VARIABLE BLOCK-SIZE TRANSFORM AND ENTROPY CODING AT THE ENHANCEMENT LAYER OF FGS

Jungong Han[1], Xiaoyan Sun[2], Feng Wu[2], Shipeng Li[2], Zhaoyang Lu[1]*

[1] ISN national laboratory of Xidian University, Xi'an, 710071
[2] Microsoft Research Asia, Beijing, 100080

ABSTRACT

This paper proposes the variable block-size transform and context-based entropy coding techniques for the enhancement layer of FGS (Fine Granularity Scalable) video coding. First, the variable block-size transform is introduced into the enhancement layer to improve the performance of FGS in terms of both visual quality and PSNR. Different from that used in the traditional single layer coding, an R-D selection algorithm is proposed to optimally decide the transform size of each block under consideration of the consistent performance at a range of bit rate. Furthermore, to fully take the advantage of the characteristics and correlations of symbols coded in the FGS enhancement layer, different context models are designed for the arithmetic coding according to symbol type and transform size. Experimental results show that the coding efficiency of FGS can be increased 0.2-0.9dB with the proposed techniques.

1. INTRODUCTION

Recently, Fine Granularity Scalability (FGS) video coding technique defined in the Amendment of MPEG-4 [1] has become an attractive topic because it can provide scalable bitstream to different bandwidth channels. However, the low coding efficiency is the main disadvantage that prevents MPEG-4 FGS from being widely deployed in video streaming applications since its motion prediction is always based on the lowest quality base layer. The Progressive Fine Granularity Scalable (PFGS) coding scheme [2] is a significant improvement over MPEG-4 FGS by introducing two prediction loops with different quality references. Moreover, macroblock-based PFGS (MBPFGS) [3] and its improved version [4] provides a good trade-off between coding efficiency improvement and drifting error reduction by optimally selecting the reference of the enhancement layer at macroblock level. In addition, RFGS proposed in [5] also presents the similar performance with two-loop prediction and drifting attenuation techniques. Generally, all these schemes mainly focus on motion estimation, motion compensation and mode decision. Other coding modules in the FGS coding scheme, such as DCT transform and entropy coding, attract few researchers' attention, which does not means that they can not be improved further.

Currently, no matter what techniques are applied in the modules of motion prediction and motion compensation, the residual data coded in enhancement layer of FGS schemes, such as PFGS and RFGS, firstly goes through the 8x8 DCT transform. The generated DCT coefficients are coded by bit-plane variant length coding (VLC) method [6] to produce the bitstream. Compared with other VLC coding methods, bit-plane coding technique has some advantages besides fine scalability. Firstly, the RUN and EOP are coded jointly as a 2D symbol to save coded bits. Secondly, since the bit-plane statistics are much dependent of the bit plane level, four VLC tables are designed to code the corresponding bit planes and achieve better coding performance. As a result, the bit-plane coding can outperform the traditional VLC used in MPEG-2 and MPEG-4 up to 20% at moderate and high bit rates [6]. However, since the data of enhancement layer has different local correlations, the 8x8 DCT transform may not fully utilize such correlations. Furthermore, the correlations between coded symbols are not been taken into account in the current bit-plane coding because the symbols coded in one bit-plane are independent from that in another bit-plane.

To solve the above problems, a variable block-size transform technique is proposed to the enhancement layer coding in this paper. For simplicity, this paper only uses 8x8 DCT transform and 4x4 DCT transform in the enhancement layer coding to accommodate different local correlations. The extension of 4x8 and 8x4 DCT transforms can be done with the similar way. Since there is no quantization in the FGS enhancement layer coding, an R-D selection algorithm based on the estimated parameters is presented to select the transform size from 8x8 and 4x4. Furthermore, the context-based arithmetic coding is utilized to code bit-planes with appropriate contexts models designed according to different transform sizes. It further takes advantages of the characteristics and correlations of symbols to be coded in the FGS enhancement layer.

