Extended Temporal Scalability for Low-Performance Devices

Jonghun Lee, and Heonshik Shin
School of Computer Science and Engineering, Seoul National University, Korea
terminal@cslab.snu.ac.kr, shinhs@snu.ac.kr

Abstract — *A scalable video stream can accommodate various forms of network diversity. Scalability can be provided by H.264 SVC (scalable video coding) extension, which is an amendment to the H.264/AVC (advanced video coding) standard. Among various types of scalability, temporal scalability is easily implemented using hierarchical B structure, but the increase in decoding complexity inhibits a flexible streaming service for low-performance devices like mobile phone. In this paper, we present two ways of increasing temporal scaling by realizing non-dyadic temporal scalability with low complexity. Both methods are shown to be easily merged into the standard, H.264 SVC extension. We have observed that, at the expense of a small amount of additional data, the decoding complexity can be reduced by up to 35%, offering a more flexible streaming service for mobile devices.*

Index Terms — Video codec, H.264, AVC, low-complexity, mobile device, scalable video.

I. Introduction

There is a wide diversity of network connections in a modern video streaming service as in the case of telephone lines vs. gigabit LANs. There are also significant differences in device resources, such as display resolution and computational power. In addition, the throughput of a user's connection can fluctuate dynamically over time. Thus, we must take these dissimilarities into account when we provide high-quality video streaming efficiently over the wireless network.

A simulcast is a simple approach to stable video streaming that copes with both static and dynamic diversity. In this scheme, many encoders are used to encode a stream in parallel into many versions, each of which is suitable for a different bandwidth or resolution. To overcome dynamic diversity, the streaming server can adjust the streaming rate within the user's current reception throughput by switching the stream. However the simulcast wastes network resources and users cannot avoid a long 'zap-time' when they wish to change the stream that they are receiving.

Scalable video can be used to avoid these problems and enable a more flexible streaming framework. Thus such a provision has been included in video standards since MPEG-2 [1] and the recent H.264 SVC extension is an amendment to the H.264/AVC standard.

There are three types of scalability in H.264 SVC extension. Unlike other two scalable coding, temporal scalability can be used with the original H.264/AVC decoder without any modification because this is implemented using the hierarchical B structure, which is already compatible to the standard [2]. Because of this simplicity, there has been little investigation on temporal scalability.

It should be noted, however, that the hierarchical B structure has a grave disadvantage: Its time resolution can only be dyadic. This poses a serious problem, especially for low-performance mobile and embedded devices, because the complexity of video decoding is proportional to the number of frames. This means that adding a layer doubles the decoding complexity. A dyadic solution makes the complexity gap between two frame-rate levels very large, and thus restricts the freedom with which a streaming service can move from one layer to another. Since the frame rate is an important metric of video quality, just like the PSNR of each picture in a video, overcoming the dyadic constraint would be the major contribution to improving video quality.

In this paper, we present two methods for extending temporal scalability to provide non-dyadic temporal resolution. Our approaches are designed to provide a more flexible video streaming scalability, especially for low-performance devices, and they are based on the techniques adopted in the standard [2]; so, they do not cause any changes to the standard-compliant decoders.

The rest of this paper is organized as follows: the background is summarized in Section II, and in Section III we describe the problems of temporal scalable coding using the hierarchical B structure in more depth. In Sections IV and V, we present coding structures followed by experimental results. Concluding remarks are made in Section VI.

II. The H.264 Scalable Video Coding Extension

There are three types of scalable video coding in the H.264 extension; spatial, temporal, and quality (or SNR) scalability [3]. Temporal scalability is implemented using the hierarchical B structure. Unlike older video standards, any frame can be marked as a reference frame in H.264. Allowing any frame to be the reference for a frame in an enhancement layer makes it relatively easy to implement temporal scalability with high coding efficiency. The resulting coding structure is shown in Figure 1. In the figure, arrows indicate 'prediction'. The key frames (in the base layer) are coded using only other key frames as reference, and the frames between two key frames are predicted hierarchically.

