Optimum Quantization Parameters for Mode Decision in Scalable Extension of H.264/AVC Video Codec

Seung-Hwan Kim and Yo-Sung Ho

Gwangju Institute of Science and Technology (GIST),
1 Oryong-dong Buk-gu, Gwangju 500-712, Korea
{kshkim, hoyo}@gist.ac.kr

Abstract. In the joint scalable video model (JSVM), selection of quantization parameters for mode decision (QPMD) and bit rate control (QPBC) is important for efficient coding performance. For quality (SNR) scalability, QPMD is adjusted only to the base layer QPBC in JSVM. Thus, it reduces coding efficiency in the enhancement layer. In this paper, we propose a new method for selecting optimum quantization parameters for mode decision (OQPMD) in order to improve coding efficiency for both the base and the enhancement layers. For the base layer, we propose optimum scaling factors in each decomposition stage. We also propose an offset quantization parameter for the enhancement layer. Experimental results show that the proposed method increases the average PSNR value up to 0.8dB.

Keywords: Scalable Video Coding, H.264, Quantization, Mode Decision.

1 Introduction

In recent years, scalable extension of the H.264/AVC video codec using motion-compensated temporal filtering (MCTF) has been investigated [1] [2]. The moving picture experts group (MPEG) of ISO/IEC and the video coding experts group (VCEG) of ITU-T agreed to jointly finalize the scalable video coding (SVC) project as an amendment of their H.264/AVC standard [3], and the scalable extension of H.264/AVC was selected as the first working draft [4]. The working draft provides a specification of the bit-stream syntax and the decoding process. The reference encoding process is described in the joint scalable video model (JSVM) [5].

The main idea of JSVM is to extend the hybrid video coding approach of H.264/AVC towards MCTF using a lifting framework [1]. Since the lifting structure is invertible without requiring invertible prediction and update steps, motion-compensated prediction using any possible motion model can be incorporated into the prediction and update steps. Using an efficient motion model of the H.264/AVC standard [1], both the prediction and update steps are processed as the motion-compensated prediction of B slices specified in the H.264/AVC standard.

Furthermore, the open-loop structure of the temporal subband representation enables us to incorporate temporal and quality (SNR) scalabilities efficiently [1].

The basic coding scheme for achieving spatio-temporal scalability and quality scalability can be classified as a layered video codec. The coding structure depends on the scalability space that is required by the application [6]. Figure 1 shows a block

Y.-S. Ho and H.J. Kim (Eds.): PCM 2005, Part I, LNCS 3767, pp. 179 – 190, 2005.

diagram for a typical scenario with three spatial layers. In each layer, an independent hierarchical motion-compensated prediction structure with motion parameters is employed. This hierarchical structure provides a temporal scalable representation of a sequence of input pictures and that structure is also suitable for efficiently incorporating spatial and quality scalability. Redundancy between different layers is exploited by inter-layer prediction that includes a prediction mechanism for motion parameters as well as texture data.