---

* This work was performed when the author was with Microsoft Research Asia

The rest of this paper is arranged as follows. Section 2 introduces the basic framework of the proposed coding scheme. Section 3 describes the variable block-size transform and the R-D selection of the transform size. The context models of arithmetic coding are designed in section 4. Section 5 gives the experimental results. Finally, Section 6 concludes this paper.

2. THE PROPOSED CODING SCHEME

Figure 1 depicts the block diagram of the proposed scheme, which can be applied in either MPFG-4 FGS or other advanced FGS coding methods (e.g., PFGS and RFGS). For the better coding performance, the MBPFGS built upon JVT H.26L [7] is chosen as the exemplified scalable video codec in this paper.

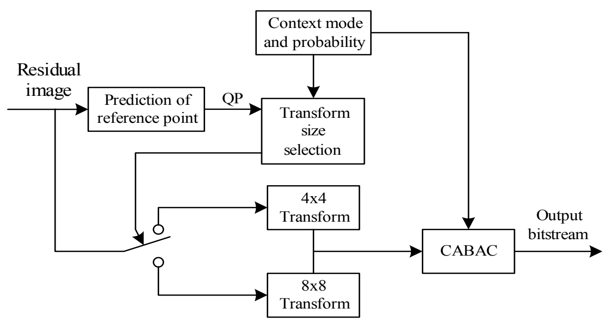

Figure 1: The block diagram of the proposed coding scheme.

As shown in Figure 1, the proposed improve coding scheme consists of five main modules, i.e., prediction of the reference point, transform size selection, context mode and probability, transform and arithmetic coding. Since the quantization parameters of the reference point at the enhancement layer, which has to be used in the R-D mode selection, can only be obtained after the enhancement bitstream generated, the residual image is first input the prediction module to estimate the values. Then the R-D costs of the 4x4 and 8x8 DCT transforms under the corresponding arithmetic coding are computed for each block so as to make a proper transform mode selection. Note that the context model and probability should be updated for each block in order to get more accurate predictive bit cost. Finally, the context-based arithmetic coding is applied to produce bit stream relying on the selected transform size.

3. THE VARIABLE BLOCK-SIZE TRANSFORM FOR FGS

The advantages of variable block-size transforms (VBS) have been proven in non-scalable coding schemes [8]. But to the best of our knowledge, VBS has not been applied in current FGS coding schemes. In this paper, we introduce VBS algorithm into scalable coding schemes and propose an algorithm to adaptively select transform size for each block at the enhancement layer by using the R-D optimal method.

The 8x8 and 4x4 transform matrixes used in this paper are exactly same as that given in [8]. The key problem to be solved in this paper is how to optimally decide the transform size. The VBS-supported non-scalable coding [8] makes use of SAD as the criterion at a given bit-rate to select optimal transform size. It should be pointed out that this criterion is not appropriate for the scalable coding case, because DCT coefficients of enhancement layer are decomposed as binary format and are coded bit-plane by bit-plane. That is, minimal SAD does not assure higher coding efficiency at the whole range of bit rate.

In order to decide the feasible signal length exploited by transform, the following R-D optimal algorithm is proposed, denoted as

$$\min \sum_i \left[D(x_i, M) + \lambda_t \times R(x_i, M) \right], \qquad (1)$$

where x_i represents the block i, M represents the candidate transform size, λ_t is the Lagrange multiplier, which is associated with the estimated QP, $D(x_i,M)$ means the distortion of a block, measured by SSD in the proposed algorithm. $R(x_i,M)$ represents the bits needed to code a block. In general, it's difficult to get an optimal solution of (1) in an image because of the dependence among blocks. The sub-optimal solution is used instead. In this case, the optimization problem in (1) is simplified as

$$\sum_i \min \left[D(x_i, M) + \lambda_t \times R(x_i, M) \right]. \qquad (2)$$

Obviously, the solution of (2) is to optimally select the transform size for each block. In details, the R-D costs of 8x8 and 4x4 transforms are computed separately. Then the mode with lower cost will be selected as the transform for this block. However, different from non-scalable coding, there are some special problems should be taken into account when computing the R-D information of each block in FGS enhancement layer.