This work was supported by Varo Vision Co., Ltd.

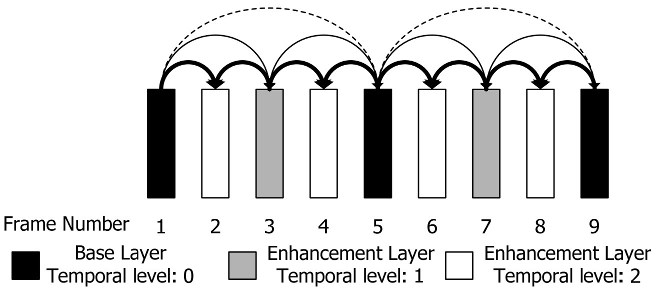

Figure 1. The coding structure of the hierarchical B structure.

Spatial scalability is used when a stream services the users who demand different resolutions. The stream with the lowest resolution consists of the base layer only, but streams with higher resolutions include one or more of the enhancement layers.

In terms of SNR (or quality), there are two types of scalabilities; FGS (fine-grain scalability) and CGS (coarse-grain scalability). CGS uses the same concept of spatial scalability, but the up-sampling operations of the inter-layer prediction mechanisms are omitted [3].

The FGS technique was standardized in MPEG-4 [4], but some additional techniques are included in the H.264 SVC extension. For example, intra-layer prediction is allowed to increase coding efficiency. Also, Van der Schaar and Radha have also proposed [5] using FGS to improve temporal resolution. Their scheme achieves good video quality and makes streaming services more flexible, but it is still a dyadic solution like the hierarchical B structure.

III. Problems with The Hierarchical B Structure

Before introducing our proposal, we will look at the problems of legacy temporal scalability caused by the hierarchical B structure.

A. Decoding Complexity

The decoding complexity of a video sequence is mainly determined by the number of frames. It is easy to see that this complexity is proportional to the number of frames. And the decoding complexity of a single frame is determined by its prediction type, if other factors such as the QP (quantization parameter) are similar. Ray et.al. [6] showed how the complexity of a frame varies with its prediction type in the H.264 JVT decoder. They demonstrated that a B frame has the highest complexity, which can be 20% more than that of other frame types in some sequences.

In the hierarchical B structure, switching to the upper layer forces the decoder to deal with double the number of frames, all of which are of bi-directional prediction type. The observations above suggest that the complexity will be at least doubled in this situation. We conducted a preliminary experiment using the 'Harbor' stream. First, we extracted a sequence at a frame-rate of 20Hz from the original 60Hz video. After this, we downsampled the sequence to CIF (352x288) size for more general scenario of low-performance devices. Table 1 summarizes the details of this stream.

Table 1. Test stream.

Sequence	Resolution	Frame rate (Hz)
Harbor	352 x 288	20

We encoded this 20Hz sequence using the hierarchical B structure with the JSVM H.264 SVC encoder [7]. The encoded bit-stream had three different temporal resolutions with frame rates of 5, 10, and 20Hz. After encoding, we decoded it using the JSVM H.264 decoder and evaluated the complexity by measuring the decoding time. The JSVM decoder was slightly modified to obtain the decoding time, and we ran it on a Pentium 3 platform. Figure 2 shows the decoding times at each frame-rate.

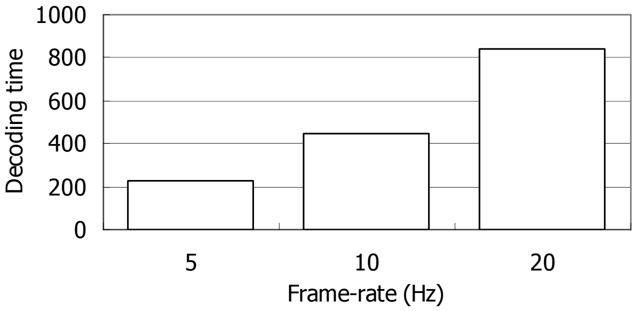

Figure 2. The decoding complexity of a hierarchical B structured stream.

