# An Approach to Worldly Quantifiability with Scalable Video Coding in H.264

### Murthy SVN

### Dept. Of CSE, Jain University

*Abstract* - In recent years, scalable video has gained momentum for a variety of reasons. One of the primary advantages is that scalable video provides more flexibility during transmission; it helps to adapt the bit rate midway during the transmission either in the server or in the network. Other advantages include error resilience through unequal protection and extremely fast transcode/transrate like operations.

This application report explains temporal scalability mechanisms that can be used in streaming or video conferencing systems. Temporal scalability refers to the ability to reduce the frame rate of an encoded bitstream by dropping packets, thereby, reducing the bit rate of the stream.

Key Words - SVC, H.264, Quantifiability, RTP Packet, AVC.

### I. TEMPORAL SCALABILITY IN H.264/AVC ANCED VIDEO CODING (AVC)

H.264/AVC introduced flexible coding structures to be signalled in the bit stream and Dyadic hierarchical coding structures became popular. Figure 1 and Figure 2 show hierarchical-B and hierarchical-P structures that are widely being used. The letter inside the box shows the picture coding type: Intra, predictive or bi-directionally predictive. Note that the Gop-Size is not the intra period but the minimum number of pictures that covers all the levels of prediction.



Figure 1. Hierarchical-B With Gop-Size = 4



Each set of pictures referenced on the same height is commonly referred to as a hierarchy level. The pictures in the lowest hierarchy level are sometimes referred to as the key frames.

### Dr. Sujatha B K

Dept. of Telecommunication, MSRIT, Karnataka, India

### A. Hierarchical Coding and Delay

B frame coding needs a reference from the future; therefore, the future frame has to be encoded before the encoding of the B frame can begin. Since there are more B frames, there are more delays in hierarchical-B coding. For example, in Figure 3, the encoder encodes the frame marked with 0 first, then it has to wait until the frame marked *I* is captured available for encoding; this causes a three frame delay in encoding. In 30 FPS encoding, one frame corresponds to 33 msec; therefore, three frames causes a 99msec delay. In general, the delay in hierarchical-B with a Gop-Size of N is N-1 frames worth of delay.



Figure 3. Encoding/Decoding Order for Hierarchical-B (shown inside blue ellipses)

There is no delay required in hierarchical-P as it does not need a reference from the future. The frames can be coded in the order that they are captured. Therefore, hierarchical-P is a low-delay coding structure and can be used in low-delay applications such as video conferencing.



Figure 4. Encoding/Decoding Order for Hierarchical-P (shown inside blue ellipses)

### B. Signalling of Temporal Scalability in H.264

A hierarchical structure can be divided into subsequences. Sub-sequence is the name used in the H.264/SVC specification without SVC. The concept of subsequence was introduced in to H.264/AVC long before full fledged scalability (Annex-G/SVC) was included; therefore, it has several restrictions. A better way of signalling temporal scalability is by using the syntax of SVC as explained in Section 2.1.

### Volume III, Issue VII, July 2014

### II. TEMPORAL SCALABILITY USING H.264/AVC ANNEX-G (SVC)

**IJLTEMAS** 

SVC also uses hierarchical coding to achieve temporal scalability. The only thing that SVC changes in temporal scalability is that it adds some optional signalling mechanisms into the bitstream. This does not refer to spatial/quality scalable bitstream, but a bitstream with only temporal scalability.

An SVC temporal scalable bitstream is completely decodable in an H.264/AVC decoder that does not support SVC. Therefore, SVC temporal scalability is completely backward compatible with H.264/AVC.

### A. Signalling of Temporal Scalability in SVC

1) *Prefix NAL Unit:* In SVC, the base layer (the lowest layer) is always H.264/AVC compatible, i.e., those slices use the exact SPS/PPS/Slice Header/Macroblock syntax as H.264/AVC.

But, the H.264/AVC NAL units do not convey the temporal layer that it belongs to. Due to this, a prefix NAL unit, a NAL unit that sits before the H.264/AVC NAL unit, is introduced to convey the layer indexes. It has a field called temporal\_id, which conveys the hierarchy-level that the current slice belongs to. Temporal layer is another term that is used to indicate a hierarchy level in the SVC context, but has the same meaning.



Figure 5. Temporal Layers in a Hierarchical-B Coded Stream

The prefix NAL unit is a NAL unit that sits before an H.264/AVC NAL unit to convey extra information including the temporal\_id. The prefix NAL unit can and will be ignored by a compliant H.264/AVC NAL decoder, even by an older decoder since it was a reserved value for H.264/AVC then.

NAL Unit Type = 14		temporal_id		NAL Unit Type = 1 or 5	
Prefix NAL Unit			H.264/AVC NAL Unit		

Figure 6	Prefix NAI	Unit Syntax	as Specified	by SVC
riguie 0.	FIGHX NAL	Unit Syntax	as specified	Uy SVC

The temporal\_id can be used by a bitstream extraction module to discard the associated NAL unit if necessary for frame-rate/bit-rate reduction.

2) Scalability Info SEI Message: This SEI message conveys sequence level information such as:

• How many layers are present?