Firstly, each block has a flag, namely CBP, indicating whether or not its most significant bit plane reaches. And, this sign will be coded before DCT coefficients of bit-planes. In some most significant bit-planes, such as MSB-0 and MSB-1, CBP will consume many bits. Apart from the DCT coefficients, the selected transform will also affect the number of bits used to code the CBP information. For example, for the 8x8 transform, a block needs only one CBP flag; while for the 4x4 transform, a block has to code four CBP flags in representing the states of different sub-blocks. Thus, the R-D criteria should be extended to

$$\sum_i \min \left[D(x_i, M) + \lambda_t \times (R(x_i, M) + R_{cbp}(x_i, M)) \right] \qquad (3)$$

where $R_{cbp}(x_i,M)$ represents bits needed to code the CBP flags. The R-D cost of different transforms will be computed with the equation (3).

Secondly, since the reference plays a very important role in PFGS, the reference point is selected as the optimization target for variable block-size transform in this paper. However, the entropy coding deals with bit-planes at frame level, while the DCT transform is done in block level. In other words, for each enhancement block, it is difficult to know how many bit-planes are used to reconstruct the reference. One way to solve this problem is to pre-encode the residual image with 8x8 DCT transform and the normal CABAC [9] coding method. In this case, the number of bit-planes for the reconstruction of reference in each enhancement block is roughly obtained and will be utilized as the predicted information of the reference point. Similarly, the proposed scheme can also make use of this method to predict the information of any optimal target in FGS.

Thirdly, how to exactly compute the second item of (3) is another problem because the number of output bits of context-based arithmetic coding is greatly influenced by context index and block's coding order. They are not available until the transform is selected. Thus, we have to predict the value of the second item of (3). Assume that the mode of block *i* is being decided in the proposed scheme. Firstly, initial probabilities of all context indexes before any process are recorded. Then, the R-D cost will be computed using (3), and the alterative probabilities of all context indexes will be re-recorded. If the 8x8 transform is selected, the probabilities of context indexes corresponding to the 8x8 transform will be updated with re-recorded values, meanwhile the probabilities of context indexes corresponding to the 4x4 transform will be updated by recoded initial values. Definitely, this method considers correlations between different block's context models and is able to provide more accurate rate prediction.

4. THE CONTEXT-BASED ARITHMETIC CODING

The context-based arithmetic coding has been successfully applied in non-scalable coding [9] because it can fully exploit the characteristics and correlations between symbols by establishing different context models. In this paper, we also make use of this technique to code various symbols. In this section, we will briefly introduce context models designed for different symbols.

A. Context model for overhead information.

The overhead information in the PFGS scheme includes the flag of transform size and PFGS modes. It can be observed that these data of neighboring blocks have strong correlation in consistent regions. So, the left block and the top block of the current block are used for designing the context model.

$$S : S_L + S_T + C \tag{4}$$

Here, S represents the mode of the current block, S_L and S_T represent modes of the left block and the top block, respectively. C is a constant for producing different context indexes. When the flag of transform size is coded, C is equal to 3, otherwise C is 6.

B. Context models for CBP

CBPs of neighboring blocks also have strong correlation. Similar context model as (4) is used except that the constant is equal to 9. It should be pointed out that one block has additional four CBPs except for one original CBP symbol if the transform 4x4 is selected, which indicates we have to code four additional CBPs in this case. Thus, how to code CBP of one sub-block if the transform size of its neighboring blocks used to compute context index are 8x8 DCT is another problem. In this paper, we use an empirical value as substitute. When coding MSB-0, the value is equal to 1, otherwise it is equal to 0.