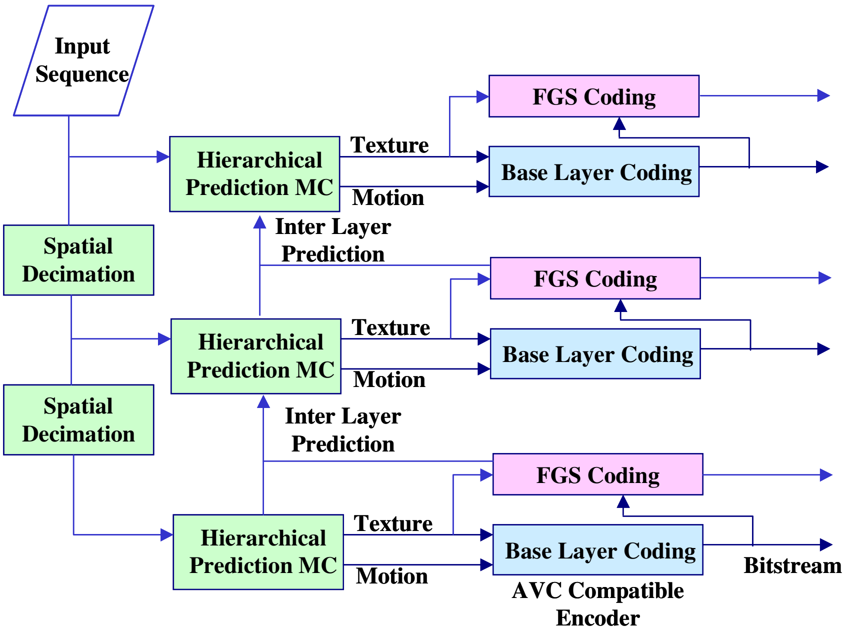


Fig. 1. Basic Structure for the Scalable Extension of H.264/AVC

A base representation of the input sequence in each layer is obtained by transform coding similar to that of H.264/AVC. In the H.264/AVC video coding standard, quantization parameters for mode decision (QPMD) and quantization parameters for bit rate control (QPBC) are always the same. However, in JSVM, QPMD and QPBC are often different in each decomposition stage. QPMD is also adjusted to the base layer QPBC. Therefore, these mismatches between QPMD and QPBC degrade the coding performance. In this paper, we propose a new method to select optimum quantization parameters for mode decision (OQPMD). Selection of OQPMD is achieved by improving coding efficiency for the base and enhancement layers.

This paper is organized as follows. After we introduce the lifting scheme for two-channel decomposition in Section 2, we describe fine granular scalability in Section 3. Then we explain two kinds of QPs in JSVM in Section 4, we propose a new method for optimum quantization parameter selection for mode decision in Section 5. After experimental results are presented in Section 6, we conclude this paper in Section 7.

2 Motion-Compensated Temporal Filtering (MCTF)

Figure 2 illustrates the lifting representation of the analysis-synthesis filter bank. At the analysis side, the odd samples $s[2k+1]$ of a given signal s are predicted by a linear combination of the even samples $s[2k]$ using a prediction operator $\mathrm{P}(s[2k+1])$ and a high-pass signal $h[k]$ is formed by prediction residuals. A corresponding low-pass signal $l[k]$ is obtained by adding a linear combination of the prediction residuals $h[k]$ to the even samples $s[2k]$ of the input signal s using the update operator $\mathrm{U}(s[2k])$ [7].

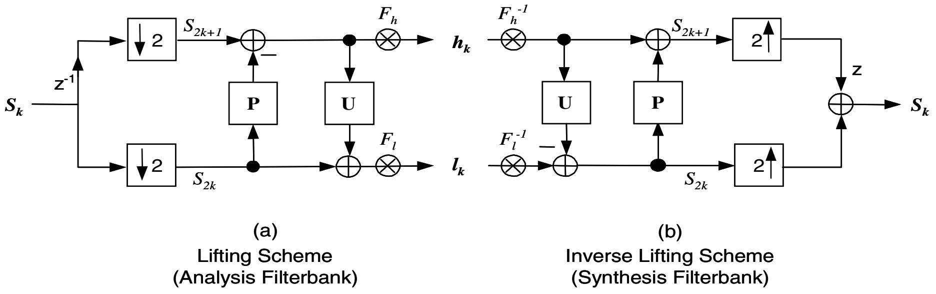


Fig. 2. Lifting Representation of the Analysis-Synthesis Filter Bank

By repeating the analysis process in the lifting scheme, we have implemented the hierarchical prediction structure in JSVM. The hierarchical prediction structure can either be realized by the coding of hierarchical pictures, or by the generalized motion-compensated temporal filtering (MCTF). The MCTF process is composed of two separate operation; prediction and update. The prediction operation is very similar to the conventional motion compensation, except that it uses the original image as the reference frame. However, with update operation, we compensate the drift due to open-loop structure and improve the coding efficiency.

In Figure 3, an example of the hierarchical prediction structure for a group of eight pictures with dyadic temporal scalability is depicted. The first picture of a video sequence is intra-coded as the instantaneous decoder refresh (IDR) picture that is a kind of the key picture.

