Enhanced Stochastic Bit Reshuffling for Fine Granular Scalable Video Coding

Wen-Hsiao Peng, Tihao Chiang, Hsueh-Ming Hang, and Chen-Yi Lee

National Chiao-Tung University
1001 Ta-Hsueh Rd., HsinChu 30010, Taiwan.
pawn@mail.si2lab.org, {tchiang, hmhang}@mail.nctu.edu.tw,
cylee@cc.nctu.edu.tw

Abstract. In this paper, we propose an enhanced stochastic bit reshuffling (SBR) scheme to deliver better subjective quality for fine granular scalable (FGS) video coding. Traditional bit-plane coding in FGS algorithm suffers from poor subjective quality due to zigzag and raster scanning order. To tackle this problem, our SBR rearranges the transmission order of each bit by its estimated rate-distortion performance. Particularly, we model the transform coefficient with a maximum likelihood based Laplacian distribution and incorporate it into the context probability model for content-aware parameter estimation. Moreover, we use a dynamic priority management scheme for the SBR. Experimental results show that our enhanced SBR together with context adaptive binary arithmetic coding offers up to 1.5dB PSNR improvement and shows better visual quality as compared to the scheme in MPEG-4 FGS.

1 Introduction

MPEG-4 fine granularity scalability (FGS) [1] offers a bit-plane coding scheme using variable length code (VLC). The same approach is also widely used in most of the advanced FGS algorithmes for fine granular SNR scalability. For each bit-plane, current approach performs the coding in a block-by-block manner. When the enhancement-layer is partially decoded, zigzag and raster scanning order may only refine the upper part with one extra bit-plane. Such uneven quality distribution causes subjective quality degradation.

In our prior work [2], a context adaptive binary arithmetic coding (CABAC) with an in-bit-plane bit reshuffling scheme was proposed to deliver higher coding efficiency and better subjective quality for the FGS algorithms. To improve coding efficiency, we partition each transform coefficient into significant and refinement bits. For the significant bit, we construct context model based on both the energy distribution in a block and the spatial correlation in the adjacenet blocks. The spatial correlation is considered using the significance status of the co-located coefficients in the adjacent blocks as context model. For the refinement bit, we simply use a fixed context probability model for coding. To improve subjective quality, the in-bit-plane reshuffling scheme reorders the coding bits of

K. Aizawa, Y. Nakamura, and S. Satoh (Eds.): PCM 2004, LNCS 3332, pp. 521–528, 2004.

a bit-plane in a rate-distortion sense. Particularly, for each bit, we use the associated non-zero context probability model to approximate its distrotion reduction at the decdoer and exploit the binary entropy to estimate its bit rate. Experimental results in [2] show that our CABAC significantly improves the coding efficiency while the in-bit-plane reshuffling only shows slight improvement on the subjective quality.

In this paper, we propose an enhanced stochastic bit reshuffling (SBR) scheme based on the CABAC in [2]. Instead of bit-plane by bit-plane coding, our enhanced SBR allows different coefficients be coded with different bin numbers. Moreover, for more accurate rate-distortion function estimation, we exploit a maximum likelihood based discrete Laplacian distribution with the context probability model to approximate the transform coefficients. Furthermore, to implement the SBR, we propose a dynamic priority management scheme that makes the SBR content aware considering the content dependent priority and a rate-distortion data update mechanism. Experimental results show that our CABAC [2] with the enhanced SBR can offer up to 1.5dB PSNR improvement and show better subjective quality as compared to the traditional scheme in MPEG-4 FGS.

The remainder of this paper is organized as follows: Section 2 introduces the SBR. Section 3 shows our parameter estimation schemes. Section 4 presents the dynamic coding flow for the SBR. Section 5 shows the experimental results. Finally, Section 6 summaries and concludes our work in this paper.

2 Stochastic Bit Reshuffling (SBR)

To address the uneven subjective quality issue, we propose a content-aware SBR scheme. Instead of deterministic zigzag and raster scanning order, we propose to reshuffle the coding bits in an optimized stochastic rate-distortion sense.

