Inter-layer Texture and Syntax Prediction for Scalable Video Coding

Woong Lim, Hyomin Choi, Junghak Nam, and Donggyu Sim

Department of Computer Engineering, Kwangwoon University, 447-1, Wolgye-dong, Nowon-gu, Seoul - Korea
{limwoong, hyomin06, qejixfyza, dgsim}@kw.ac.kr

***** Corresponding Author: Donggyu Sim

Received December 16, 2015; Accepted December 24, 2015; Published December 31, 2015

** Regular Paper*

Abstract**:** In this paper, we demonstrate inter-layer prediction tools for scalable video coders. The proposed scalable coder is designed to support not only spatial, quality and temporal scalabilities, but also view scalability. In addition, we propose quad-tree inter-layer prediction tools to improve coding efficiency at enhancement layers. The proposed inter-layer prediction tools generate texture prediction signal with exploiting texture, syntaxes, and residual information from a reference layer. Furthermore, the tools can be used with inter and intra prediction blocks within a large coding unit. The proposed framework guarantees the rate distortion performance for a base layer because it does not have any compulsion such as constraint intra prediction. According to experiments, the framework supports the spatial scalable functionality with about 18.6%, 18.5% and 25.2% overhead bits against to the single layer coding. The proposed inter-layer prediction tool in multi-loop decoding design framework enables to achieve coding gains of 14.0%, 5.1%, and 12.1% in BD-Bitrate at the enhancement layer, compared to a single layer HEVC for all-intra, low-delay, and random access cases, respectively. For the single-loop decoding design, the proposed quad-tree inter-layer prediction can achieve 14.0%, 3.7%, and 9.8% bit saving.

Keywords**:** HEVC, Scalable coding, Scalable video, Inter-layer texture prediction, Inter-layer syntax prediction

1. Introduction

High definition television (HDTV) has become common as a home appliance due to the rapid development of multimedia and network technologies. Recently, there have been demands for even higher video resolutions, such as ultra-definition television (UDTV). Although MPEG-2 and H.264/AVC have been widely employed in many current video applications, there is a need for a new compression technology to support UDTV content. This new compression technology is required to provide significant and better coding efficiency for UDTV-based services with a limited bandwidth. Recently, the joint collaborative team on video coding (JCT-VC) was formed to create a new coding standard called High Efficiency Video Coding (HEVC). Its main goal is to achieve around two times rate reduction for the same video quality compared to H.264/AVC. We can expect it to be employed in many new commercial products. It is also reasonable to expect that many of these new applications will involve heterogeneous users in terms of device capabilities, resolutions, and access bandwidth, and also will be extended to support various forms of scalability [1, 2]. In this paper, one of the feasible scalable video coding frameworks is proposed. The proposed framework is designed by not only making full utilization of the single layer capabilities, but also employing proposed quad-tree inter-layer prediction tools to improve rate distortion (RD) performance at enhancement layers.

In this paper, we propose a foundational framework of future scalable video coding for any single layer coder. The scalable video coder supports not only spatial, quality, and temporal scalabilities, but also view scalability [3] within a flexible framework. The proposed framework alternatively works with single-loop and multi-loop decoding design depending on applications or heterogeneous video environment. To improve coding efficiency of the proposed SVC framework, the framework needs also inter-layer prediction tools. The proposed quad-tree inter-layer prediction tools alternatively work depending on the

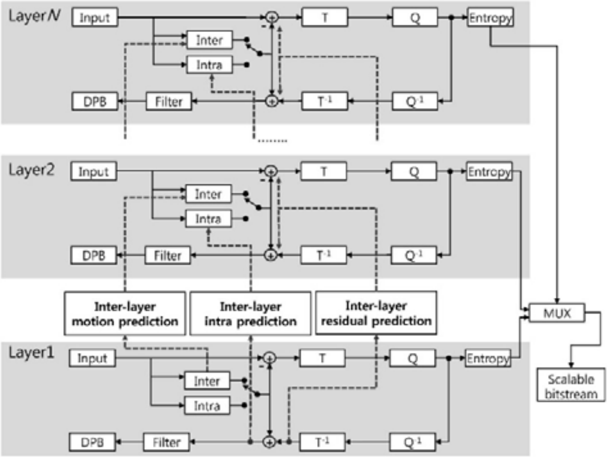

Layer*N*
Input
Inter
Intra
T
Q
Entropy
DPB
Filter
T-1
Q-1
Layer2
Layer1
Inter-layer motion prediction
Inter-layer intra prediction
Inter-layer residual prediction
MUX
Scalable bitstream

Fig. 1. H.264/SVC encoder block diagram.

decoding loop design. In addition, the proposed framework with the proposed inter-layer prediction tools does not enforce the constraint intra prediction tools at the base layer. Therefore, it guarantees not to degrade RD performance of the base layer. Furthermore, extra de-blocking filter and padding algorithm are not required.

The remaining parts of this paper are as follow. In Section 2, inter-layer prediction methods of the conventional H.264/SVC are presented, and HEVC features are briefly introduced. In Section 3, the proposed framework with quad-tree inter-layer prediction tools are presented. In Section 4, the proposed framework with single-loop and multi-loop decoding design is evaluated on diverse coding conditions and performance is summarized. Finally, we conclude and give suggestions for further work in Section 5.

2. Scalable Eextension of H.264/AVC and High Efficiency Video Coding

In this section, we review the basic structures of the scalable video coding extension of H.264/AVC [4-7]. This standard technology employs three main coding tools in order to eliminate redundancy between layers. These three main coding tools, inter-layer intra prediction, inter-layer motion prediction, and inter-layer residual prediction, are employed in H.264/SVC in a single-loop design.

2.1 Basic Structure of Scalable Video Coding

Fig. 1 shows a block diagram of H.264/SVC. It allows eight different layers for combined spatial and quality scalability. Therefore, N in Fig. 1 can be up to eight. The coding structure of each layer is identical to that of H.264/AVC, except for several inter-layer prediction tools. In H.264/SVC, the inter-layer prediction is always conducted for sizes of MB (Macroblcok) at the enhancement layer. For an MB predicted by the inter-layer intra prediction needs to send only the residual data without many other syntax elements. Thus, it gives as much as 9.98% bit-saving at the enhancement layer [8]. However, it is not employed for inter slices because the H.264/SVC was developed based on a single-loop design, so it does not reconstruct texture signals for inter slices in the reference layer. To maximize the RD performance at the enhancement layer with the inter-layer intra prediction, H.264/SVC compulsively employs the constraint intra prediction for a base layer. As employing the constraint tool, intra blocks in inter slices of the base layer can be fully reconstructed. Whereas, RD performance of the base layer is degraded because the predicted blocks by the constraint prediction tool generate many residual coefficient signal. Therefore, it affects the overall RD performance of scalable video coding. Furthermore, extra de-blocking filter and padding algorithm are required only for the blocks. It also increases decoding complexity and is considerable burden in hardware implementation.

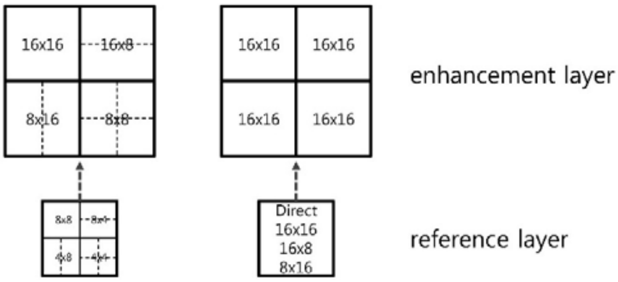


Fig. 2. Block partitioning for inter-layer motion prediction.