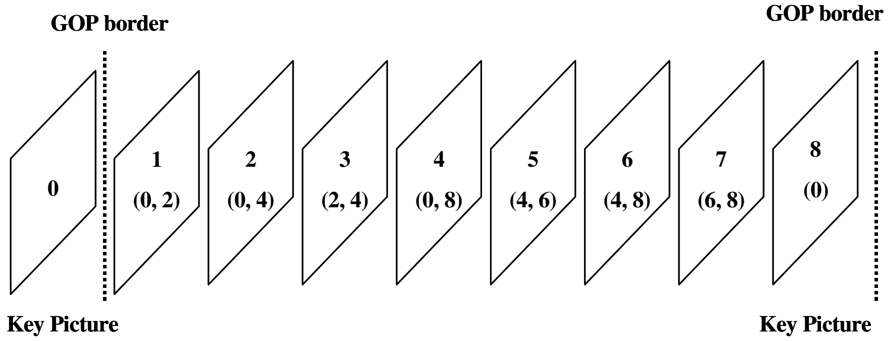


Fig. 3. The Hierarchical Prediction Structure

The key picture is located in regular or even irregular intervals. It also either intra-coded or inter-coded by using previous key picture as references for motion-compensated prediction. The sequence of the key picture is independent from any other pictures of the video sequence, and it generally represents the minimal temporal resolution that can be decoded. Furthermore, the key picture can be considered as re-synchronization points between encoder and decoder. For the other pictures, an open-loop encoder control can be used and the reference pictures including progressive refinement slices are used for motion-compensated prediction. However, the motion-compensated prediction signal for the key pictures is generated by using only the base layer representation of the reference key picture without FGS enhancement. Thus, the base representation of the key picture is always identical at the encoder and decoder side regardless of how many NAL units have been discarded or truncated in the decoder side [8].

In Fig. 3, the hierarchical prediction structure is implemented as follows. The picture numbered four is predicted by using the surrounding key pictures (zero and eight) as references. It depends only on the key pictures, and represents the next higher temporal resolution together with the key pictures. The pictures numbered two and six of the next temporal level are predicted by using only the picture of the lower temporal resolution as references, etc. It is obvious that this hierarchical prediction structure inherently provides temporal scalability; but it turned out that it also offers the possibility to efficiently integrating quality and spatial scalability [8].

The hierarchical picture coding can be extended to motion-compensated filtering. Motion-compensated update operations in MCTF are introduced in addition to the motion-compensated prediction. At the encoder side, the MCTF decomposition process starts at the highest temporal resolution. The group of pictures is partitioned into two groups: picture *A* and picture *B*. The picture *B* is predicted using the picture A and replaced by the motion-compensated prediction residuals. The prediction residual of the picture *B* is then again motion-compensated, but towards the picture *A*. The obtained motion-compensated prediction residual is added to the picture *A*, so that the picture *A* is replaced by a low-pass version that is effectively obtained by low-pass filtering along the motion information. These processes are iteratively applied to the set of low-pass pictures in each decomposition stage until a single low-pass picture is obtained as key picture [8].

Since the motion vectors for the motion-compensated update steps are derived from the motion information for the prediction steps, no additional motion information needs to be transmitted for the update steps. Therefore, the motion-compensated temporal filtering without update steps is identical to an open loop encoding with the hierarchical picture structure.

Figure 4 illustrates the decomposition process of a group of eight pictures, where levels of temporal scalability with temporal resolution ratios of 1/2, 1/4, and 1/12 are provided. In the synthesis filterbank in the lifting scheme, we reconstruct the encoded video sequence. We first reconstruct L^3 and H^3 pictures. Using the reconstructed pictures, we reconstruct two L^2 pictures. L^1 pictures are obtained by previous reconstructed L^2 and H^2 pictures. We can repeat the reconstruction process until we obtain the highest temporal resolution of the sequence.