C. Context models for DCT coefficients

As shown in Figure 1, DCT coefficients of one block will be coded according to its transform size.

i) Context models for 8x8 DCT coefficients

It is reasonable to design different context models for different bit-planes because of distinct properties. Let's define $f = 0$ if both CBPs of the left and top blocks are 1; otherwise $f = 1$. The zigzag order of the preceding symbol, denotes as m. Define $n = 0$, if the number of non-zero bits of the upper bit plane is less than 20; otherwise $n = 1$. The context model of MSB-0 is chosen as:

If $f = 0$

$$x_c : \begin{cases} 12 & if \quad m < 10 \\ 13 & if \quad 10 <= m < 35 \\ 14 & else \end{cases} \tag{5}$$

Here, x_c represents current symbol to be coded. If $f = 1$, the same model is used except for different constants. Differently, the context models of MSB-1 are shown as following.

If $n = 0$

$$x_c : x_{uc} + \begin{cases} 15 & m < 20 \\ 17 & 20 <= m < 64 \end{cases} \tag{6}$$

Here, x_{uc} represents the value of co-located symbol of the upper bit plane.

If $n = 1$

$$x_c : x_{uc} + \begin{cases} 19 & m < 30 \\ 21 & 30 <= m < 64 \end{cases} \tag{7}$$

In addition, similar context models are formed for other bit planes except for different constants.

ii) Context models for 4x4 DCT coefficient

The properties and correlations of 4x4 DCT are similar to that of 8x8 DCT. Thus, we design context models with the same idea.

5. EXPERIMENTAL RESULTS

The comparisons between 8x8 DCT with bit-plane coding and the proposed technique are performed and the experimental results are given in this section. The testing sequences Mobile and Foreman with CIF format are coded at 10 fps. Only the first picture is I picture and the others are P pictures. The new software of PFGS based on H.264 6.1e software is used in this paper.

Firstly, we compare 8x8 DCT transform with the normal CABAC and variable block-size transform with CABAC proposed in this paper frame by frame for verifying R-D selection algorithm. We truncate the enhancement layer bitstream at the reference point, as shown in Figure 2, because VBS algorithm of this scheme selects the reference point as the optimization target. It can be observed that the proposed techniques show about 0.2-0.5 dB gain over 8x8 DCT at this point.

We also compare traditional VLC-based bit-plane coding with the proposed scheme by using the average PSNR. As shown in Figure 3, the PSNR versus bit-rate curves of the proposed scheme is always higher than that of the traditional scheme. The gain is about 0.2-0.9 dB at different bit-rates.

6. CONCLUSIONS

In this paper, the variable block-size transform and CABAC-based coding techniques are proposed to the enhancement layer of FGS. It first introduces the basic scheme about how to use variable block-size transform techniques to FGS. After that, different context models are designed for arithmetic coding according to different transform size. Experimental results show that the proposed scheme can significantly improve the coding efficiency.

REFERENCES

[1] W. Li, Overview of fine granularity scalability in MPEG-4 video standard, IEEE Trans. on CSVT, vol.11, no 3, 301-317, 2001.

[2] F. Wu, S. Li, Y.-Q. Zhang, A framework for efficient progressive fine granular scalable video coding, IEEE trans. on CSVT, vol. 11, no 3, 332-344, 2001.

[3] X. Sun, F. Wu, S. Li, W. Gao and Y.-Q. Zhang, Macroblock-based progressive fine granularity scalable video coding, ICME, 461-464, 2001.

[4] Z. Yang, F. Wu, S. Li, Rate distortion optimized mode decision in the scalable video coding, ICIP, 2003.

[5] H. Huang, C. Wang, T. Chiang, A robust fine granularity scalability using trellis-based predictive leak, IEEE Trans. on CSVT, vol.12, no 6, 372-385, 2002.