As mentioned before, the texture information for the inter-coded blocks of the reference layer is unavailable for the enhancement layer coding in the single-loop design. Inter-layer prediction for inter slices of the enhancement layer can be conducted using the syntax elements of the reference layer instead of the reconstructed texture. In particular, motion vectors of the reference layer are known to be effective for enhancement layer prediction. This prediction method is called the inter-layer motion prediction. The motion vectors and reference index are computed depending on the block partitioning information. Then, the computed motion vectors of the reference layer are used at the enhancement layer for prediction of an MB

Fig. 2 shows block partitions for the inter-layer motion prediction. In this case, we can save a few bits by not encoding the block partition information and motion information. This inter-layer motion prediction is known to save around 8% in BD-bitrate [9, 10]. Along with the inter-layer intra and motion prediction, H.264/SVC also employs inter-layer residual prediction. Here, the residual of the enhancement layer is predicted by that of the corresponding block in the reference layer. The residual signals from the reference layer are interpolated by using the pixel-wise bi-linear filter.

2.2 High Efficiency Video Coding

HEVC standardization effort has been kicked off in Apr. 2010. Working draft for HEVC has been released with reference software namely HEVC test model (HM). It

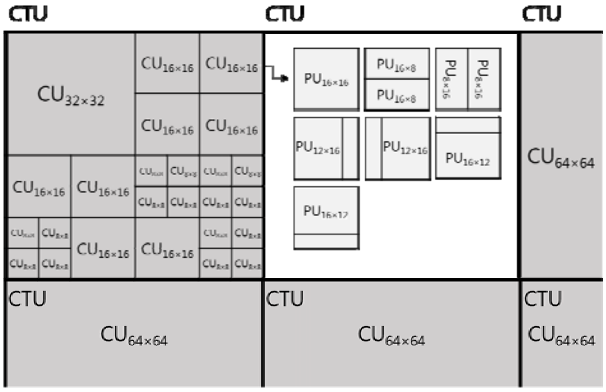


Fig. 3. Diverse CU sizes and PUs in a slice.

also consists of prediction, transformation, in-loop filters, and entropy coder, much like conventional video coders. However, it employs more diverse block sizes, instead of just an MB (16×16 pixels) of H.264/AVC and earlier standards. The diverse coding block that corresponds to MB is called a coding unit (CU). The size of a CU ranges from 8×8 to 64×64, where the largest CU, called CTU is set to 64×64. The CU structure is based on a quad-tree representation, and each CU can have several prediction units (PUs) which is a unit block of intra or inter prediction with diverse sizes and shapes. Moreover, advanced motion vector prediction (AMVP) with motion vector 'competition' and 'merging' is adopted for more efficient compression of motion vectors. For residual coding, residuals of a PU can be divided in a quad-tree fashion, and each block is transformed. The transform block sizes and shapes can be determined based on RD optimization in the encoder. Transform coefficients are quantized with the uniform quantizer. The reconstructed blocks are able to be filtered using in-loop filters.

Fig. 3 shows diverse CU and PU realizations in a slice, when a CTU size is 64×64. A slice is divided into multiple CTUs and each CTU is again divided into multiple CUs. All the CUs in a CTU are hierarchically represented in a quad-tree fashion. These larger and diverse partitions of a CTU yield significant coding gain in RD performance with consolidated compact syntax, prediction, and transformation for larger areas compared to the conventional H.264/AVC. Sizes of CU block can be derived from quad-tree split flags. As shown in Fig. 3, a CU can be coded using several prediction units (PU). The CU can have eight PU types: 2N×2N, 2N×N, N×2N, N×N and four asymmetric shapes. However, the N×N shape is allowed only for the smallest CU. The residual signal in the CU can be coded using discrete cosine transform (DCT) in separate quad-tree fashion. HEVC support transform sizes from 32×32 to 4×4 [11].

3. The Proposed Scalable Video Coding

We propose a foundational framework of future scalable video coding for the diverse scalable applications. In addition, we present the proposed quad-tree inter-layer prediction in order to improve rate distortion (RD) performance at enhancement layers.

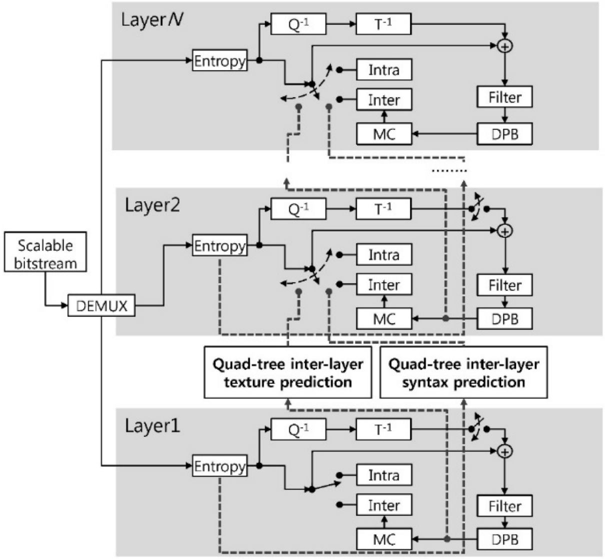


Fig. 4. Block diagram of the proposed scalable framework.

Fig. 4 shows the feasible proposed scalable video coding framework. The proposed framework supports not only spatial, temporal, and quality scalability, but also view scalability as employing alternative two decoding loop designs. To support view scalability, the framework conducts scalable coding with multi-loop decoding design. For multi-loop decoding, switches which connect adding of inverse transformed pixels and prediction pixels should be on for lower layers in Fig. 4. Therefore, all the layers are fully reconstructed and also redundant data between layers are eliminated by proposed quad-tree inter-layer predictions. The proposed quad-tree inter-layer predictions also work for the framework with single-loop decoding design. For single-loop decoding design, the switches should be off. Depending on the applications and heterogeneous video environments, the framework alternatively choose the decoding loop design, which is signaled by high level syntax such as sequence parameter set (SPS). The proposed quad-tree inter-layer prediction consists of two different prediction methods, which are quad-tree inter-layer texture prediction (Q-ILTP) and quad-tree inter-layer syntax prediction (Q-ILSP). The proposed tools refer to reconstructed signals, syntaxes, and residuals depending on the decoding loop design, slice type of the reference layer and the prediction mode of corresponding block in the reference layer. For a framework with single-loop decoding design, the proposed Q-ILTP works along with the reconstructed textures of a reference layer when the slice of the reference layer is intra type. When the slice type is not an intra-slice, the proposed Q-ILSP exploits syntaxes from a reference layer instead of the reconstructed texture. Therefore, we do not enforce to use constraint intra prediction at the base layer, so the proposed framework does not affect RD performance for the base layer. In addition, extra de-blocking filter and padding algorithm are not required at all.