The bit reshuffling concept and criterion was first proposed in [3] for improving the rate-distortion performance of wavelet based image codec. For the reshuffling, in [3] each input bit is first assigned with two factors that are squared error reduction, $\triangle D$, and coding cost, $\triangle R$. With these parameters, the bit reshuffling reorders the input bits such that the associated $(\triangle D/\triangle R)$ is in descending order. Such an order leads to minimum distortion at any bit rate. However, the decoder generally does not have actual $\triangle D$ and $\triangle R$ for each input bit. Thus, estimated $\triangle D$ and $\triangle R$ are used to avoid sending the coding order. With the same estimation scheme at both encoder and decoder, the coding order can be implicitly known to both sides. Eq. (1) shows the condition of optimal coding order in *stochastic* rate-distortion sense where $\widehat{E}$ denotes taking estimation, the subscript of $\triangle D$ and $\triangle R$ represents the bit identification and the one of $\left(\widehat{E}\left[\triangle D\right]/\widehat{E}\left[\triangle R\right]\right)$ specifies the coding order.

In this paper, we use the same concept for the enhancement-layer bit reshuffling. Particularly, our estimation incorporates the context probability model to make the priority assignment content-aware so that the subjective quality can be improved. Moreover, we follow the optimized coding order in stochastic rate-distortion sense as in Eq. (1) to have similar or even better rate-distortion performance.

$$\left(\frac{\widehat{E}\left[\triangle D_i\right]}{\widehat{E}\left[\triangle R_i\right]}\right)_1 \geq \left(\frac{\widehat{E}\left[\triangle D_j\right]}{\widehat{E}\left[\triangle R_j\right]}\right)_2 \geq \left(\frac{\widehat{E}\left[\triangle D_k\right]}{\widehat{E}\left[\triangle R_k\right]}\right)_3 \geq .... \geq \left(\frac{\widehat{E}\left[\triangle D_l\right]}{\widehat{E}\left[\triangle R_l\right]}\right)_N \tag{1}$$

3 Parameter Estimation

To estimate the $\triangle D$ and $\triangle R$ of each bit, we use their expectations. For calculating the expection, we need the probability distribution of each transform coefficient. However, decoder gnerally does not have actual distribution. Thus, to minimize the overhead, we model each 4x4 integer transform coefficient [4] with discrete Laplacian distribution as defined in Eq. (2) where $X_{n,k}$ represents the nth zigzag ordered coefficient of block k and $x_{n,k}$ stands for its outcome. Particularly, we assume that the co-located coefficients are independently and identically distributed (i.i.d.).

$$P[X_{n,k} = x_{n,k}] \triangleq \frac{1-\alpha_n}{1+\alpha_n} \times (\alpha_n)^{|x_{n,k}|} \text{ where } n = 0\tilde{}15 \tag{2}$$

3.1 Estimation of Laplacian Parameter α_n

In Eq. (2), α_n (n:0˜15) is to be estimated. For the estimation, we use maximum likelihood principle. Given a set of M observed data and a presumed joint probability with an unknown parameter, the maximum likelihood estimator for the unknown parameter is the one that maximizes the joint probability.

For an enhancement-layer frame with M 4x4 blocks, the joint probability of the co-located coefficients at nth zigzag position can be written as in Eq. (3). According to the i.i.d. assumption, we can simplify the joint probability as M multiplication terms. Further, by substituting Eq. (2) into Eq. (3), we can obtain a close form formula for the joint probability.

$$\begin{aligned} &P[X_{n,1} = x_{n,1}, X_{n,2} = x_{n,2}, ..., X_{n,k} = x_{n,k}, ..., X_{n,M} = x_{n,M}] \\ &= \left(\frac{1-\alpha_n}{1+\alpha_n}\right)^M \times (\alpha_n)^{\sum_{k=1}^{M}|x_{n,k}|} \end{aligned} \tag{3}$$

By definition, the maximum likelihood estimator of α_n is the one that maximizes Eq. (3). To find the solution, one can take differentiation with respect to α_n and look for the root. Eq. (4) shows the maximum likelihood estimator of α_n based on Eq. (3). Note that the estimation is conducted at the encoder and the estimated parameters are transmitted to the decoder.

$$\alpha_n = \frac{-\mu_x^{-1} + \sqrt[2]{\left(\mu_x^{-1}\right)^2 + 4}}{2} \text{ where } \mu_x = \frac{\sum_{k=1}^{M}|x_{n,k}|}{M} \tag{4}$$