The results suggest that the complexity almost doubles for each increment in frame-rate. The increases will, therefore, inhibit layer switching. This problem motivates the development of a scalable video streaming policy which improves the picture quality (i.e. the PSNR) of each frame, rather than the temporal resolution.

B. The Relationship between Stream Quality and Frame-Rate

It is difficult to evaluate the quality of a video stream from its frame rate. There have been intensive research efforts on this problem; still, objective quality metrics like PSNR have not been agreed upon. It is, however, always the case that a higher frame-rate will improve the quality of a video if the quality of each frame is fixed. Furthermore, most researchers agree [8] that a video encoder should allocate additional bit-rate to increase the frame-rate rather than to improve the PSNR of each frame, especially when a video contains more motion information. This observation supports our contention that more flexible temporal scalability is required.

IV. Method 1: Using SNR Scalability

In this section and the next, we will introduce two methods that can provide non-dyadic temporal resolution.

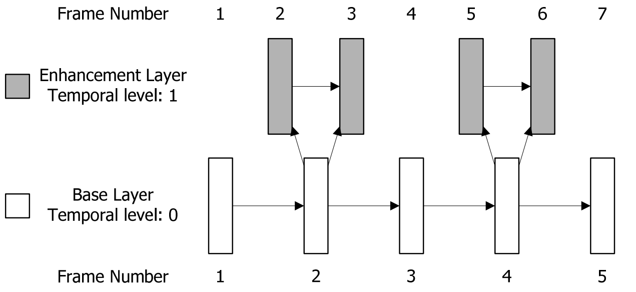

Figure 3. The coding structure of the Method 1.

A. Coding Structure

The first method does not use bi-directional prediction, making it suitable for low-delay applications and low-performance decoders. Instead of using a B frame, this method uses SNR scalability as shown in Figure 3. The base layer contains a video stream at a low frame rate and the enhancement layer contains differences between the videos of the low frame rate and a higher frame-rate. If the display durations of those two frames are the same, there would be no difference and thus no data for enhanced layer is needed. Otherwise the enhancement layer needs to incorporate some residual data to correct the errors caused by the difference in display time.

B. Picture Quality

We extracted 10 and 15Hz video sequences from the 60Hz 'Harbor' sequence and encoded them into a stream with two temporal resolutions, 10 and 15Hz, using Method 1, which we had implemented on a JSVM SVC encoder. The target bit-rates were 192, 256, 320 and 384Kbps. For comparison we encoded four streams with the same target bit-rate using the hierarchical B structure. The details of the encoded streams are summarized in Table 2.

Table 2. Encoded streams.

Sequence	Spatial Resolution	Bit-rate (Kbps)	Temporal scalability	Temporal Resolution (Hz)
Harbor	352 x 288	183	Type 1 method	10, 15
		271		
		318		
		391		
		191	Hierarchical B structure	5, 10, 20
		262		
		338		
		384		

All streams have just one IDR (instantaneous decoding refresh) frame (which is, of course, the first frame) and are encoded during the first second. The coding efficiency of SVC is not high when compared to the use of bi-directional frames [3, 9]. Therefore, we use CGS, which achieves better picture quality than FGS [3, 9] as the SNR scalable coding for Method 1. To measure the coding overhead of our method, we calculate the PSNR of the stream for each frame rate. Figure 4 shows the averaged PSNR of each stream.