3.1 Quad-tree Inter-layer Texture Prediction (Q-ILTP)

Reconstructed textures from a reference layer contain relatively abundant information for the enhancement layer, compared to the neighboring pixels. Therefore, the texture prediction blocks are generally more efficient compared to the predicted blocks by the intra prediction within the same layer [12-17]. Note that the inter-layer intra prediction in H.264/SVC works with an MB, whose size is fixed at 16×16. The proposed Q-ILTP generates texture prediction blocks for a large coding unit of the enhancement layer with reconstructed textures of the reference layer. The textures are up-sampled with regard to the spatial aspect ratio between the enhancement layer and its reference layer. The predicted block with a large coding unit is exploited partly with quad-tree fashion in the large coding unit. The conventional intra prediction and inter prediction used in the intra-layer coding can also work adaptively in conjunction with the proposed Q-ILTP with the additional flag bit. This additional flag bit indicates whether a coding unit is predicted by the conventional intra, inter prediction or by the proposed Q-ILTP. Alternatively, the proposed Q-ILTP can be used for all the coding units without the conventional intra and inter prediction. In this case, we do not need an additional flag bit for each coding unit. At the encoder side, the best mode which has the least RD cost is competitively selected the conventional intra/inter coding and the proposed Q-ILTP during Rate-Distortion Optimization (RDO) stage. In the RDO stage, the encoder considers not only the distortion but also the amount of bits for encoding parameters such as quantization parameter (QP), coding mode and etc. By using those coding parameters (P_{enc}), the encoder calculates the *RDcost* as

$$RDcost = D(P_{enc}) + \lambda R(P_{enc}) \qquad (1)$$

where D and R are the distortion between the original input signal and the reconstructed signal and the amount of bits consumed for encoding, respectively. λ is used as Lagrange multiplier for Lagrangian optimization. As a result, the encoder decides the optimal coding parameters which minimize the *RDcost*.

Partial coding unit block predicted by the proposed Q-ILTP can be as,

$$P_{Q-TP,(E)}^{2N\times 2N}(x,y) = U_8\left(T_{(R)}^{\frac{2N}{S_x}\times\frac{2N}{S_{yx}}}\left(\left\lfloor \frac{x}{S_x} \right\rfloor, \left\lfloor \frac{y}{S_y} \right\rfloor \right)\right) \qquad (2)$$

where, Q-TP indicates the block P is predicted by the proposed quad-tree inter-layer texture prediction (Q-ILTP). Subscript (E) presents that the block P is at the enhancement layer with size of superscript $2N \times 2N$. When we assume that sizes of the coding unit and smallerst coding unit are 64×64 and 8×8, respectively, the N can vary from 32, 16, 8, and 4. U8(·) represents the DCT-IF eight-tap interpolation filter. The texture T in the reference layer (R) is up-sampled by using U_8 function. The quad-tree partitioning is determined based on RD optimization at each coding unit level. Fig. 5(a) denotes one reconstructed CTU in a reference layer. Note that reconstructed CTUs are filtered using in-loop filters, which are deblocking filter and sample adaptive offset (SAO), sequentially.

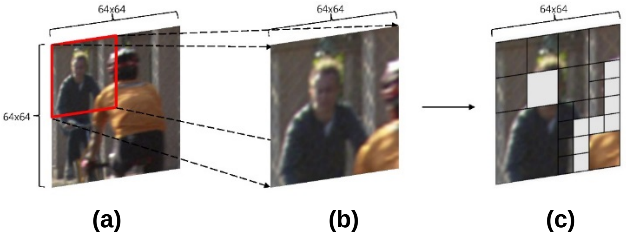


Fig. 5. Proposed quad-tree inter-layer texture prediction (a) Reconstructed CTU in a reference layer, (b) Up-sampled CTU from the reference layer, (c) Quad-tree blocks used for Q-ILTP.

3.2 Quad-tree Inter-layer Syntax Prediction (Q-ILSP)

Reconstructing reference layer is necessary to accomplish texture prediction. However, inter-slice is not reconstructed on single-loop decoding design because motion compensation (MC) is available only for a target layer. Therefore, texture prediction with reconstructed signals is valid when a corresponding slice type of the reference layer is intra slice. For inter slice, texture prediction with the proposed Q-ILSP should be accomplished by the other scheme. According to a lot of aforementioned studies for the inter-layer prediction, syntaxes such as motion information, block partitioning, and so on, in the enhancement layer, are highly correlated with those in a reference layer. Therefore, we try to generate texture signals using motion syntax elements from a reference layer, so the texture signal can remove much redundancy between the two layers, especially in low frequency components. In addition, we employ intra prediction mode syntaxes of the corresponding block in the reference layer. As a result, we could avoid enforcedly employing the constraint intra prediction. The intra prediction modes from the reference layer are exploited in conducting basic intra prediction method with predicted pixels by motion syntaxes within a large coding unit. It does not require any additional tools for the process.

Fig. 6 shows how to generate the predicted texture signals for a coding unit size with several syntax elements of the reference layer. Fig. 6(a) shows a block structure for an coding unit in the reference layer. We can derive all the syntax elements from those in the corresponding block (32 × 32) of the reference layer with proper scaling. With the derived syntaxes, we can obtain the predicted coding unit using motion compensation and then intra prediction with the derived intra mode syntaxes is followed. As a coding unit should be predicted using motion and intra prediction together, where intra prediction is conducted using the blocks that are predicted by motion compensation. The predicted texture block P by motion and intra

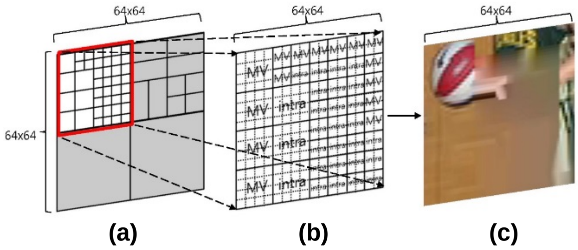


Fig. 6. Texture prediction blocks with motion syntax for the proposed Q-ILSP (a) Coding structure in the reference layer, (b) Derived syntaxes for a large coding unit in the enhancement layer, (c) Predicted texture with the derived syntaxes for a coding unit.

mode syntaxes elements with size of the coding unit can be expressed as

$$P^{LCU}_{SE,(E)}(x,y)=\begin{cases} MC_{(E)}\left(\begin{array}{l} U_s\left(MV^{x}_{(R)}\left(\left\lfloor \frac{x}{S_x}\right\rfloor,\left\lfloor \frac{y}{S_y}\right\rfloor\right)\right), \\ U_s\left(MV^{y}_{(R)}\left(\left\lfloor \frac{x}{S_x}\right\rfloor,\left\lfloor \frac{y}{S_y}\right\rfloor\right)\right) \end{array}\right), \\ \quad if\ \ BCM_{(R)}(x,y)=='INTER' \\ INP_{(E)}\left(IPM_{(R)}\left(\left\lfloor \frac{x}{S_x}\right\rfloor,\left\lfloor \frac{y}{S_y}\right\rfloor\right)\right), \\ \quad if\ \ BCM_{(R)}(x,y,r)=='INTRA' \end{cases} \tag{2}$$

where subscript *SE* of the prediction block *P* denotes that the texture signal of the block is generated by syntax elements, which are motion and intra mode syntaxes. The block *P* has the size of a coding unit at the enhancement layer (*E*) as remarked superscript. The motion vector from a reference layer is up-scaled by $U_s(\cdot)$ operation regarding the resolution ratios S_x and S_y. *BCM* represents the block coding mode either inter or intra mode in the reference layer (*R*) and *IPM* is an intra prediction mode of the corresponding block in the reference layer. *IP* $(\cdot)$ is an intra prediction operation, which is done using the neighboring pixels in the enhancement layer (*E*). However, the accuracy of the texture predicted by motion and intra mode syntax elements can be worse compared to that with reconstructed textures, when temporal redundancy is not very high. In order to compensate the remaining redundant data, residual signal of corresponding block is added to the predicted texture signal for more accurately predicted texture signals. Fig. 7(a) shows the residual signal for a coding unit of the reference layer. The up-sampled residual is computed through bilinear interpolation with a proper scaling, as shown in Fig. 7(b). In this example, the top left 32×32 residual block of the reference layer is interpolated by a factor of two. The up-sampled residual is added to the texture blocks with syntax elements and the added signals can be a final predicted texture for a coding unit at the enhancement layer.