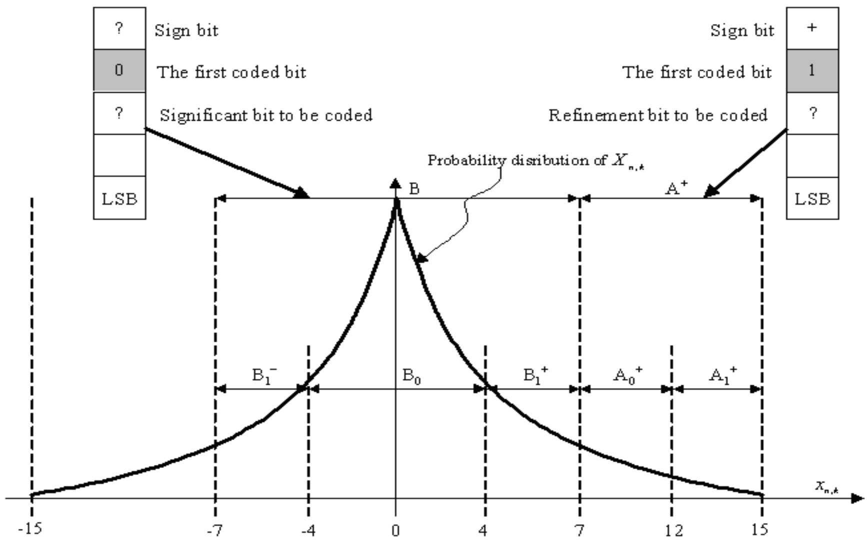

Fig. 1. Examples of ΔD estimation for significant bit and refinement bit

3.2 Estimation of ΔD

To estimate the $\triangle D$ of a coefficient bit, we calculate the expected squared error reduction at the decoder. Since the decoding of a coefficient bit is to reduce the uncertainty of a coefficient, we can calculate the expected squared error reduction from the decrease of uncertainty interval.

In Fig. 1, we depict the estimated distribution of a 4-bit coefficient and give two examples for illustrating the decrease of uncertainty interval. Without loss of generality, we show the example of significant bit at the left hand side where the first bit is coded as zero. On the other hand, the right hand side depicts the case of refinement bit where the first bit is non-zero. From the coded bits, we can identify the uncertainty interval in which the actual value is located. For the significant bit case, we know that the actual value is confined within the interval B. Similarly, for the refinement bit case, we learn that the actual value falls in the interval A^+. Given the interval derived from the previously coded bits, the next bit for coding is to further decrease the uncertainty interval. For instance, the significant bit to be coded is to determine that the actual value is in the subinterval B_0 or $\left\{B_1^+ \cup B_1^-\right\}$. Similarly, the refinement bit to be coded is to determine that the actual value is in the subinterval A_0^+ or A_1^+.

From the decrease of uncertainty interval, we can calculate the reduction of expected squared error. Specifically, at the decoder, the expected squared error in an interval is the variance within the interval. Thus, we can express our $\triangle D$ estimation as the reduction of variance. Eq. (5) formulates our $\triangle D$ estimation for the significant bit case in Fig. 1 where $\mathrm{Var}[X_{n,k}|X_{n,k} \in B]$ represents the conditional variance of $X_{n,k}$ given that $X_{n,k}$ is in the interval B. Similarly, we have the conditional variances for the other subintervals. Since we do not know in which subinterval the actual value is located, each subinterval variance is further weighted by its probability. To simplify the expression, we merge the vairances of B_1^+ and B_1^- in Eq. (5) because they are identical.

$$\begin{aligned}
&\widehat{E}[\triangle D_{n,k,B,significant}] \\
&\triangleq \mathrm{Var}[X_{n,k}|X_{n,k} \in B] \\
&- (P[X_{n,k} \in \{B_1^+ \cup B_1^-\}\,|X_{n,k} \in B]) \times \mathrm{Var}[X_{n,k}|X_{n,k} \in B_1^+] \\
&- P[X_{n,k} \in B_0|X_{n,k} \in B] \times \mathrm{Var}[X_{n,k}|X_{n,k} \in B_0] \\
&\cong \mathrm{Var}[X_{n,k}|X_{n,k} \in B] \\
&- \mathit{SignificantContextP}(\mathit{ContextIndex}(n,k,B),1) \times \mathrm{Var}[X_{n,k}|X_{n,k} \in B_1^+]) \\
&- \mathit{SignificantContextP}(\mathit{ContextIndex}(n,k,B),0) \times \mathrm{Var}[X_{n,k}|X_{n,k} \in B_0]
\end{aligned} \tag{5}$$

In Eq. (5), the co-located significant bits within the same interval have the same estimated $\triangle D$ because the co-located coefficients have identical Laplacian model. To perform the reshuffling with content aware so that the regions that contain more energy are with higher coding priority, in Eq. (5) the subinterval probabilities are approximated with the context probability models where *SignificantContextP*$(ContextIndex(n,k,B),1)$ denotes the non-zero context probability model for the significant bit of $X_{n,k}$ in the interval B and the *SignificantContextP*$(ContextIndex(n,k,B),0)$ represents its zero probability. Recall that we exploit the significance status of co-located coefficients in the adjacent blocks as significant bit context model [2]. Using the context probability model for substitution makes the $\triangle D$ estimation of significant bit become region dependent, i.e., content aware.

