TRANSCODING FROM H.264/AVC TO SVC WITH CGS LAYERS

Jan De Cock, Stijn Notebaert, and Rik Van de Walle

Ghent University - IBBT
Department of Electronics and Information Systems - Multimedia Lab
Gaston Crommenlaan 8 bus 201, B-9050 Ledeberg-Ghent, Belgium

ABSTRACT

In this paper, we show that it is possible to efficiently transcode single-layer H.264/AVC bitstreams to SNR-scalable SVC streams with CGS layers. Using requantization error compensation techniques, our architecture is able to restrain the drift that arises due to the absence of a closed prediction loop at the different dependency layers. Implementation results show that transcoding can generate SVC bitstreams with rate-distortion performance approaching that of the rate-distortion optimal encoder within 1 to 2 dB. Gains of more than 2 dB are obtained when compared to open-loop requantization.

Index Terms— Transcoding, requantization, SNR scalability, H.264/AVC, SVC

1. INTRODUCTION

In the context of Universal Multimedia Access, scalability of video bitstreams is an essential property, which capacitates the delivery of video content to a wide range of possible devices, irrespective of their bandwidth availability or computational or display capabilities. In many different scenarios and applications, rate adaptation is an important feature. Different techniques exist which enable rate adaptation, also known as SNR scalability.

Transcoding is a popular technique for adaptation of video content, allowing scalability without imposing constraints on the original bitstream, i.e., the bitstream does not have to be scalable to allow transcoding. In the past, different transcoding solutions have been presented, with architectures that provided such features as SNR, spatial, and temporal scalability [1, 2].

Currently, joint efforts of MPEG and VCEG are leading to the standardization of a new state-of-the-art scalable video codec [3]. This scalable extension of H.264/AVC, from here on denoted by SVC, makes it possible to encode scalable video bitstreams containing several dependency, spatial, and temporal layers. By parsing and extracting, lower layers can easily be extracted, providing different types of scalability.

A disadvantage of the paradigm used for scalability in SVC is that scalability has to be provided at the encoder side by providing different layers during encoding. This also implies that existing H.264/AVC-coded bitstreams cannot benefit from the scalability tools in SVC due to the lack of intrinsic scalability provided in the bitstream.

Transcoding, however, can be used for adaptation of video bitstreams that are coded using standard H.264/AVC compression tools. In [4, 5, 6], we presented techniques for requantization transcoding of H.264/AVC bitstreams. We have shown that the increased number of dependencies in H.264/-AVC bitstreams imposes a number of non-negligible issues for requantization transcoding. When compared to previous video coding standards, such as MPEG-2, the increased coding efficiency of new coding tools introduces the need for H.264/AVC-tailored transcoding solutions. In particular, attention has to be paid to the requantization transcoding and compensation of intra-coded macroblocks in P and B pictures in order to obtain acceptable video quality at the decoder [6].

Here, we extend these techniques to provide an architecture for H.264/AVC-to-SVC transcoding. Starting from H.264/AVC bitstreams with hierarchically B-coded pictures, we show that it is possible to transcode these streams to multilayer SVC bitstreams for combined temporal and SNR scalability. In this way, existing, previously encoded H.264/AVC bitstreams can be efficiently converted into bitstreams with inherent scalability layers that can be easily extracted at a later moment or in a further stage in the distribution chain.

Recently, a normative bitstream rewriting process [7] was added to the SVC specification, that allows converting an SVC bitstream with multiple CGS layers into a single-layer H.264/-AVC stream, i.e., the converse of the proposed transcoding operation. In the remainder of this paper, we discuss transcoding to SVC streams without this functionality.

The remainder of this paper is organized as follows. In Sect. 2, we briefly describe the SNR scalability mechanisms in SVC. In Sect. 3, we lay out our architecture for H.264/-AVC-to-SVC transcoding. In Sect. 4, we give implementation results for our architecture. Finally, we conclude in Sect. 5.

2. SNR SCALABILITY IN SVC

Different techniques exist in the Joint Scalable Video Model (JSVM) [8] for providing SNR scalability. Fine-Grain Scalability (FGS) uses an advanced form of bitplane coding for encoding successive refinements of transform coefficients. The design for FGS coding, however, is still under construction.