In conclusion, the proposed Q-ILSP generates texture prediction blocks for the size of a coding unit at the enhancement layer with motion, intra mode and residual signal syntaxes from a reference layer. Then, the predicted blocks are partly exploited with quad-tree fashion as same as Q-ILTP. Note that the proposed Q-ILSP also supports the mixed predictions, which are basic intra and inter prediction, with itself within a coding unit.

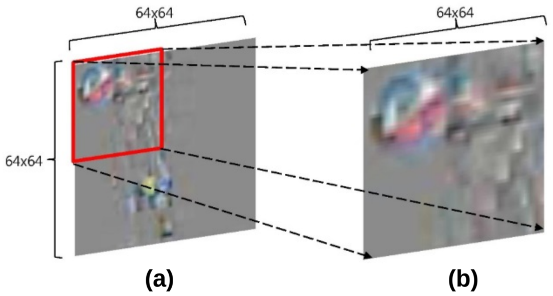


Fig. 7. Residual signals for the proposed Q-ILSP (a) Residual in a coding unit of the reference layer, (b) Up-sampled residual for a coding unit in the enhancement layer.

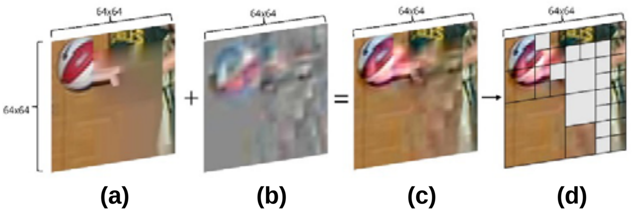


Fig. 8. Proposed quad-tree inter-layer syntax prediction (a) Predicted texture with the derived syntaxes for a coding unit, (b) Up-sampled residual for a coding unit, (c) Final texture signal for a coding unit, (d) Quad-tree blocks used for Q-ILSP.

Fig. 8 shows a process to get the final partly exploited texture blocks and how to the blocks are used by the proposed Q-ILSP with quad-tree fashion. Fig. 8(a) and Fig. 8(b) are described by Fig. 6(c) and Fig. 7(b), respectively. The final texture block for a coding unit size of the enhancement layer is shown in Fig. 8(c). The predicted texture signals can be divided into multiple square blocks in the quad-tree fashion, as shown in Fig. 8(d). Regardless of the partitioning shape in the reference layer, the predicted block can have one of the diverse block sizes from 64×64 to 8×8 in a coding unit with the quad-tree representation. Each partly exploited texture blocks by the proposed Q-ILSP is computed by

$$P^{2N\times 2N}_{Q-SP,(E)}(x,y)=P^{2N\times 2N}_{SE,(E)}(x,y)+U_2\left(D^{\frac{2N}{S_x}\times\frac{2N}{S_{yx}}}_{(R)}\left(\left\lfloor \frac{x}{S_x}\right\rfloor,\left\lfloor \frac{y}{S_y}\right\rfloor\right)\right) \tag{3}$$

where, *Q-SP* indicates the block *P* is predicted by the proposed quad-tree inter-layer syntax prediction (Q-ILSP). Subscript (*E*) presents that the block *P* is at the enhancement layer with size of superscript 2*N*×2*N*. When we assume that sizes of coding unit and smallest coding unit are 64×64 and 8×8, respectively, the N can vary from 32 to 4. The block is texture signal derived by formula (2) and it is exploited by $2N \times 2N$ size. To get the residual signal D from a reference layer (R), block position is computed by x and y with S_x and S_y. The residual signal from the corresponding block of the reference is up-sampled by the bilinear interpolation filter, $U_2(\cdot)$. The quad-tree partitioning is represented with coding unit split flags and it is independent of the derived partitioning shape. In Fig.8 (d), each coding unit has a flag bit which indicates whether the coding unit employs the conventional inter, intra prediction, or the proposed Q-ILSP. The shaded blocks are predicted by the conventional intra or inter prediction. The quad-tree partitioning is determined based on RD optimization. Alternatively, for the reduction in the computational complexity of RD optimization at the encoder, the proposed Q-ILSP can always be used without the indication bits.

4. Experimental Results

The standardization of HEVC scalable coding extension, so-called SHVC, has been finalized in October 2014. The main feature of SHVC is an architecturally simple multi-loop reference index design, i.e. the decoded picture of the base layer is put into the reference picture list of the enhancement layer and referred by the to-be-coded enhancement picture as the same manner of inter-prediction. That is the core coding process is not much different from HEVC single-layer coding (HEVC version 1). In addition, as the reference software of HEVC has been matured and improved, the recent SHVC reference software also achieves higher coding performance based on the improved encoding methods for the enhancement layer. However, the SHVC reference software is only suitable for multi-loop coding as aforementioned. Therefore, in this paper, to evaluate the coding performance of the proposed Q-ILTP and Q-ILSP without any syntax changes for future scalable coding, HM2.0 is one of the suitable platform to implement the proposed method which can support both single-loop and multi-loop coding.

For comparative study, a single-layer coder is employed as an anchor. In our evaluation, the number of spatial layers is set to two, and dyadic resolution change is employed. Input sequences for the reference layer are generated by JSVM ‘DownConverter’. Note that the JSVM is the reference software for H.264/SVC. All the experiments are performed with considering the JCT-VC common test conditions with the common test sequences [18]. In the common test conditions, there are high efficiency (HE) and low complexity (LC) mode. Each coding mode includes all-intra (AI), low-delay (LD), and random access (RA) case. In this evaluation, we employed only the HE mode, and also test on the three different cases of it. The intent of HE is to obtain the best performance for high resolution and high quality video services. Therefore, most of the tools that can contribute to the improvement of the RD performance are enabled in this mode. The coding condition on the AI case specifies that all frames are coded with intra slice and intra mode. The LD case has an intra slice only for beginning of a sequence. Remaining frames are coded with inter slices.

Table 1. Coding conditions for proposed frameworks, HM2.0, and JSVM9.18.