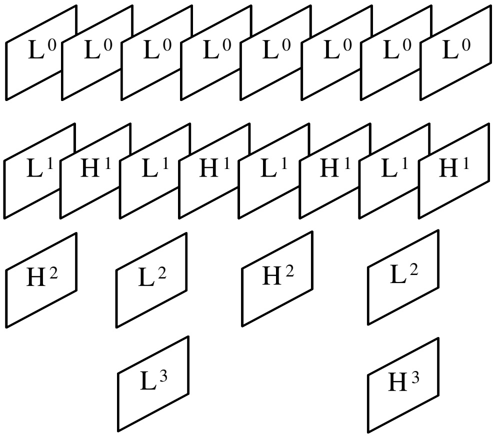


Fig. 4. Temporal Decomposition of a Group of Eight Pictures

3 Fine Granular Scalability

The fine granularity scalability (FGS) algorithm in the current JSVM encodes coefficients within an FGS slice in an order such that “more significant” coefficients are coded first. By arranging the bit stream in this way, the extraction process is biased so that a simple truncation is likely to retain those “more significant” coefficients, and therefore improve the reconstructed quality [8].

When FGS data is truncated, the decoder assumes the missing values to be zero. Consequently, coding zero values into the bit stream contributes nothing to the reconstruction, and coefficients with the greatest probability of being zero should be deferred until the end of the slice. Conversely, coefficients with the greatest probability of being non-zero should be coded first.

In cyclical block coding, unlike subband coding, the current scan position in a given coding pass will differ from one block to another. Furthermore, a correlation was observed between the scan position and probability of the next coefficient being non-zero. The function describing this correlation varies according to quantization parameter and sequence. Therefore, progressive refinement slices using cyclical block coding have been introduced in order to support fine granular quality scalability. A picture is generally represented by a non-scalable base representation, which includes all corresponding motion data as well as a “coarse” approximation of the intra and residual data, and zero or more quality scalable enhancement representations, which represent the residual between the original subband pictures and their reconstructed base representation.

Each enhancement representation contains a refinement signal that corresponds to a bisection of the quantization step size. The refinement signals are directly coded in the transform coefficient domain. Thus, at the decoder side, the inverse transform has to be performed only once for each transform block of a picture.

In order to provide quality enhancement layer NAL units that can be truncated at any arbitrary point, the coding order of transform coefficient levels has been modified for the progressive refinement slices. Instead of scanning the transform coefficients macroblock by macroblock as it is done in "normal" slices, the transform coefficient blocks are scanned in several paths, and in each path only a few coding symbols for a transform coefficient block are coded.

4 Quantization Parameters in JSVM

QPMD is used to determine the best mode for a macroblock and QPBC is used to encode low-pass and high-pass pictures in each decomposition stage. Thus, selection of QPMD is important for coding efficiency. In general, QPMD and QPBC are initialized in the input parameter file and updated in the encoding process [9] [10] [11]. For convenience, we define the following terminology for referencing different QPs.

QPMD: quantization parameters for mode decision (QP_{m1}, QP_{m2}, … QP_{mn});

QPBC: quantization parameters for bit rate control (QP_{H1}, QP_{H2}, … QP_{Hk});

QP_{mn}: QPMD for the n-th decomposition stage;

QP_{Hk}: QPBC for the high-pass pictures in the k-th decomposition stage;

QP_L: QPBC for the low-pass picture;