Coarse-Grain Scalability (CGS) uses techniques similar to the ones used for spatial scalability, the difference being that no upsampling is required between successive enhancement layers. In every layer, quality refinements of the transform coefficients are stored by using a decreasing quantization step size.

FGS SNR scalability has the advantage that it provides a larger degree of flexibility, allowing a quasi-continuous spectrum of achievable bitrates, while CGS is limited to a number of pre-determined bitrates. The layered design for CGS is shown in Fig. 1 for two dependency layers. In future applications, it is expected that the most commonly used number of dependency layers will be equal to two.

For more flexibility, Medium-Grain Scalability (MGS) was added to the Joint Draft [3]. MGS allows the use of up to 16 quality levels per dependency layer, hereby significantly increasing the number of achievable rate extraction points. Also, the MGS quality levels can be removed at any point at the bitstream, while switching between CGS dependency layers is only possible at pre-defined points in the bitstream.

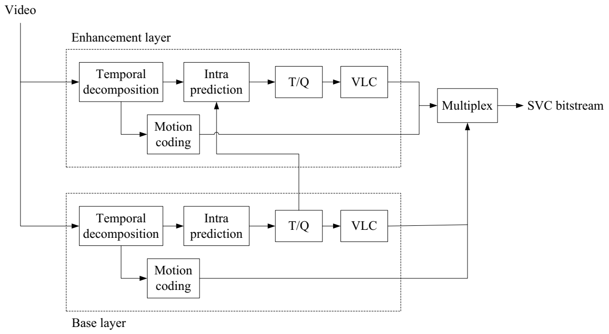

Fig. 1. JSVM design for 2 CGS layers.

More information about the SNR scalability tools in the SVC specification can be found in [8, 9].

3. TRANSCODING ARCHITECTURE

In order to obtain different quality layers after transcoding, the incoming transform coefficient values have to be requantized using their respective layer-dependent quantization parameter. Since a coarser quantization step size introduces errors, measures have to be provided in order to prevent drift in the base and enhancement layers of the SVC bitstream. To avoid this requantization error propagation, we use compensation techniques we developed for H.264/AVC in [6]. Depending on the slice and macroblock type currently being processed, a different compensation technique is applied. As we have determined in previous research, an important point of attention is the compensation of intra-coded macroblocks in P and B pictures. We use the low-complexity compensation techniques as discussed in [6].

In Fig. 2, the resulting architecture is shown for transcoding to SVC bitstreams with two dependency layers. The architecture can be easily extended for n dependency layers ($n \leq 8$). Depending on the slice and macroblock type, a distinction is made between spatial (intra-prediction based, S-TDC) and temporal (motion-compensation based, T-TDC) transform-domain compensation.

In the architecture, two buffers are provided. The first buffer contains the requantization error values from the current frame, and is used to compensate surrounding macroblocks according to the sparse compensation matrices we derived in [4, 5]. When a reference frame is transcoded, the content of the current frame buffer is copied to the reference frame buffer. The latter is used for temporal compensation of inter-predicted macroblocks. It is clear that the number of compensation frames used as a reference determines to a large extent the complexity and memory usage of the overall architecture. In order to retain a low-complexity transcoding architecture, compensation is only applied at the GOP borders. For a GOP length of 8, this implies that frames with a frame number equal to a multiple of 8 will be compensated. Intermediate hierarchically B-coded pictures are not compensated, which has only a minor impact on quality due to the low transform coefficient energy retained in the B-coded pictures. In this way, drift may arise within the GOP structure, but it will not propagate across the GOP borders. The method used here is similar to the key picture concept used in the JSVM for FGS and MGS SNR scalability. In this concept, key pictures prevent drift from propagating across GOP structure borders by only using the base layer as a reference for prediction.

A scalability information SEI message has to be provided at the beginning of the bitstream, containing information for every temporal and dependency layer present. This is necessary to allow fast bitstream extraction, so that NAL units can be dropped without the need to parse and decode more data than the NAL unit headers. Also, a NAL unit has to be added to the H.264/AVC base layer, which specifies the dependency and temporal layer id [3].