	Proposed&HM2.0	JSVM 9.18
AI case	-	
RA case	Identical GOP size and intra period	
LD case	B slice by GPB	P slice
ME	EPZS	
QP	22, 27, 32, 37 for all layers	

To present RD performance, ‘Bjøntegaard-Delta’ [19] is well known and quite popular measurement. It shows relative average bitrate increment for the same PSNR range or conversely, relative PSNR increment for the same bitrate. BD-Bitrate which denotes relative average bitrate increment for the same PSNR is fair assessment method. However, it is not proper to evaluate the RD performance of scalable video coding because scalable video coding has multi-layer. Therefore, it has two different bitrate and PSNR values. The scalable video coding used to be evaluated as comparing bitrate for all layers of scalable video coding which has PSNR of the enhancement layer against to single-layer. In addition, there is also a method of evaluation by comparing bitrate and PSNR of the enhancement layer with a single-layer to find increases in RD performance from inter-layer prediction tools. Therefore, we briefly define BD-T-Bitrate and BD-E-Bitrate. BD-T-Bitrate compares bitrate for all layers of the proposed framework with PSNR of the highest layer with single-layer coder. BD-E-Bitrate evaluates the bitrate on the same PSNR for the enhancement layer only between the proposed framework and single-layer HEVC. Furthermore, we evaluate the JSVM9.18 test model of H.264/SVC for more relative experiments with similar test conditions. Details of the test conditions for each platform are specified in Table 1. The results of JSVM9.18 are just employed as auxiliary data in RD-curves.

At first, we evaluated the RD performance of the proposed Q-ILTP based on the multi-loop design. For the multi-loop design, only Q-ILTP is needed, since all the frames in a reference layer are fully reconstructed. Table 2 shows RD performance of the proposed framework with proposed Q-ILTP. For all the sequences, the proposed algorithm shows the best RD performance in the AI case. Especially, the proposed algorithm yields a large amount of bit savings for higher resolution sequences, compared to the smaller resolution sequences. According to the BD-T-Bitrate of AI case, the proposed framework can support scalability with just about 18.6% overhead bits against to HEVC single-layer coding. For the LD case, the proposed Q-ILTP works for all slices regardless of slice type at the enhancement layer.

The amount of bit savings ratio in BD-E-Bitrate for the AI case is larger than that for the LD. Although it is available to exploit reconstructed texture signal from a

Table 2. RD performance of the proposed framework with Q-ILTP on the multi-loop decoding design against the single layer HM2.0.

Class	Sequence	All-Intra	
		BD-E-BR (%)	BD-T-BR (%)
B (1920×1080)	Kimono	-32.4	13.6
	ParkScene	-20.3	13.7
	Cactus	-17.5	19.4
	BasketballDrive	-14.5	22.2
	BQTerrace	-8.3	19.2
	Average	-18.6	17.6
C (832×480)	BasketballDrill	-10.3	25.1
	BQMall	-11.8	24.6
	PartyScene	-7.8	18.2
	RaceHorses	-13.0	17.3
	Average	-10.7	21.3
D (416×240)	BasketballPass	-12.0	24.7
	BQSquare	-18.1	4.5
	BlowingBubbles	-8.7	18.3
	RaceHorses	-12.4	22.3
	Average	-12.8	17.4
Average		-14.0	18.6

(a)

Class	Sequence	Low-delay	
		BD-E-BR (%)	BD-T-BR (%)
B (1920×1080)	Kimono	-10.9	31.7
	ParkScene	-4.2	26.6
	Cactus	-6.5	28.4
	BasketballDrive	-8.8	26.8
	BQTerrace	-1.4	17.1
	Average	-6.4	26.1
C (832×480)	BasketballDrill	-8.1	29.6
	BQMall	-4.1	29.6
	PartyScene	-3.9	18.7
	RaceHorses	-5.0	23.6
	Average	-5.3	25.4
D (416×240)	BasketballPass	-6.8	28.5
	BQSquare	-0.5	18.3
	BlowingBubbles	-3.0	20.4
	RaceHorses	-4.5	29.3
	Average	-3.7	24.1
Average		-5.1	25.2

(b)

Class	Sequence	Random access	
		BD-E-BR (%)	BD-T-BR (%)
B (1920×1080)	Kimono	-22.7	19.9
	ParkScene	-13.8	17.2
	Cactus	-20.1	13.0
	BasketballDrive	-13.2	23.0
	BQTerrace	-13.7	5.7
	Average	-16.7	15.7
C (832×480)	BasketballDrill	-16.2	20.2
	BQMall	-10.1	23.5
	PartyScene	-9.3	14.0
	RaceHorses	-9.8	19.6
	Average	-11.3	19.3
D (416×240)	BasketballPass	-11.1	24.4
	BQSquare	-5.9	15.2
	BlowingBubbles	-7.8	16.7
	RaceHorses	-8.7	25.2
	Average	-8.4	20.4
Average		-12.1	18.5

(c)

reference layer at the enhancement layer for both cases, temporal reference frames are more suitable for prediction signals than fully reconstructed textures from the corresponding reference layer for inter-slice in LD case. In LD case, the proposed framework enables to support the spatial scalability with 25.2% overhead bits against to single-layer coding. For the RA case, the hierarchical reference structure in a GOP is employed with periodic intra slices.

BD-E-Bitrate on RA case shows about 12.1% bits saving compared to the single-layer coding. It is larger than BD-E-Bitrate on LD case. In addition, the proposed framework with multi-loop decoding design on RA case is able to support spatial scalability with just 18.5% overhead bits. A reason that RD performance on RA is better than LD case is that the fully reconstructed textures of the spatial reference layer are more frequently used than temporal reference frames for lower temporal level frames in a hierarchical reference structure. Since the temporal reference frames are usually temporally far from the frames that are being coded. Therefore, 0bits saving in BD-E-Bitrate for the RA case are about 7% higher than that for the LD case. Therefore, the proposed Q-ILTP on the proposed framework with multi-loop decoding design saves significant coding gain about 10.4% at the enhancement layer compared to the single-layer coding with HM2.0. In addition, the proposed framework with multi-loop decoding design supports spatial scalability with about 20.7% overhead bits.

Table 3 shows RD performance of the proposed framework on single-loop decoding with quad-tree inter-layer prediction tools. If the slice type in the reference layer is intra slice only, then Q-ILTP is selected. The Q-ILSP is selected when the slice type in the reference layer is an inter slice. For the AI case, the proposed framework with single-loop decoding design shows an identical RD performance with the multi-loop design framework as both the frameworks employ only Q-ILTP for the enhancement layers. For the LD case, all slices are coded as inter-slice

Table 3. RD performance of the proposed framework with Q-ILTP and Q-ILSP on the single-loop against the single layer HM2.0.