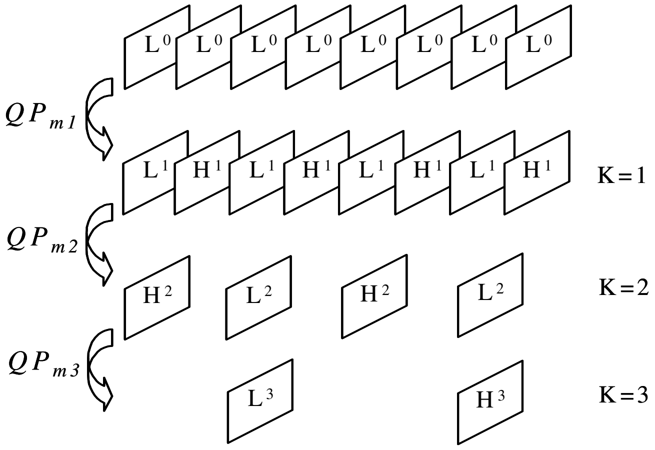


Fig. 5. Quantization Parameters in Each Decomposition Stage

Figure 5 graphically shows the function of QPMD and QPBC in each decomposition stage. For QPMD, QP_{m1}, QP_{m2}, and QP_{m3} are used to determine the best mode and to decompose two low-pass pictures in the previous stage into low-pass and high-pass pictures. For QPBC, the low-pass picture L^3 is quantized by QP_L and high-pass pictures, H^3, H^2, and H^1 are quantized by QP_{H3}, QP_{H2}, and QP_{H1}, respectively. Therefore, QP_L and QP_{Hk} are determined by

$$QP_L = \max(0, \min(51, Round(QP - 6\frac{\log_{10} SF_L}{\log_{10} 2}))) \quad (1)$$

$$QP_{Hk} = \max(0, \min(51, Round(QP - 6\frac{\log_{10} SF_{Hk}}{\log_{10} 2}))) \quad (2)$$

where SF_L is the scaling factor for low-pass picture. SF_{Hk} is the scaling factor for high-pass pictures in the k-th decomposition stage. The operator Round ( ) specifies rounding to the nearest integral number. QP represents the initialised quantization parameter in the input file. QPMD for the k-th decomposition stage, QP_{mn}, is obtained by

$$QP_{mn} = \max(0, \min(51, Round(QP - 6\frac{\log_{10} SF_{mn}}{\log_{10} 2}))) \quad (3)$$

where SF_{mn} is the scaling factor for quantization parameters for mode decision in the n-th decomposition stage. QP represents the initial quantization parameter for each decomposition stage. We have used the same QP for each decomposition stage.

5 Optimum Quantization Parameters for Mode Decision

In this section, we propose a new method for selecting optimum quantization parameters for mode decision (OQPMD). Proposed method is based on quality (SNR) scalability depicted in Figure 6.

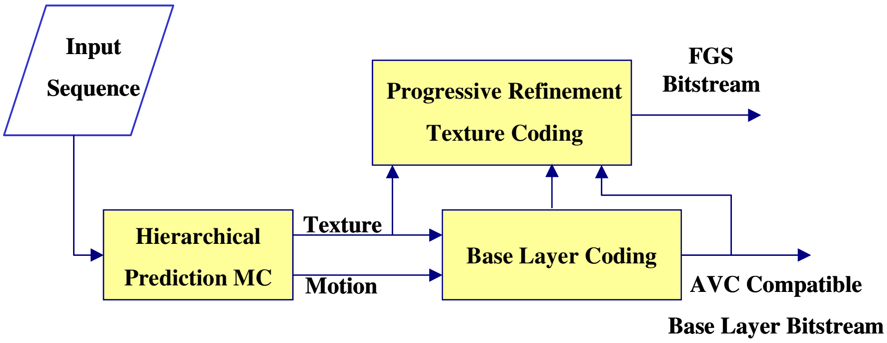

Input Sequence
Progressive Refinement Texture Coding
FGS Bitstream
Hierarchical Prediction MC
Texture
Motion
Base Layer Coding
AVC Compatible Base Layer Bitstream

Fig. 6. Basic Structure for SNR Scalability

In the non-scalable video codec, such as the H.264/AVC video coding standard, quantization parameters for mode decision and bit rate control are the same. Hence, no error is caused by the mismatch between QPMD and QPBC. In JSVM using the motion-compensated temporal filtering technique, we should set QPMD and QPBC in each decomposition stage. If decomposition stages are independent of each other, each QPMD and QPBC would be set to the same value in the given decomposition stage. However, decomposition stages in MCTF have the hierarchical structure, as shown in Figure 4. Therefore, when we select QPMD in each decomposition stage, we need to consider the relationship between stages.