• What are the frame-rate and bit-rate values for each layer, etc...

The scalability info SEI, along with the SVC prefix NAL units, can be used to help the bitstream extraction module to identify and remove the unwanted NAL units while it is trying to reduce the bit rate/frame rate.

## III. BITSTREAM TRIMMING/EXTRACTION PROCESS

All the above syntaxes were introduced in the bitstream in order for the extractor to be able to trim the bitstream, if needed. For example, Figure 7 shows one scenario where the extraction of lower layers, commonly referred to as bitstream trimming/extraction, might be required.



Figure 7. Bitstream Trimming in a Streaming/Conferencing Scenario

Receiver1 is getting a low frame-rate and, therefore, a low bit-rate stream, whereas, the Receiver2 is getting the stream as-is. This kind of trimming, where frame rate is reduced by dropping some temporal layers, can be called frame-rate trimming. Temporal scalability allows only frame-rate trimming; whereas, SVC in general allows other kinds of trimming as well.

The scalability info SEI message allows the router to obtain the information such as the resolutions (all resolutions are the same in the case of pure temporal scalability), frame rate and bit rates of the contained layers. It can then decide to remove or keep layers; removal of layers can make use of temporal id. For example, if the scalability info SEI messages indicate that there are three temporal layers; this means that the SVC stream contains packets with temporal\_id 0, 1, 2. It also indicates the frame rates associated with each temporal layer. The bitstream extraction module can choose to remove the higher temporal layers. The temporal id of a NAL/packet is indicated by the temporal id in its prefix NAL unit. For reducing the frame rate, the bitstream extraction module has to remove the packet and its associated prefix NAL unit.

Consider this example: If the bitstream extraction module removes all packets with temporal\_id equal to 2, the frame rate typically becomes half in a hierarchical coded stream. The scalability info SEI message has to be modified to reflect the result of the bitstream extraction process. For example, after removing all of the packets with temporal\_id equal to 2, the stream now contains only two temporal layers; therefore, the scalability info SEI message has to be modified to indicate this.

For offline examination and trimming of a temporal scalability stream stored in a file, a public domain tool called BitstreamExtractorStatic, part of the SVC JSVM reference code [8], can be used. The following is the information displayed by the tools for a typical temporal scalable stream:

### Contained Layers:

Layer	<b>Resolution</b>	Framerate	Bitrate	MinBitrate	DTQ
Õ	1280×720	6.2500	829.70	829.70	(0,0,0)
1	1280×720	12.5000	1054.70	1054.70	(0.1.0)
2	1280×720	25.0000	1289.30	1289.30	(0,2,0)

This shows that the stream contains three temporal layers and the details for each layer. When decoding/playing this stream, the highest layer (layer 2) is played.

### Contained Layers:

Layer Resolution Framerate Bitrate MinBitrate DIQ 0 1280x720 6.2500 B22.70 822.70 (0.00) For more information regarding the usage of this tool, see the SVC JSVM reference code documentation that can be downloaded from URL: http://wftp3.itu.int/av-arch/jvtsite/2009\_06\_London/JVT-AE013.zip [8].

### A. Typical Bit Rates of the Lower Temporal Layers

When reducing the frame rate by half, the bit rate typically does not become half. In the hierarchical structure, the references are far apart in the lower layers; therefore, they take more bits to encode. Also, it is a common practice to allocate more bits to the lower temporal layers, since they are used in prediction of several layers above; therefore, allocating more bits to the layers increases the overall quality of the encoded video.

### IV. VIDEO QUALITY IMPACT DUE TO HIERARCHICAL CODING

Several publications ([3], [4]) have shown that hierarchical-B frame coding provides significantly better coding gains compared to conventional IPP coding. Not only is this an improvement in objective (PSNR) sense, visual quality improvement is also observed. Hierarchical-P frame coding is also provided in low and medium motion sequences.

While it is true in the case of a high-end encoder, encoders with limited search range may not experience this quality improvement. This is because the key frames need a much larger search range, since its reference is quite far away.

If so much search range is not available, as in the case of an encoder in a low-cost device targeted for consumer market, there can be quality degradation for medium to high motion sequences. However, in the latest TI devices, this quality degradation is seen to be within acceptable range for a wide range of medium motion sequences.

### V. RTP PACKET FORMAT

RFC 3984 [6] specifies the RTP packet format used to transmit H.264/AVC in RTP format. The same format can be used to transmit SVC temporal scalability stream without any backward compatibility issues. The prefix NAL unit can either be sent as a separate NAL unit as in the *single NAL unit mode* or aggregated into an aggregation packet.

### VI. DECODED PICTURE BUFFER AND MEMORY MANAGEMENT

**IJLTEMAS** 

Decoded reference pictures are stored in the decoded picture buffer (DPB) as per the H.264/AVC standard. These pictures can be used for prediction of pictures that are going to be decoded. The decoder cannot store all of the decoded pictures in the DPB. At some point, when new decoded pictures are being stored into the DPB, older pictures have to be removed. The standard specifies mechanisms called *decoded reference picture marking process* to determine which picture to be flushed out of the DPB to create space for a new decoded picture being inserted into the DPB.