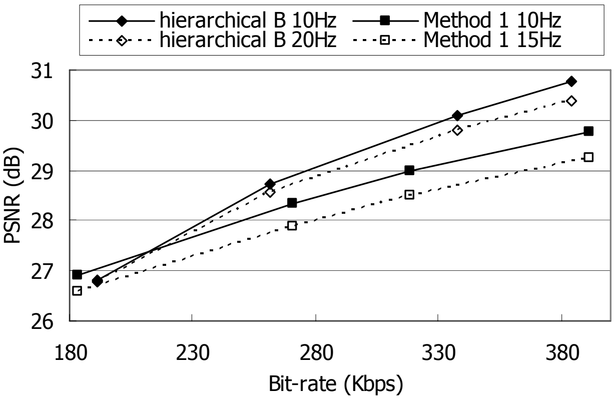

Figure 4. The averaged PSNRs of the streams.

We can see that the quality of the 10Hz stream is a little lower with Method 1 than that of the stream of the same frequency with the hierarchical B structure. This is because of the low efficiency of the scalable coding [3, 9] and also because the 10Hz sequence produced by Method 1 does not use bi-directional prediction for low-delay applications. However, this level of degradation is not considered severe and it can be offset against the improved temporal quality of the 15Hz stream.

C. Decoding Complexity

We evaluated the decoding complexity of our scheme in the same way as we did in the preliminary experiment in Section III. Then we compared this result with the decoding times for the 20Hz stream encoded using the hierarchical B structure. Since the decoding complexity is dominated by, and almost proportional to, the number of frames, we only show representative results for a 318Kbps stream encoded using Method 1 and the hierarchical B structure, which can be found in Figure 5.

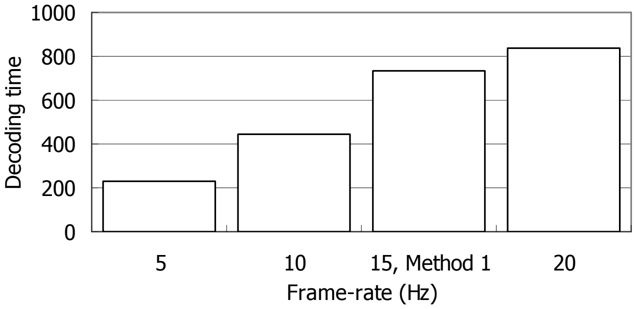

Figure 5. The decoding complexity of a stream encoded using Method 1, compared with a hierarchical B structured stream.

The reduction in complexity does not match the reduction in frame rate because increased compression is achieved by SVC at the expense of greatly increased computational complexity [10]. The decoding complexity of the 15Hz stream is only 13% less than that of the 20Hz hierarchical B structured stream, but this is enough to enable more generous temporal scalability. There are additional savings from the drop in frame rate, such as a lower transmission cost for the decoded data. Recent research has addressed the complexity of SVC [10, 11], and any resulting reduction in decoding complexity would promote more flexible streaming.

V. Method 2: Sharing Key Pictures

Method 1 has a simple structure and can be easily merged with a current streaming service system. But its compression performance is not as high as expected. Method 2 can overcome the efficiency problem, assuming that some cooperation is available from routers, a gateway, an access point or a base station. The required cooperation will be described later.

A. Coding Structure

Method 2 uses bi-directional prediction like the hierarchical B structure, but with the additional feature that there are two parallel upper enhancement layers. Two streams at different frame rates to share a single anchor stream, which is the base (or a lower layer), as shown in Figure 6.

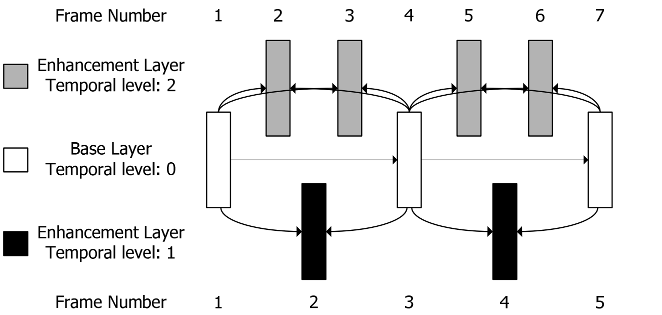

Figure 6. The structure of Method 2.