In order to find OQPMD, we first consider that decomposition processes in MCTF are implemented hierarchically and design a metric to measure errors caused by the mismatch between QPMD and QPBC in each decomposition stage. Based on this fact, we find the optimum scaling factor SF_{mn} in each stage. The errors caused by the mismatch between QP_{m1} and QP_{H1} are represented by

$$E = \alpha_1 (QP_{m1} - QP_{H1}) \tag{4}$$

where α_1 represents the importance of the high-pass picture H_1. Since the decomposition process in MCTF has a hierarchical structure, the mode decision in the first stage influences pictures in the following stages, such as H_1, H_2, … H_L, and L-pictures. Therefore, in order to find the total errors caused by QP_{m1}, we consider the mismatch among QP_{H1}, QP_{H2}, … QP_{HL}, and QP_L. Hence, the total errors E_1 caused by QP_{m1} are represented by

$$E_1 = \sum_{k=1}^{L} \alpha_k (QP_{m1} - QP_{Hk}) + \alpha_L (QP_{m1} - QP_L) \tag{5}$$

where α_k and α_L represent the weighting factor of the high-pass picture in the k-th decomposition and the low-pass picture, respectively. For simplicity, these weighting factors are regarded as the corresponding scaling factors (SF_{Hk}, SF_L) of pictures in the decomposition stage. Based on Eq. (5), we can derive the total error E_t by

$$E_t = \sum_{k=1}^{L} \sum_{n=1}^{L} \alpha_k (QP_{mn} - QP_{Hk}) + \alpha_L (QP_{mn} - QP_L) \tag{6}$$

where L is the total number of stages. From Eq. (1), Eq. (2) and Eq. (3), the total error E_t is adjusted by controlling SF_{mn}. For the JSVM reference software, scaling factors in decomposition stages are listed in Table 1. As shown in Table 1, there is a large difference between QP_{m0} and QP_L, and difference reduces coding efficiency.

Table 1. Scaling Factors in Each Decomposition Stage

QP_m		QP_r	
Scaling factor (SF_{mn})	QP_{mn}	Scaling factor (SF_{Hk})	QP_{Hk}
0.847791 (SF_{m1})	44(QP_{m1})	0.7915 (SF_{H1})	44(QP_{H1})
1.009482 (SF_{m2})	42(QP_{m2})	0.942 (SF_{H2})	42(QP_{H2})
1.18386 (SF_{m3})	41(QP_{m3})	1.105 (SF_{H3})	41(QP_{H3})

* Scaling factor for low-pass frame (SF_L= 1.6062), QP_L=38, GOP=8

Therefore, we replace SF_{mn} by a weighted scaling factor WSF_{mn}.

$$WSF_{mn} = W_n \cdot SF_{mn} \tag{7}$$

where W_n represents the weighing factor of SF_{mn} which is determined by minimizing the total error E_t in Eq. (6). For simplicity, in our experiments, the weighing factors W_1, W_2, and W_3 are fixed to 2.5, 1.5, and 0.7, respectively.

In Figure 7, for quality (SNR) scalability, the difference between the original and reconstructed pictures in base layer is encoded with finer quantization parameters in the enhancement layer. For quality scalability, coding efficiency of the enhancement layer is much less than that of the base layer. From our intuition and extensive experiments, we find that the statistical distribution of the residual data is determined by

QPMD. Hence, coding efficiency in the base and enhancement layers can be adaptively controlled by QPMD.

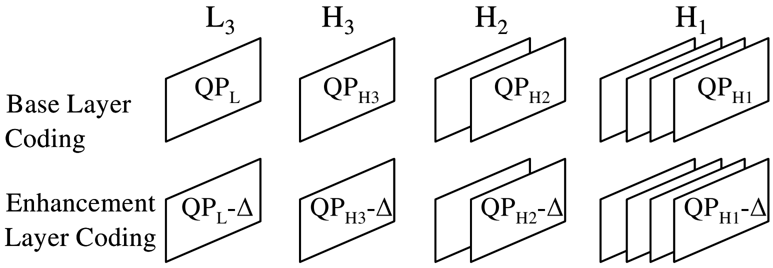


Fig. 7. Adjustment of Quantization Parameters for the Enhancement Layer

We also propose an offset quantization parameter Δ_m for mode decision to enhance coding efficiency of the enhancement layer. Coding efficiency in the base and enhancement layers is controlled by the proposed offset quantization parameter Δ_m for mode decision by which each QP_{mn} is equally shifted. Therefore, the optimal QPMD (OQPMD) is represented by