Class	Sequence	All-Intra	
		BD-E-BR (%)	BD-T-BR (%)
B (1920×1080)	Kimono	-32.4	13.6
	ParkScene	-20.3	13.7
	Cactus	-17.5	19.4
	BasketballDrive	-14.5	22.2
	BQTerrace	-8.3	19.2
	Average	-18.6	17.6
C (832×480)	BasketballDrill	-10.3	25.1
	BQMall	-11.8	24.6
	PartyScene	-7.8	18.2
	RaceHorses	-13.0	17.3
	Average	-10.7	21.3
D (416×240)	BasketballPass	-12.0	24.7
	BQSquare	-18.1	4.5
	BlowingBubbles	-8.7	18.3
	RaceHorses	-12.4	22.3
	Average	-12.8	17.4
Average		-14.0	18.6

(a)

Class	Sequence	Low-delay	
		BD-E-BR (%)	BD-T-BR (%)
B (1920×1080)	Kimono	-10.7	31.9
	ParkScene	-3.5	27.3
	Cactus	-4.8	30.2
	BasketballDrive	-5.3	30.1
	BQTerrace	-1.6	16.7
	Average	-5.1	27.2
C (832×480)	BasketballDrill	-5.8	32.0
	BQMall	-3.2	30.4
	PartyScene	-2.2	20.4
	RaceHorses	-3.1	25.5
	Average	-3.6	27.1
D (416×240)	BasketballPass	-4.0	31.1
	BQSquare	-0.6	18.1
	BlowingBubbles	-1.9	21.4
	RaceHorses	-3.1	30.7
	Average	-2.4	25.3
Average		-3.7	26.5

(b)

Class	Sequence	Random access	
		BD-E-BR (%)	BD-T-BR (%)
B (1920×1080)	Kimono	-19.5	23.0
	ParkScene	-12.4	18.6
	Cactus	-16.9	16.1
	BasketballDrive	-8.9	27.1
	BQTerrace	-13.6	5.7
	Average	-14.3	18.1
C (832×480)	BasketballDrill	-12.4	24.2
	BQMall	-8.1	25.6
	PartyScene	-7.4	15.9
	RaceHorses	-7.3	21.9
	Average	-8.8	21.9
D (416×240)	BasketballPass	-6.7	28.7
	BQSquare	-5.6	15.5
	BlowingBubbles	-6.9	17.6
	RaceHorses	-6.5	27.4
	Average	-6.4	22.3
Average		-9.8	20.7

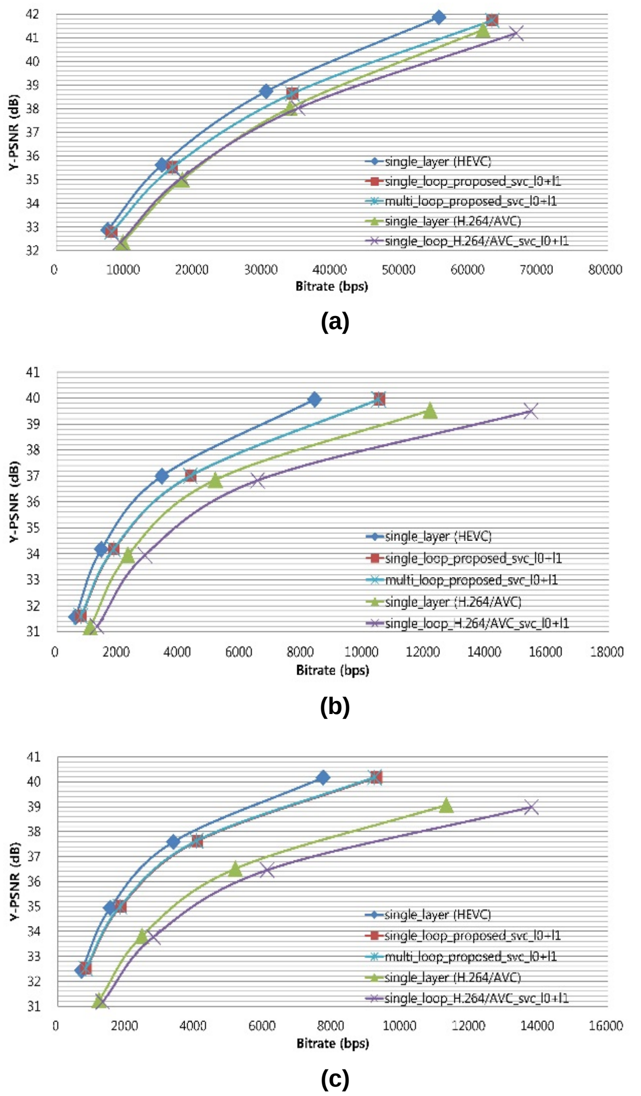
(c)

except for the first slice. Therefore, the proposed Q-ILTP is applied only for beginning of a sequence since the first frame coded by intra mode can be fully reconstructed. Inter slices are able to exploit the proposed Q-ILSP instead of the fully reconstructed texture for the proposed. The accuracy of prediction by Q-ILSP and Q-ILTP significantly impacts on the RD performance for the single-loop design framework. We found out that the RD performance of single-loop design is about 2% worse in terms of BD-E-Bitrate when compared to that of the multi-loop framework for the LD case. Therefore, the proposed framework with single-loop design is able to support scalability with 26.5% overhead bits. For the RA case, higher bit savings are found in BD-E-Bitrate than in the LD case in single-loop decoding framework since the periodic intra slices are fully reconstructed and it can be exploited by the Q-ILTP at the enhancement layer. However, it is worse about 2.3% than that of multi-loop decoding framework in terms of BD-E-Bitrate. The proposed framework with single-loop decoding design saves bits about 9.1% at the enhancement layer compared to the HEVC single-layer coding. In addition, the framework supports spatial scalability with about 21.9% overhead bits.

Fig. 9 shows RD curves for 'ParkScene' sequence which has Full-HD resolution in the Class B. Three different plots are test results on each coding condition cases, AI, LD, and RA. Each plot has five RD curves. First, coding results by single-layer HEVC is remarked by 'single_layer (HEVC)'. The proposed framework on single-loop and multi-loop decoding design are remarked by 'single_loop_proposed_svc_l0+l1' and 'multi_loop_proposed_svc_ l0+l1', respectively. For RD curves of the proposed scalable video coding framework, total bitrate by all layers are exploited with PSNR of the highest layer.

RD curve of H.264/SVC, it also employs total bitrate by all layers with PSNR of the highest layer as same as the proposed framework.

Fig. 9(a) shows the RD curves for the AI case. For 'All-intra' case, the proposed Q-ILTP works in an identical

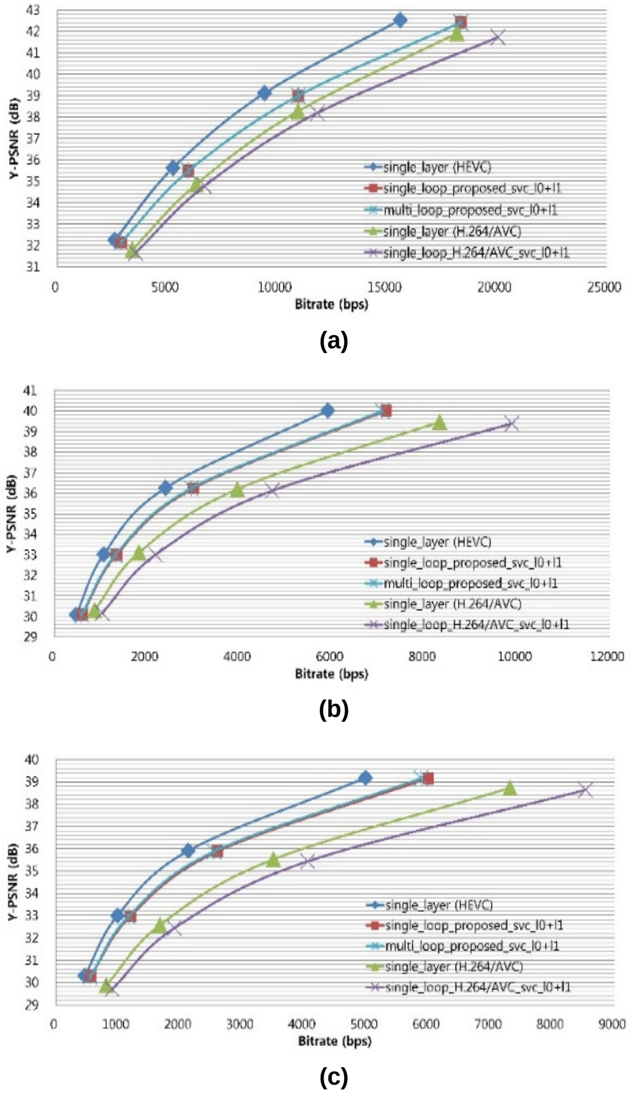


(a)

(b)

(c)

Fig. 9. RD curves for 'ParkScene' in Class B for (a) AI, (b) LD, and (c) RA.



