Fast decoder for H.264 scalable video coding with selective up-sampling for spatial scalable video coding

IlHong Shin, Haechul Choi, Jeong Ju Yoo, and Jin Woo Hong
Electronics and Telecommunications Research Institute, Convergence Media Research Team, Broadcasting Communication Research Group, 161 Gajeong-dong, Yuseong-gu, Daejeon, South Korea

Abstract. A simple and effective method is presented for fast decoding of H.264 scalable video coding (SVC). The up-sampling operation in H.264 SVC makes the decoder very complex, because convolution and complex memory transactions are inevitable. The proposed method exploits coded modes of neighboring macroblocks (MBs) for determining up-sampling operation on MB by MB. The experimental validation shows considerable improvement in decoding time, and the proposed method reduces the complexity by about 25% on average.
[DOI: 10.1117/1.2957042]

Subject terms: video decoder; scalable video coding; upsampling.

Paper 080173LR received Mar. 9, 2008; revised manuscript received May 18, 2008; accepted for publication May 20, 2008; published online Jul. 16, 2008.

1 Introduction

Although transcoding methods[1–4] support spatial, temporal, and quality adaptation, they require additional tools in transmission. Consequently, a new standard was proposed to provide scalable video models for scalable extension of H.264/AVC [H.264 scalable video coding (SVC)].[5–7] It inherited most building blocks of H.264 with some improved features for scalability such as hierarchical B pictures.[6] For spatial scalability, 12 taps and four tap filters were exploited in the current joint scalable video model (JSVM) for down/up-sampling, respectively.[7–9]

The decoding complexity of H.264 SVC is generally higher than single layer coding compatible with H.264, because new standards require various tools adopted in H.264 SVC. Also, spatial scalable video coding employs reconstruction of the spatial base layer. Hence, H.264 SVC adopts a single-loop decoding mode, where it only reconstructs intra-macroblocks (MB) on base layers for enhancement of decoder performance, excluding a complex inter-reconstruction process.[6] However, H.264 SVC still needs up-sampling for spatial SVC to provide prediction signals in the spatial enhancement-layer.[7,8] Table 1 shows the complexity portion of each decoding block with spatial SVC with two layers. As shown in Table 1, the up-sampling operation is the most time-consuming part within the main coding block. The convolutions with a four-tap filter and complex memory operation are the main reasons for inefficient decoders of H.264 SVC.

Table 1 The complexity portions of each decoding block between JSVM and the proposed method.

Decoding module	JSVM (%)	Proposed (%)
Texture up-sample	30	4
Residual up-sample	5	6
Motion up-sample	0.6	1
Entropy	5.7	7.4
Loop filter	3.5	5.5
Motion compensation	12	15
Transform and quantization	8	14
Others	35	47

We proposed a fast decoding method with selective up-sampling, which determines a valid base-layer block for up-sampling on MB by MB. Since a single-loop decoding mode allows the prediction of only MB having an intra-mode in the base layer, the validity of MB for up-sampling is checked by investigating the coded mode of the neighboring MB.

We proposed a selective up-sampling method for a two-fold case in Sec. 2, the experimental result is described in Sec. 3, and Sec. 4 concludes the work.

2 Proposed Method

Mainly, H.264 SVC employs texture, residual, and motion reuse for coding efficiency. All the methods use up-sampling for generating prediction signals in the spatial enhancement layers. However, major complexity arises from texture up-sampling due to four-tap convolution filters and memory operation. The I_BL mode[6] defined in H.264 SVC uses texture up-sampled signals to predict the current MB. However, H.264 SVC adopts the single-loop decoding mode for efficient decoding, where only intraMB can be reconstructed and up-sampled. It should be noted that intramode defined in the context of H.264 SVC includes I_4×4, I_16×16, pulse code modulation (PCM) (conventional mode in H.264), and I_BL (new in H.264 SVC). Hence, the single-loop decoding mode works for any number of spatial layers. However, the current JSVM up-sample reconstructed signal of the base layer has no special considerations for the single-loop decoding mode. The required up-sampling of texture information can be checked by the coded mode of the spatial base layer. Assume that the base layer has no intraMB; in this case, a spatial enhancement layer does not use texture information of the base layer due to a single-loop decoding mode. We can make up-sampling operations efficient in the H.264 SVC decoder, where the proposed operation selectively up-samples on the basis of macroblock modes. However, determination of up-sampling of current MB in the spatial enhancement layer is complex due to the four-tap convolu-