$$OQPMD = QP_{mn} - \Delta_m \tag{8}$$

In Table 2, we compare the rate-distortion performance for various Δ_m which influences coding efficiency in the enhancement layer. We can regard the current JSVM reference software as the case of the offset value 0. Since the mode decision is only adjusted by quantization parameters for the base layer coding (QP_{H1}, QP_{H2}, … QP_{HL}, and QP_L), the case of offset value 0 (JSVM) provides the best coding efficiency in the base layer but poor coding efficiency in the enhancement layer.

Table 2. Quantization Parameter Offset for Mode Decision

Offset Δ_m= 0		Offset Δ_m= -4		Offset Δ_m= 4	
Bit rate	PSNR	Bit rate	PSNR	Bit rate	PSNR
243(Base)	35.68	254(Base)	35.88	247(Base)	35.37
432	37.25	443	37.10	431	37.25
505	37.84	510	37.56	504	38.13
740	38.93	796	38.99	751	39.11
1117	40.19	1131	39.85	1119	40.82

If the offset value is four, though coding efficiency in the base layer is poor, coding efficiency in the enhancement layer is better than JSVM (Δ_m=0). By adjusting the offset value Δ_m, we can find the optimum offset value Δ_m without little reducing coding efficiency in the base layer. In our experiments, we use the offset value 2, which reduces coding efficiency but little in the base layer. According to the channel variation, we can adjust the offset value properly.

6 Experimental Results

In order to evaluate the efficiency of the proposed weighting factors and QP offset, we have implemented the OQPMD scheme in the JSVM reference software version 1.0. We have fixed the QP offset value to two and the weighing factors W_1, W_2, and W_3 to 2.5, 1.5, and 0.7, respectively. These values can be selected adaptively if we know channel conditions. We have used “FOREMAN” and “MOBILE” sequences of 352×288 pixels (CIF format). Table 3 lists encoding parameters in our experiment.

Table 3. Encoding Parameter

GOP Size	8	Base Layer QP	42
Resolution	352×288	Spatial Layers	1
Frame Rate	30	FGS Layers	0.5, 1, 1.5, 2

In Table 4 and Table 5, we compare PSNR values for different sizes of FGS layers. From Table 4 and Table 5, we observe that the proposed method provides slightly lower coding efficiency in the base layer, but much higher coding efficiency in the enhancement layer.

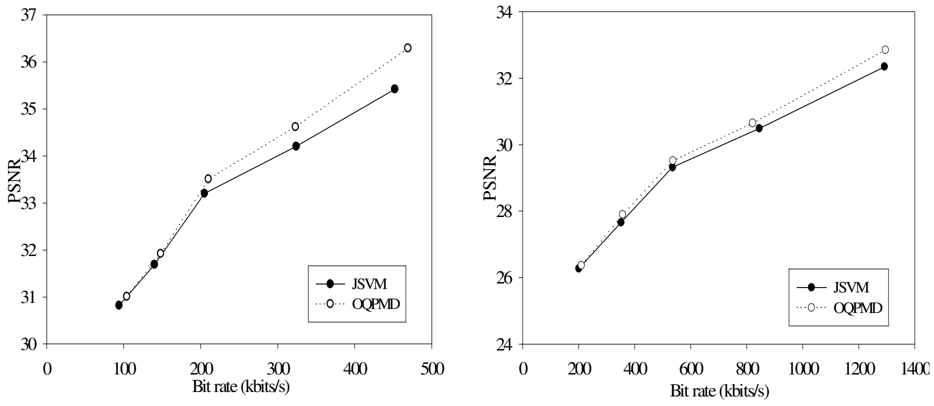
Table 4. PSNR Values Comparison (“FOREMAN”)

	JSVM		Proposed Method	
FGS Layer	Bit rate (kbps)	PSNR	Bit rate (kbps)	PSNR
0	94	30.82	104	31.01
0.5	140	31.69	148	31.92
1	205	33.20	210	33.51
1.5	324	34.20	323	34.62
2	452	35.42	469	36.29

