006.
[16] D. Chen, D. Varodayan, M. Flierl, and B. Girod, "Wyner-Ziv coding of multiview images with unsupervised learning of disparity and Gray code," Proc. IEEE International Conference on Image Processing, ICIP 2008, San Diego, California, Oct. 2008.
[17] F. Gray, "Pulse code communication," United States Patent Number 2632058. March 1953.
[18] Y. Tonomura, N. Takayuki, and T. Fujii, "Efficient index assignment by improved bit probability estimation for parallel processing of distributed video coding," Proc. IEEE International Conference on Speech and Signal Proc. ICASSP 2008, Las Vegas, Nevada, March 2008.
[19] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," EURASIP Signal Processing Journal, Special Section on Distributed Source Coding, vol.86, no.11, pp.3123–3130, Nov. 2006.
[20] S. Cheng and Z. Xiong, "Successive refinement for the Wyner-Ziv problem and layered code design," IEEE Trans. Signal Process., vol.53, no.8, pp.3269–3281, 2005.
[21] J. Ascenso, C. Brites, and F. Pereira, "Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding," 5th EURASIP Conf. on Speech and Image Processing, Multimedia Communications and Services, Smolenice, Slovak Republic, July 2005.
[22] H. Imai and S. Hirakawa, "A new multilevel coding method using Error-Correcting codes," IEEE Trans. Inf. Theory, vol.23, no.3, pp.371–377, 1977.
[23] C. Brites and F. Pereira, "Encoder rate control for transform domain Wyner-Ziv video coding," ICIP2007, USA, Sept. 2007.
[24] H. Jin, A. Khandekar, and R. McEliece, "Irregular repeat accumulate code," 2nd Int. Symp. on Turbo Codes and Related Topics, pp.1–8, Sept. 2000.
[25] LdpcOpt: http://lthcwww.epfl.ch/research/ldpcopt/

Yoshihide Tonomura received his B.S. and M.S. degrees in electronics engineering from Nagaoka University of Technology in 2002 and 2004, respectively. He joined NTT Network Innovation Laboratories in 2004. His research is focused on image processing theories and applications.

Takayuki Nakachi received a Ph.D. degree in electrical engineering from Keio University in 1997. Since he joined NTT Laboratories in 1997, He has been engaged in research on Super High Definition (SHD) image coding, especially in the area of lossless and near-lossless coding. In recent years, he has been researching scalable image/video coding in order to distribute SHD image contents. From 2006 to 2007, he was a visiting scientist at Stanford University, California. He is currently a senior research engineer of the media processing systems research group in NTT Network Innovation Laboratories. He is a member of IEEE.

Tatsuya Fujii received his B.S., M.S. and Ph.D. degrees, all in electrical engineering from the University of Tokyo, Tokyo, Japan, in 1986, 1988, and 1991, respectively. He joined NTT, Japan, in 1991. He has been researching parallel image processing and super-high-definition image communication networks. In 1996, he was a visiting researcher at Washington University in St. Louis. He is currently a group leader of the media processing systems research group in NTT Network Innovation Laboratories. He is a member of ITE of Japan and IEEE.

Hitoshi Kiya received his B.E. and M.E. degrees from Nagaoka University of Technology, Japan and a Dr.Eng. degree from Tokyo Metropolitan University, Tokyo, in 1980, 1982 and 1987, respectively, all in electrical engineering. In 1982, he joined Tokyo Metropolitan University where he is currently a Professor at the Graduate School of System Design. From 1995 to 1996 he attended the University of Sydney, NSW, Australia as a Visiting Fellow. His research interests lie in the areas of multirate systems and image processing, including filter bank design and theory, image and video coding, image watermarking and data hiding, and subband adaptive filtering. He received an Excellent Paper Award in 2008 from IEICE. He served as an associate editor of IEICE Transactions Fundamentals and IEEE Transactions on Signal Processing from 1998 to 2002 and from 1998 to 2000, respectively. He was also the Guest Editor-in-Chief for the Special Issues of IEICE Transactions published in 1999, 2006 and 2007, respectively. He was the Technical Committee Chair of Digital Signal Processing of IEICE from 2006 to 2007 and the General Chair of the Japan DSP Educator Conference from 2005 to 2007. He currently serves as a Vice President of IEICE Engineering Science Society and the Editor-in- Chief of IEICE ESS Magazine.