0091-3286/2008/$25.00

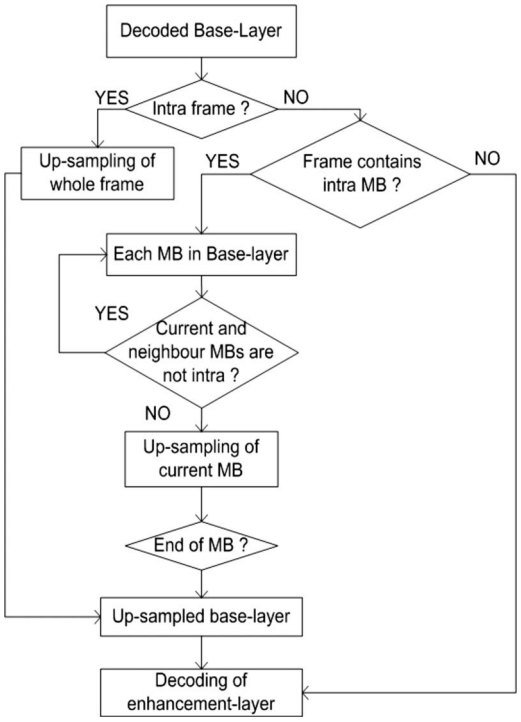

Fig. 1 The proposed block diagram for selective up-sampling.

tion filter. The convolution process needs pixels across the current MB; therefore, four neighboring MBs should be considered for selective up-sampling. Four neighboring MB mean upper, lower, left, and right MBs in regards to the current MB. Also, isolated intraMB should be padded for correct up-sampling,[6] because single-loop decoding modes in the base layer have residual signals only, if the current MB is not intramode.

Figure 1 shows the proposed selective up-sampling method. The frame type, such as intra- or interframe, should be checked before the up-sampling operation in the decoder. When the frame type is intra, all frames should be up-sampled to provide prediction signals for the spatial enhancement layer, because the base layer contains only intraMBs. If the current frame type is not intra, the number of intraMBs is counted in the base layer. When the number of intraMBs is zero, the enhancement layers do not have the intra BL (I_BL) mode, because the single-loop decoding mode does not allow the I_BL mode for spatial coding efficiency. Then, the up-sampling operation for texture information can be skipped for computational efficiency. When the base layer has intraMBs, the coded mode of four neighboring MBs should be checked in the base layer, including current MBs. Up-sampling of current MBs in the base layer should be done in the case of at least one intramode within five MBs for providing correct padding signals. Following the proposed method, computational reduction is achieved with coded modes of the current MB and neighbor MBs. As shown in Fig. 1, the up-sampling process can be done selectively in accordance to the coded mode of the base layer. Therefore, we can reduce H.264 SVC decoder complexity considerably.

Table 2 The decoding speed and time comparison between JSVM and the proposed method.

Sequence	Spatial configuration	JSVM (Hz/sec)	Proposed (Hz/sec)	Improvement (%)
Stefan	QCIF-CIF	39.72/7.5	50.39/5.9	26
Football	QCIF-CIF	38.98/7.7	49.28/6.1	26
Mobile and Calender	QCIF-CIF	35.45/8.5	43.37/6.9	22
City	CIF-4CIF	10.43/28.8	13.93/21.5	33
City	QCIF-CIF-4CIF	9.18/32.7	12.65/23.7	37

3 Experimental Results

Four video sequences [Stefan, Football, Mobile, and Calendar (352×288, Common Intermediate Format (CIF)), and City (704×576, 4CIF), all 30 Hz and 300 frames] were used for experimental validation. Although H.264 SVC supports arbitrary up-sampling ratios called extended spatial scalability (ESS),[6] the proposed method can be used with only the two-fold case, which means that the ratio of width and height between base and enhancement videos is two, respectively. All sequences were down-sampled correctly by a JSVM down-sampling filter of 12 taps. Therefore, 176×144 (QCIF) and 352×288 (City) sequences were encoded as base-layer bitstreams, where CIF and 4CIF original resolution were used for the enhancement layer. The simulation was done with an Intel Pentium 4 3.0 GHz; an Intel VTune 7.0 profiler was also used for profiling.