(a)

(b)

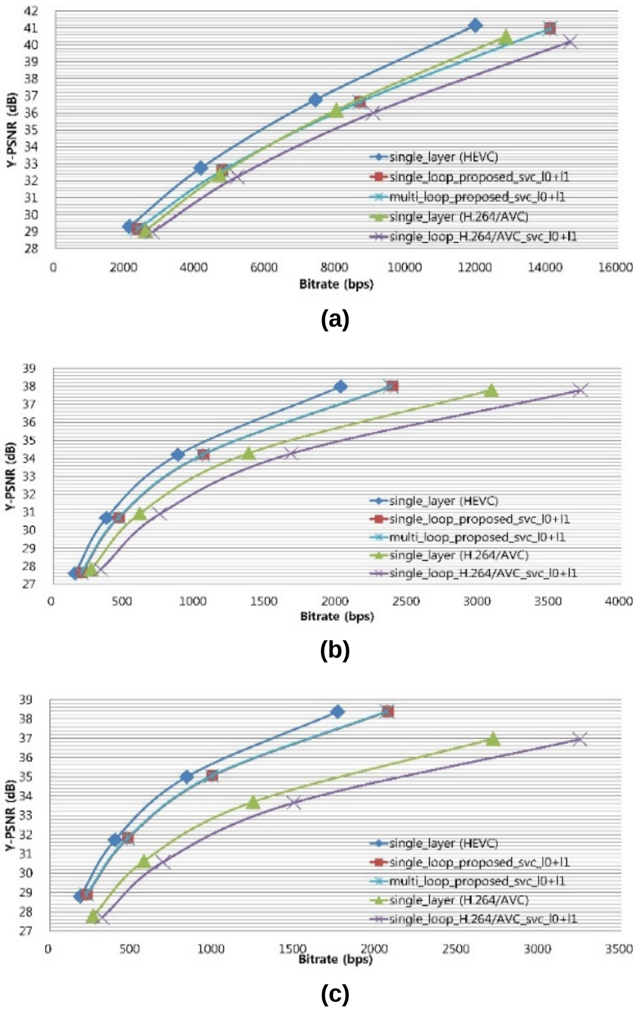
(c)

Fig. 10. RD curves for 'RaceHorses' in Class C with (a) AI, (b) LD, and (c) RA.

manner on both the proposed framework. Therefore, RD curves are on the identical position as shown in the plot. Fig. 9(b) shows the RD curves for 'ParkScene' on LD case. Gaps between 'multi_loop_proposed_svc_l0+l1' and 'single_loop_proposed_svc_l0+l1' are narrow. However, it does not mean that there are no difference between two loops design in coding performance because we found that not only the gaps, but also coding gains in BD-E-Bitrate and BD-T-Bitrate are highly different depending on characteristics of sequences as shown in Table 2 and Table 3. The RD curves shown in Fig. 9 are depicted with respect to the total bitrate of all layers are exploited with PSNR of the highest layer. Therefore, it shows only the overall coding performance compared to the single-layer coding using HEVC encoder and H.264/SVC. For a sequence which has slow movement with simple texture such as 'Kimono', there are high bits saving in BD-E-Bitrate for both single and multi-loop decoding design on LD case and the gaps are also narrow. In addition, there are poor coding gains in BD-E-Bitrate, but the gaps are narrow for a sequence which has too complex texture such as 'BQTerrace'. Therefore, accuracy of high-frequency in predicted texture is a key point to make high bits saving at the enhancement layer and lower overhead bits in spatial scalability. RD curves on RA case are shown in Fig. 9(c).

Fig. 10 shows RD curves for 'RaceHorses' which has WVGA resolution in Class C. The sequence is also tested on AI, LD, and RA case. Entire tendency of curves are similar to the 'ParkScene' sequence as well. However, gaps between curves change because the characteristics of sequences are different. For LD case, gaps between 'single_loop_proposed_svc_l0+l1' and 'multi_loop_proposed_svc_l0+l1' are about 2%. It means that the high-frequency in predicted textures by the proposed Q-ILP was more efficient in multi-loop decoding design for the sequence.

Fig. 11 shows RD curves for 'BlowingBubbles' which has WQVGA resolution. The overhead bits between proposed SVC and single layer are just about 17%, but the bits between H.264/SVC and single layer H.264/AVC are



(a)

(b)

(c)

Fig. 11. RD curves for ‘BlowizgBubbles’ in Class D with (a) AI, (b) LD, and (c) RA.

21%. According to experiments, the proposed scalable video coding framework based on HEVC shows highly corresponding RD performance to the H.264/SVC because the tendency of gaps between single layer and SVC framework are similar to both the proposed SVC and H.264/SVC.

5. Conclusion

In this paper, we presented a foundational framework of scalable video coding for the future scalable coder. To support not only spatial, quality, temporal scalabilities, but also view scalability, the proposed scalable video coding framework alternatively works with single-loop decoding design and multi-loop decoding design. Furthermore, we proposed quad-tree inter-layer prediction tools which work depending on the decoding loop design of the proposed framework to improve RD performance at enhancement layers. The proposed tools exploit reconstructed texture, motion information, and intra prediction mode syntaxes from a reference layer. The proposed framework guarantees the rate distortion performance for a base layer because it does not have any compulsion such a constraint intra prediction. Moreover, additional de-blocking filter and padding algorithm for only the inter-layer prediction are not required.

According to experiments, the framework supports the spatial scalable functionality with just about 18.6%, 18.5% and 25.2% overhead bits against to HEVC single layer coding. Furthermore, the proposed inter-layer prediction tools enable to achieve coding gains of about 14.0%, 4.4%, and 10.5% in BD-Bitrate at the enhancement layer, compared to a simulcast platform for all-intra, low-delay, and random access cases, respectively. In addition, the proposed framework objectively achieves about 41% bits saving compared to H.264/SVC on top of the similar quality.

According to experiments, the framework supports the spatial scalable functionality with about 18.6%, 18.5% and 25.2% overhead bits against to HEVC single layer coding. Furthermore, the proposed inter-layer prediction tools enable to achieve coding gains of about 14.0%, 4.4%, and 10.5% in BD-Bitrate at the enhancement layer, compared to a simulcast platform for all-intra, low-delay, and random access cases, respectively. In addition, the proposed framework objectively achieves about 41% bits saving, compared to H.264/SVC on top of the similar quality. As described in the previous section, we implemented the proposed method on HEVC reference software rather than SHVC reference software because the proposed method can be performed on both single-loop and multi-loop coding and SHVC reference software which supports only multi-loop coding is not suitable. However, for the future scalable coding, the proposed method can be used and achieve the additional coding gain for enhancement layer based on the texture and syntax redundancies among layers.