[6] F. Ling, W. Li, H. Sun, Bitplane coding of DCT coefficients for image and video compression, Visual Communications and Image Processing, vol. 3653, 500-508, 1999.

[7] Y. He, F. Wu, S. Li, Y. Zhong, S. Yang, H.26L-based fine granularity scalable video coding, ISCAS, vol. 4, 548-551, 2002.

[8] M. Wien, Variable block-size transform for H.264/AVC, IEEE Trans. on CSVT, vol.13, No.7, 604-619, 2003.

[9] D. Marpe, H. Schwarz, T. Wiegand, Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard, IEEE Trans. on CSVT, vol.13, no.7, 620-636, 2003.

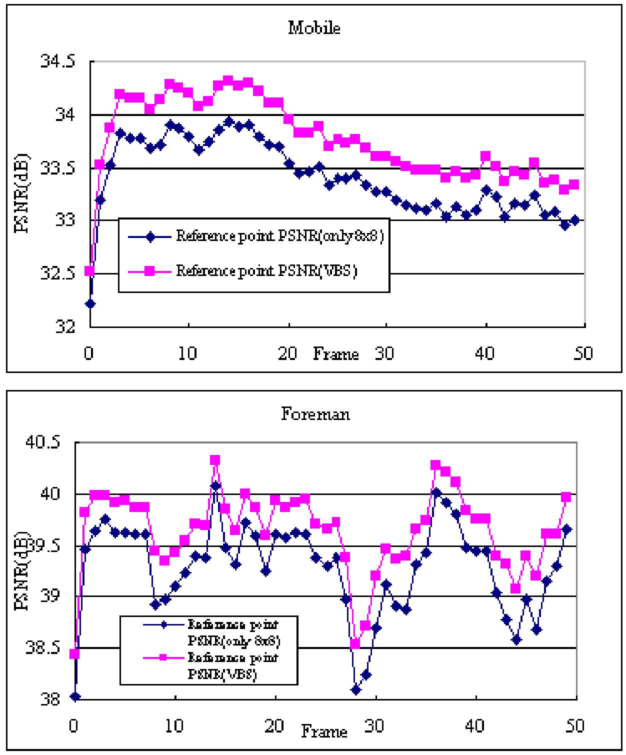

Figure 2: The PSNR versus frames at the reference point.

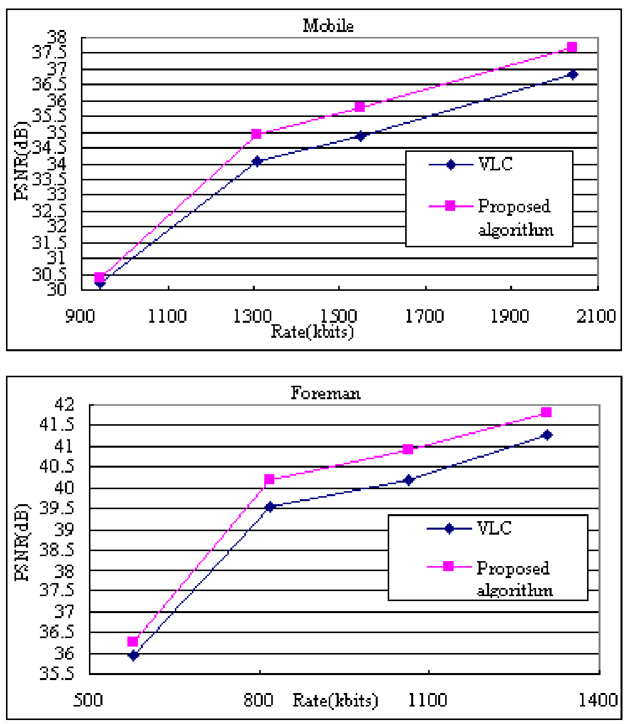

Figure 3: The PSNR versus rate curves of 8x8 DCT with bit-plane coding and the proposed technique.