All test sequences were codes such as IPP (low delay configuration in the JSVM[4]). The quantization parameters (QP)s of the base and enhancement layers were set to 30. The base layer for spatial scalability was coded by H.264 AVC compatible, which means the base layer is compatible with the H.264 coded bistream,[10] and context adaptive binary arithmetic coding (CABAC) was used for entropy coding.[6]

Table 1 shows relative complexities of modules in the proportion of their computation times in the JSVM and proposed method, where the major complexities came from the up-sampling operation for spatial scalability. The proposed method reduces decoding complexity of texture up-sampling, as shown in Table 1, where other operations such as loop filters increased due to the reduced ratio complexity of texture up-sampling. Table 2 shows the decoding speed comparison between JSVM and the proposed method. The spatial configuration means the number of spatial layers in H.264 SVC. Hence, QCIF-CIF-4CIF supports three spatial layers. Improvement of the decoding speed with QCIF-CIF configuration is about 25%, while the CIF-4CIF case shows 33% improvement, because the number of memory accesses is reduced by the proposed method. The last row in Table 2 shows experimental results with three spatial layers. It shows a great reduction in decoding complexity.

4 Conclusion

We propose a simple and efficient method for reducing the complexity of the H.264 SVC decoder with a spatial scalable application. The proposed method exploits the single-loop decoding mode adopted in H.264 SVC, where only intrablock is reconstructed and used as a prediction signal for texture in the spatial enhancement layer. The proposed method checks the coded mode of four neighboring MBs for determining selective up-sampling. Following the proposed method, experimental validation shows that complexity reduction is about 25% (QCIF-CIF configuration), and about 33% (CIF-4CIF configuration), respectively. The proposed method will be essential for fast decoder implementation of H.264 SVC, especially in spatial scalable applications. However, the proposed method is only for two-fold up-sampling, such as from QCIF to CIF. Currently, extension to arbitrary-ratio scalable video coding adopted in H.264 SVC, called as ESS (extended spatial scalability)[6] is underway by the authors.

Acknowledgment

This work was supported by the Ministry of Knowledge and Economy of the Korean government.

References

1. G. Keesman et al., "Transcoding of MPEG bitstreams," *Signal Process. Image Commun.* **8**, 481–500 (1996).
2. C. L. Salazar and T. D. Tran, "On resizing images in the DCT domain," *Proc. Intl. Conf. Image Process.* (Oct. 2004).
3. T. D. Nguyen, G. Lee, J. Y. Chang, and H. J. Cho, "Efficient MPEG-4 to H.264/AVC transcoding with spatial downscaling," *ETRI J.* **29**(6), 826–828 (2007).
4. T. Frajka and K. Jegger, "Downsampling dependent upsampling of images," *Signal Process. Image Commun.* **19**, 257–265 (Mar. 2004).
5. Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, "Joint scalable video model JSVM-9," JVT-V202, Marrakech, Morocco, Jan. 2007.
6. H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable extension of the H.264/MPEG-4 AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.* **17**(9), 1103–1120 (Sep. 2007).
7. F. Wu, S. Li, and Y. Q. Zhang, "A framework for efficient progressive fine granular scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.* **11**(3), 332–344 (Mar. 2001).
8. C. A. Segall and G. J. Sullivan, "Spatial scalability within the H.264AVC scalable video coding extension," *IEEE Trans. Circuits Syst. Video Technol.* **17**, 1121–1135 (Sep. 2007).
9. I. H. Shin and H. W. Park, "Efficient down-up sampling using DCT kernel for MPEG-21 SVC," *Proc. Intl. Conf. Image Process.*, pp. 640–643 (Sep. 2005).
10. G. Sullivan and T. Wiegand, "Video compression—from concepts to H.264/AVC standard," *Proc. IEEE* **93**(1), 18–31 (Jan. 2005).