Following the same procedure, one can obtain the $\triangle D$ estimation for the other significant bits and refinement bits. Eq. (6) shows the one for the refinement bit example in Fig. 1. Particularly, the conditional probability in Eq. (6) is from the estimated Laplacian model.

$$\begin{aligned}
&\widehat{E}[\triangle D_{n,k,A^+,refinement}] \\
&\triangleq \mathrm{Var}[X_{n,k}|X_{n,k} \in A^+] \\
&- P(X_{n,k} \in A_1^+|X_{n,k} \in A^+) \times \mathrm{Var}[X_{n,k}|X_{n,k} \in A_1^+] \\
&- P(X_{n,k} \in A_0^+|X_{n,k} \in A^+) \times \mathrm{Var}[X_{n,k}|X_{n,k} \in A_0^+]
\end{aligned} \tag{6}$$

3.3 Estimation of ΔR

To estimate the expected coding bit rate of an input bit, we use the binary entropy[1] which represents the minimum expected coding bit rate for an input bit. Eq. (7) defines our $\triangle R$ estimation for the significant bit example in Fig.

[1] $H_b(P(1)) = -P(1) \times \log_2(P(1)) - (1 - P(1)) \times \log_2(1 - P(1))$, where $P(1)$ is the non-zero probability of the coding bit

1. The first term represents the binary entropy of a significant bit using the associated context probability model as an argument while the second term denotes the cost from a sign bit. The sign bit is considered as partial cost of significant bit because the decoder can only perform the reconstruction after the sign is received. Recall that each sign bit averagely consumes one bit and it is only coded after a non-zero significant bit. Thus, the cost of sign is weighted by the non-zero context probability model.

$$\begin{aligned}&\widehat{E}[\triangle R_{n,k,B,significant}] \\ &\triangleq H_b(\mathit{SignificantContextP}\,(ContextIndex(n,k,B),1)) \\ &+ \mathit{SignificantContextP}\,(ContextIndex(n,k,B),1) \times 1\end{aligned} \tag{7}$$

Eq. (8) illustrates our $\triangle R$ estimation for the refinement bit example in Fig. 1. For the binary entropy calculation, we use the estimated probability of refinement bit as an argument. For instance, we use $P(X_{n,k} \in A_1^+ | X_{n,k} \in A^+)$ as the argument for the refinement bit example in Fig. 1. For the other significant bits and refinement bits, one can use the same methodology to estimate the $\triangle R$.

$$\begin{aligned}&\widehat{E}[\triangle R_{n,k,A^+,refinement}] \\ &\triangleq H_b(P(X_{n,k} \in A_1^+ | X_{n,k} \in A^+))\end{aligned} \tag{8}$$

4 Dynamic Priority Management for SBR

To maintain the coding priority, we exploit two dynamic coding lists for the reshuffling of significant and refinement bits. Each bit in the associated list is allocated a register to record its bit location and estimated rate-distortion data. Particularly, to avoid sending the bin position for each bit, the coding of a coefficient is sequentially from the MSB to the LSB. In addition, to avoid coding the redundant bits after the EOSP, the coding of significant bits in Part II always follows zigzag order. In [2], EOSP denotes the location of last non-zero significant bit of a bit-plane. For each bit-plane, the significant bits in Part II refer to those after the EOSP of previously coded bit-plane in zigzag order. Given these constraints, our priority management includes the following 5 steps:

1. **Coding list initialization:** The initialization estimates the rate-distortion data for all the DC coefficient bits at the MSB bit-plane of the enhancemnt-layer frame and puts the data in the coding lists.
2. **Coding list reshuffling:** After the initialization, we perform the reshuffling to identify the highest priority bit in the lists, i.e., the one with maximal $\left(\widehat{E}\left[\triangle D\right]/\widehat{E}\left[\triangle R\right]\right)$. In addition, the reshuffling is performed after the coding of each input bit.
3. **CABAC:** Once the highest priority bit is identified, we follow the CABAC scheme in [2] for coding.