In this parallel scheme, an encoder can not exploit the information of the next highest layer when it encodes the highest layer. But, since Method 2 inherits the coding efficiency of the B frame, and the length of a B frame is relatively small, it can largely avoid degradation of picture quality caused by the extra data.

In a scalable video streaming service, routers, gateways, access points or base stations are important constituents. They check the priority in the NAL (network abstraction layer) packet header and drop the lowest-priority NAL packets if resources are limited. As the situation deteriorates, packets with the next lowest priority are dropped, and so on. Scalability is realized by this mechanism where the priority is assigned by its layer level in the hierarchical way. However, since two layers of Method 2 are in parallel, above packet delivery scheme can not realize the full scalability. The flowchart of a delivery scheme suitable for supporting Method 2 is given in Figure 7. Using a cooperative delivery scheme of this sort, a scalable video streaming service can offer non-dyadic temporal scalability using Method 2.

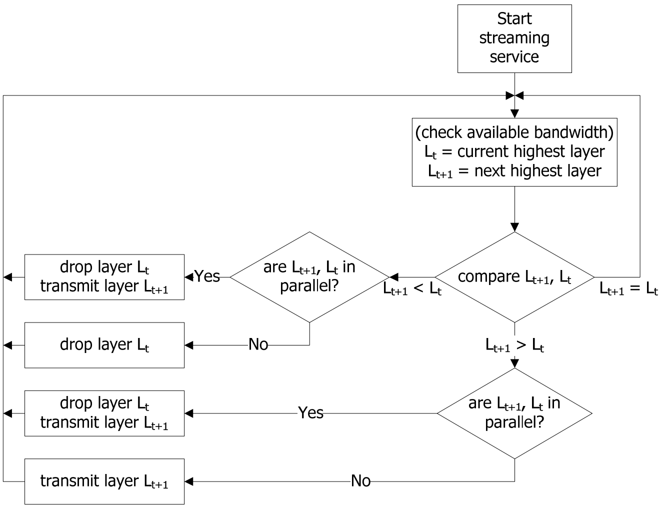

Figure 7. A delivery scheme used by routers that is capable of supporting Method 2.

B. Picture Quality

We encoded the 10 and 15Hz 'Harbor' sequences into 4 streams on the modified JSVM SVC encoder in the same way as in Section IV. The details of the streams encoded using Method 2 are in Table 3.

Table 3. Streams encoded using method 2.

Sequence	Spatial Resolution	Bit-rate (Kbps)	Temporal scalability	Temporal Resolution (Hz)
Harbor	352 x 288	199	Method 2	5, 10, 15
		270		
		322		
		377		

The picture quality was measured in the same way as in Section IV. We compared it with the quality of the hierarchical B-structured streams already itemized in Table 2. Figure 8 shows the results of the experiment.

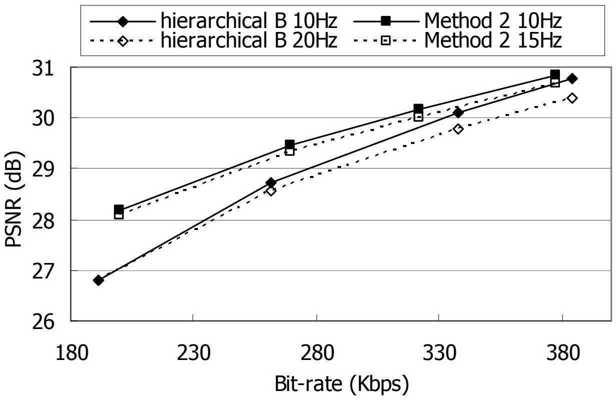

Figure 8. The averaged PSNRs of the streams.

Since the total number of frames in the two bit-streams is the same, and a similar coding structure is used, the picture quality of the two streams is similar. Note that the correlation between two neighboring frames in the hierarchical B-structured stream is larger because the time between two neighboring frames is smaller. This implies that the coding efficiency of the hierarchical B structure is better. The larger correlation and the extra coding overhead of the hierarchical B structure affect the streams of each bit-rate in different ways.