Table 5. PSNR Values Comparison (“MOBILE”)

	JSVM		Proposed Method	
FGS Layer	Bit rate (kbps)	PSNR	Bit rate (kbps)	PSNR
0	200	26.27	209	26.37
0.5	351	27.66	357	27.79
1	535	29.32	536	29.52
1.5	845	30.49	821	30.65
2	1292	32.35	1295	32.85

Figure 8 displays rate-distortion curves for “MOBILE” and “FOREMAN” sequences. These curves have been obtained by changing the size of the FGS layer size by the unit of 0.5. For “FOREMAN” sequence, the PSNR value is enhanced up to 0.8dB. As the number of FGS layers increases, we note that the proposed method provides higher coding efficiency than JSVM.



(a) FOREMAN (b) MOBILE

Fig. 8. PSNR Values for FOREMAN and MOBILE Sequences

7 Conclusions

In this paper, we proposed a new method to select optimum quantization parameters for mode decision (OQPMD) in order to improve coding efficiency for the base and enhancement layers. For the base layer, we proposed the optimum scaling factors for OQPMD in each decomposition stage. In order to find the optimum scaling factors, we designed a new metric to measure the error caused by the mismatch between quantization parameters for mode decision and bit rate control. We also proposed an offset quantization parameter for OQPMD to enhance coding efficiency in the enhancement layer. Using the proposed offset quantization parameter, we can efficiently control the coding efficiency in the base and enhancement layers. The proposed OQPMD is expected the efficiency in the video transmission under abruptly varying channel capacity. Experimental results represent that the proposed method increases the average PSNR value up to 0.8dB.

Acknowledgements. This work was supported in part by Gwangju Institute of Science and Technology (GIST), in part by the Ministry of Information and Communication (MIC) through the Realistic Broadcasting Research Center (RBRC), and in part by the Ministry of Education (MOE) through the Brain Korea 21 (BK21) project.

References

1. Schwarz, H., Hinz, T., Kirchhoffer, H., Marpe, D., and Wiegand, T.: Technical Description of the HHI proposal for SVC CE1: ISO/IEC JTC1/SC29/WG11, Document M11244, Palma de Mallorca, Spain, Oct. (2004)
2. ISO/IEC JTC1/SC29/WG11.: Scalable Video Model 3.0: ISO/IEC JTC1/SC29/WG11, Doc. N6716, Palma de Mallorca, Spain, Oct. (2004)
3. ITU-T Recommendation H.264 & ISO/IEC 14496-10 AVC.: Advanced Video Coding for Generic Audiovisual Services: version 3, (2005)

4. Joint Video Team of ITU-T VCEG and ISO/IEC MPEG.: Scalable Video Coding Working Draft 1: Joint Video Team, Document JVT-N020, Jan. (2005)
5. Joint Video Team of ITU-T VCEG and ISO/IEC MPEG.: Joint Scalable Video Model JSVM0: Joint Video Team, Document JVT-N021, Jan. (2005)
6. Taubman, D.: Successive Refinement of Video: Fundamental Issues, Past Efforts and New Directions. Proc. SPIE, Visual Communication and Image Processing (2003) 649-663
7. Schwarz, H., Marpe, D., and Wiegand, T.: Scalable Extension of H.264/AVC. ISO/IEC JTC1/WG11 Doc. M10569/SO3 (2004)
8. Scalable Extension of H.264/AVC: http://ip.hhi.de/imagecom_G1/ savce
9. Flierl, M., Girod, B.: Video Coding with Motion-Compensated Wavelet Transforms. Proc. Picture Coding Symposium (2003) 59-62
10. Wiegand, T., Schwarz, H., Joch, A., Kossentini, F., and Sullivan, G.: Rate-Constrained Coder Control and Comparison of Video Coding Sandards. IEEE Trans. on Circuit and System for Video Technology (2003) 688-703
11. ISO/IEC JTC1: Requirements and Application for Scalable Video Coding. ISO/IEC JTC1/ WG11 Doc. N6025, Oct. (2003)