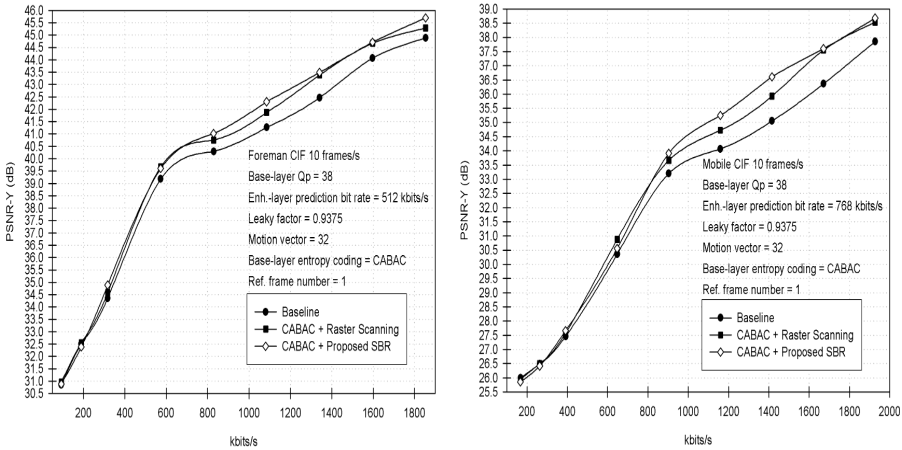

Fig. 2. PSNR comparision of traditional bit-plane coding (Baseline), CABAC [2] + Raster Scanning and CABAC + Proposed SBR.

Fig. 3. Subjective quality comparision of Baseline and CABAC [2] + SBR with enh.-layer truncated at 384kbits/s.

4. **Rate-distortion data update:** After the coding of each input bit, we update parts of the rate-distortion data in the coding lists. Such update is to fully utilize the coded information to enhance the content aware bit reshuffling. Specifically, we update those registers whose context index or context probability model are chnaged after the coding of highest priority bit. Particularly, the rate-distortion data of refinement bit is not required for update since it is from the fixed Laplacian model.

5. **Input bit inclusion:** After the update, we further include the adjacent bit in the lower bit-plane and the next zigzag ordered coefficent bit in the same bit-plane for reshuffling. The steps 2 to 5 are repeated until all the input bits are coded.

5 Experiments

In this Section, we assess the rate-distortion performance of our SBR objectively and subjectively. For the experiments, we use H.264 JM4 [4] as base-layer and RFGS [5] as enhancement-layer prediction scheme. For comparision, different bit-plane coding schemes use identical encoder parameters. Particularly, the scheme in MPEG-4 FGS is used as baseline.

Fig. 2 shows that the CABAC [2] with enhanced SBR further boosts the PSNR by 0.2~0.5dB as compared to one with raster scanning. Moreover, it reveals 1.0~1.5dB improvement over the baseline. In addition, Fig. 3 shows that the enhanced SBR offers better subjective quality. The baseline reveals obvious blocking artifact at the lower part of the decoded frame. In contrast, our CABAC with enhaced SBR shows more uniform quality over the entire frame.

6 Conclusion

In this paper, we present an enhanced SBR scheme to improve the subjective quality of traditional bit-plane coding in FGS algorithms. We generalize the concept of bit-plane coding and reshuffle each coding bit according to its estimated rate-distortion performance. Experimental results show that our CABAC [2] with enhanced SBR can deliver higher coding efficiency and better subjective quality. Furthermore, with appropriate modification, the proposed SBR can be applied in other embedded entropy coding.

References

1. W. Li, "Overview of Fine Granularity Scalability in MPEG-4 Standard," *IEEE Trans. on Circuits Syst. for Video Technol.*, vol. 11, no. 3, pp. 301–317, 2001.
2. W. H. Peng, C. N. Wang, T. Chiang, and H. M. Hang, "Context Adaptive Binary Arithmetic Coding With Stochastic Bit Reshuffling For Advanced Fine Granularity Scalability," in *IEEE ISCAS*, 2004.
3. J. Li and S. Lei, "An Embedded Still Image Coder with Rate-Distortion Optimization," *IEEE Trans. on Image Processing*, vol. 9, no. 7, pp. 297–307, 2003.
4. T. Weigand, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 — ISO/IEC 14496-10 AVC)," *JVT-G050*, 2003.
5. H. C. Huang and T. Chiang, "Stack Roubst Fine Granularity Scalability," in *IEEE ISCAS*, 2004.