C. *Decoding Complexity*

The decoding complexity is measured in the same way as in Section IV and the results are shown in Figure 9.

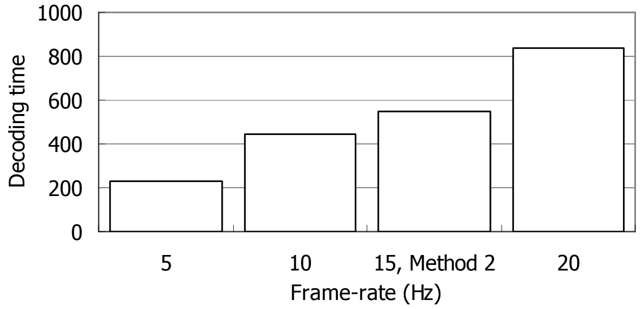

Figure 9. The decoding complexity of a stream generated using method 2 compared to a hierarchical B structured stream.

Method 2 reduces the complexity more than a reduction of frame rate, because it has less additional layer information than the hierarchical B structure. This decreased complexity can allow a low-performance device to select the appropriate temporal resolution from a set of varying resolutions.

VI. Conclusions

We have proposed two methods to enable non-dyadic temporal scalability with low complexity. Both methods can be implemented in the standard H.264 SVC extension, and easily merged into current streaming services. At the expense of a small amount of additional data, the decoding complexity can be reduced by up to 35%, offering a more flexible streaming service for low-performance devices. Recent rate distortion theory [12] suggests that we may anticipate good performance by merging our scheme into a middle layer of a hierarchical B structure, not into the highest layer. This extended temporal scalability can be easily and transparently built into the legacy temporal scalability.

References

[1] Information Technology—*Generic Coding of Moving Pictures and Associated Audio Information: Video*, ISO/IEC 13818-2-ITU-T Rec. H.262 (MPEG-2 Video), 1995.

[2] T. Wiegand, G. Sullivan, J. Reichel, H. Schwarz, and M. Wien, "Joint Draft 10 of SVC Amendment," Joint Video Team, Doc. JVT-W201, San Jose, CA, USA 21-27 April 2007.

[3] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable H.264/MPEG4-AVC extension," *IEEE International Conference on Image Processing*, Atlanta, GA, USA, 2006.

[4] I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*, John Wiley & Sons Ltd, 2003.

[5] M. van der Schaar and H. Radha, "A hybrid temporal-SNR fine-granular scalability for Internet video," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 11, March 2001.

[6] A. Ray and H. Radha, "Complexity-distortion analysis of H.264 JVT decoder on mobile devices," *Picture Coding Symposium*, 2004.

[7] J. Reichel, H. Schwarz, and M. Wien, "Joint Scalable Video Model JSVM-8," Joint Video Team, Doc. JVT-U202, Hangzhou, China 20-27 October 2006.

[8] G. Yadavalli, M. Masry, and S. S. Hemami, "Frame rate preferences in low bit rate video," *International Conference on Image Processing*, Barcelona, Spain, 2003.

[9] H.-C. Huang, W.-H. Peng, T. Chiang, and H.-M. Hang, "Advances in the scalable amendment of H.264/AVC," *IEEE Communication Magazine*, 2007, pp. 68-76.

[10] H. Li, Z. G. Li, and C. Wen, "Adaptive decoder complexity reduction for coarse granular scalability," *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Honolulu, Hawaii, 2007.

[11] C. An and T. Q. Nguyen, "Low complexity scalable video coding," *Asilomar Conference on Signals, Systems, and Computers (ACSSC)*, Pacific Grove, CA, USA, 2006.

[12] M. Dai, D. Loguinov, and H. M. Radha, "Rate-distortion analysis and quality control in scalable Internet streaming," *IEEE Transactions on Multimedia,* vol. 8, pp. 1135-1146, December 2006.