Hierarchical coding with a certain GOP-Size can be achieved using a variety of ways; it is desirable that the encoder uses a method that consumes the least DPB size at the decoder. While a variety of methods can be used, it was determined that using a long term reference frame syntax exclusively using the *adaptive memory control decoded reference picture marking process* is simple to implement and achieves the least DPB size at the decoder size.

Adaptive memory control decoded reference picture marking process uses what are called *memory management control operations* (MMCO) commands to specify how a decoded picture is to be marked. The method described below exclusively makes use of MMCO command 6, which marks a decoded picture as a long term reference, with a certain *LongTermFrameIdx*. A long term reference stays in the DPB until it is replaced by another decoded picture with the same *LongTermFrameIdx*. For more details, see [4].

LongTermFrameIdx values for hierarchical-B from progressive frame coding are shown inside the orange half-circles in Figure 8. Note that the highest level is not used for reference and does not need a LongTermFrameIdx.



Figure 8. LongTermFrameIdx Values for Hierarchical-B

LongTermFrameIdx values for hierarchical-P progressive frame coding are shown inside the orange half-circles in Figure 9. Note that the highest level is not used for reference and does not need a LongTermFrameIdx.



Figure 9. LongTermFrameIdx Values for Hierarchical-P

Each LongTermFrameIdx needs a separate storage location in the DPB; therefore, it is easily derived that the

### Volume III, Issue VII, July 2014

number of reference frames to be allocated in the DPB is 3 and 2, respectively, for hierarchical-B and hierarchical-P progressive frame coding in the case of GOP-Size 4. In general, hierarchical-B and hierarchical-P need L+1 and L reference frames, respectively; where L is the number of hierarchy levels: L = log2(Gop-Size).

### VII. ADVANTAGES AND DISADVANTAGES OF SCALABILITY WITH SVC TEMPORAL SCALABILITY

### A. Advantages

Temporal scalability provides the following advantages: • Bit-rate scalability at a gateway by trimming frame set

• Bit-rate scalability at a gateway by trimming frame-rate (this is already explained)

• Power scalability or MHz at the decoder by choosing to decode a lower frame-rate

• Backward compatible with existing H.264/AVC decoders – There is no need for an SVC decoder to decode an SVC temporal scalability stream. The additional header places in the stream for helping the extractor can be (and are expected to be) safely ignored by the H.264/AVC decoder. Temporal scalability is also the most simple to implement compared to other forms of scalability.

• Minimal or no loss in coding efficiency – By choosing to use temporal scalability, there is no loss in coding efficiency. In fact, for many platforms with adequate search range there can be gain in coding efficiency.

### B. Disadvantages

• Frame-rate reduction is one form of scalability and can be used, but losing frame-rate sometimes is not that visually pleasing. Compared to that, some users may prefer a smoothness or loss of detail due to decoding a lower resolution. But, for that, a more complicated and non-backward compatible SVC spatial scalability has to be used. Within the constraint of using H.264/AVC decoder, SVC temporal scalability is the best option.

• Another factor to consider is that when you reduce the frame rate to half, the bit rate doesn't typically drop to half; it may drop to about 3/4th. This is because the lower frame rate frames required larger bit rate to code since their reference frames are far away. Therefore, frame-rate scalability has a few limitations compared to spatial scalability.

#### REFERENCES

- Latest draft of H.264/AVC standard with Annex-G/SVC, JVT-AF11, http://ftp3.itu.int/av-arch/jvtsite/2009\_11\_Geneva/JVT-AF11.zip.
- [2] Explanation of Issues on H.264/AVC Sub-Sequences, MPEG/ITU-T Joint Video Team proposal document JVT-F094, http://ftp3.itu.ch/av-arch/jvt-site/2002\_12\_Awaji/JVT-F094-L.doc.
- [3] Overview of the Scalable Video Coding Extension of the H.264/AVC Standard, Heiko Schwarz, Detlev Marpe, and Thomas Wiegand, EEE Transactions on Circuits and Systems for Video Technology, Special Issue on Scalable Video Coding, vol. 17, no. 9, pp. 1103-1120, September 2007.
- [4] Hierarchical B Pictures, MPEG/ITU-T Joint Video Team proposal document JVT-P014, http://ftp3.itu.ch/av-arch/jvt-site/2005\_07\_Poznan/JVT-P014.zip.
- [5] Improved H.264/AVC Video Broadcast/Multicast, Tian, D. Kumar M V, V. Hannuksela, M. Wenger, S. Gabbouj, M, Proceedings- SPIE the International Society for Optical Engineering, 2005, VOL 5960; PART 1, pages 71-82.
- [6] RFC 3984 *RTP Packet Format for H.264 Video*, http://www.rfc-ditor.org/rfc/rfc3984.txt.
- [7] Proposed draft RFC for SVC *RFC Packet Format for SVC Video*, http://tools.ietf.org/html/draft-ietf-avt-rtp-svc.
- [8] "SVM Software, Text, and Conformance Status, http://wftp3.itu.int/av-arch/jvt-site/2009\_06\_London/JVT-AE013.zip.