Acknowledgement

This research was partly supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning(NRF-2014R1A2A1A11052210) and the Commercializations Promotion Agency for R&D Outcomes(COMPA) funded by the Ministry of Science, ICT and Future Planning(MISP).

References

[1] A. Luthra, “Scalable enhancement requirements for HEVC,” *Document of Joint Collaborative Team on Video Coding*, JCTVC-E502, Geneva, CH, March, 2011. Article (CrossRef Link)

[2] K. Rantelobo, W. Wirawan, G. Hendrantoro, A. Affadi and H. Zhao, “Adaptive Combined Scalable Video Coding over MIMO-OFDM Systems using Partial Channel State Information,” *KSII Transactions on Internet and Information Systems*, vol. 7, no. 12, December, 2013. Article (CrossRef Link)

[3] G. Bang, N. Hur, and S. Lee, “Post-processing of 3D

video extension of H.264/AVC for a quality enhancement of synthesized view sequences," *ETRI journal*, vol. 36, no. 2, pp. 242-252, April, 2014. Article (CrossRef Link)

[4] H. Schwarz, T. Hinz, D. Marpe, and T. Wiegand, "Constrained inter-layer prediction for single-loop decoding in spatial scalability," *IEEE International Conference on Image Processing*, pp. 870-873, Genoa, IT, September, 2005. Article (CrossRef Link)

[5] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 17, no. 9, pp. 1103-1120, September, 2007. Article (CrossRef Link)

[6] K. De Wolf, D. De Schrijver, S. De Zutter, and R. Van de Walle, "Analysis and coding performance of inter-layer prediction," *International Symposium on Signal Processing and Its Applications*, pp. 1-4, Sharjah, AE, Feb. 2007. Article (CrossRef Link)

[7] H. Ke, "myEvalSVC: an Integrated Simulation Framework for Evaluation of H.264/SVC Transmission," *KSII Transactions on Internet and Information Systems*, vol. 6, no. 1, January, 2012. Article (CrossRef Link)

[8] Y. Cho, H. Radha, J. Seo, J. Kang, and J. Hong, "Multihop rate adaptive wireless scalable video using syndrome-based partial decoding," *ETRI journal*, vol. 32, no. 2, pp. 273-280, Apr. 2010. Article (CrossRef Link)

[9] M. Wien, H. Schwarz, and T. Oelbaum, "Performance analysis of SVC," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 17, no. 9, pp. 1194-1203, Sep. 2007. Article (CrossRef Link)

[10] X. Li, P. Amon, A. Hutter, and A. Kaup, "Performance analysis of inter-layer prediction in scalable video coding extension of H.264/AVC," *IEEE Trans. on Broadcasting*, vol. 57, no.1, pp. 66-74, Mar. 2011. Article (CrossRef Link)

[11] T. Wiegand, W.-J. Han, B. Bross, J.-R. Ohm, and G.J. Sullivan, "WD2: Working draft 2 of high-efficiency video coding," *JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCT-VC D503 (m18644), Daegu, KR, Jan. 2011. Article (CrossRef Link)

[12] W. Zhang, A. Men, and P. Chen, "Adaptive inter-layer intra prediction in scalable video coding," *IEEE International Symposium on Circuits Syst.*, pp 876-879, Taipei, TW, May, 2009. Article (CrossRef Link)

[13] H.M. Choi, J.-H. Nam, D.-G Sim, and I. V, Bajić, "Scalable video coding based on high efficiency video coding (HEVC)," *IEEE Pacific Rim Conference on Communications*, *Computers and Signal Processing*, pp.346-351, Victoria, CA, Aug. 2011. Article (CrossRef Link)

[14] H.M. Choi, J.-H. Nam, and D.-G. Sim, "Scalable structures and inter-layer predictions for HEVC scalable extension," *JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCT-VC F096 (m20509), Torino, IT, Jul. 2011. Article (CrossRef Link)

[15] C.-W. Seo and J.-K. Han, "Pixel based illumination compensation for inter prediction in HEVC," *Electronics Letters*, vol. 47, no. 23, pp. 1278-1280, Nov. 2011. Article (CrossRef Link)

[16] H. Choi, K. Lee, S.-J. Bae, J.W. Kang, and J.-J. Yoo "Performance evaluation of the emerging scalable video coding," *International conference on Consumer Electronics*, pp. 1-2, Taipei, TW, Jan. 2008. Article (CrossRef Link)

[17] H. Lee, J. Kang, J. Lee, J. Choi, J. Kim, and D. Sim, "Scalable Extension of HEVC for Flexible High-Quality Digital Video Content Services," ETRI journal, vol. 35, no. 6, pp. 990-1000, Dec. 2013. Article (CrossRef Link)

[18] F. Bossen, "Common test conditions and software reference configurations," *JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, JCT-VC D600 (m19497), Daegu, KR, Jan. 2011. Article (CrossRef Link)

[19] G. Bjøntegaard, "Calculation of average PSNR differences between RD-Curves," *ITU-T SG16/Q.6*, VCEG-M33, Austin, TX, Apr. 2001. Article (CrossRef Link)

Woong Lim was born in Busan Metropolitan city, Republic of Korea, in 1981. He received the B.S. and M.S. degree in Computer Engineering from Kwangwoon University, Seoul, Korea, in 2008 and 2010, respectively. Now, he is working toward Ph.D. degree at the same University. His current research interests are high-efficiency video compression, Scalable video coding, multi-view video coding and computer vision.

Hyomin Choi was born in Seoul, Republic of Korea, in 1987. He entered the Kwangwoon University in 2006. which received the B.S. and M.S. degrees in Dept. of Computer Engineering in 2010 and 2012, respectively. He joined Image Processing Systems LAB (IPSL) in 2009. He had studied video coding standards,H.264/AVC, H.264/AVC scalable extension, HEVC, HEVC scalable and multi-view extension, and worked for its standardization. Now, he is one of alumnis of this LAB. He is working at LG Electronics since 2012

Junghak Nam was born in SangJu City, KyungBuk Province, Republic of Korea, in 1979. He entered the Kwangwoon University in 1998, which received the B.S. and M.S. degrees in Computer Engineering, in 2006 and 2008, respectively. And he received Ph.D is currently a Ph.D. in

2013. He is currently working for LG Electronics since 2013. His interesting fields are H.264/AVC, HEVC, and parallel processing

Donggyu Sim was born in Chungchung Province, Korea, in 1970. He received the B.S. and M.S. degrees in Electronic Engineering from Sogang University, Seoul, Korea, in 1993 and 1995, respectively. He also received Ph.D. degree at the same University in 1999. He was with the Hyundai Electronics Co., Ltd. from 1999 to 2000, where was involved in MPEG-7 standardization. He was a senior research engineer at Varo Vision Co., Ltd., working on MPEG-4 wireless applications from 2000 to 2002. He worked for the Image Computing Systems Lab. (ICSL) at the University of Washington as a senior research engineer from 2002 to 2005. He researched on the ultrasound image analysis and parametric video coding. He joined the Department of Computer Engineering at Kwangwoon University, Seoul, Korea, in 2005 as an Associate Professor. He was elevated to an IEEE Senior Member in 2004. His current research interests are image processing, computer vision, and video communication.