After coding the base layer, the second and subsequent layers can be obtained by subtracting the accumulated transform coefficient values of lower layers. In the JSVM, an inverse transform is applied between successive layers to decode the coefficients and perform calculations in the pixel domain. Here, for reduced complexity, we eliminate the inverse transform and perform calculations in the transform domain. This has only a minor effect on the rate-distortion performance of the transcoder. This is in fact the same method as will be used in order to support SVC-to-H.264/AVC bitstream rewriting.

In Sect. 4, we compare the performance of our proposed transcoder architecture to open-loop transcoding, i.e., when no spatial or temporal compensation is applied.

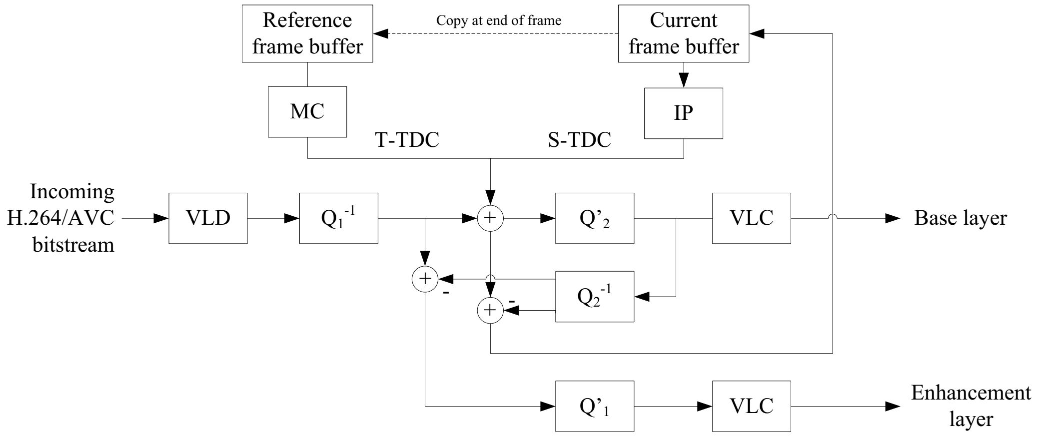

Fig. 2. Transcoding architecture.

4. IMPLEMENTATION RESULTS

In this section, we show the results from our implementation of the architecture described in Sect. 3. Test sequences with varying characteristics were used, namely Foreman, Stefan, and Paris (CIF). We encoded the sequences using the H.264/-AVC Joint Model reference software, using hierarchical GOP structures. We used GOP lengths of 8, 16, and 32, and an intra period of 32. We then transcoded the bitstreams to SVC with dependency layers. We used different values for the quantization parameters of the original bitstreams, namely 14, 20, 26, and 32. Tests were performed for two dependency layers with a typical $\Delta QP = 6$, obtaining SVC bitstreams with (QP_1, QP_2)-pairs equal to $(14, 20)$, $(20, 26)$, $(26, 32)$, and $(32, 38)$. For the results for reencoding, we started from the same H.264/AVC bitstreams mentioned above, which were first decoded, followed by an encoding step using the JSVM software, version 7.6.

The resulting rate-distortion curve for the Stefan SVC bitstream with enhancement layers is shown in Fig. 3 for all three techniques. As we can see, the transcoding solution with compensation is able to approach the rate-distortion optimal reencoding solution within 1 to 2 dB. It is clear that a major improvement occurs due to the compensation techniques, when comparing to open-loop transcoding.

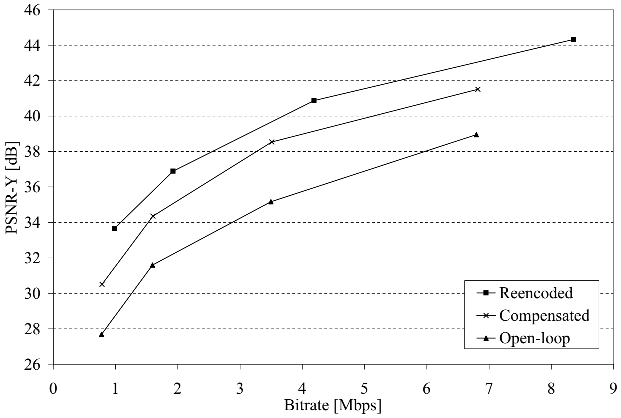

Fig. 3. Rate-distortion performance (Stefan sequence).

The same conclusions are obtained from the Foreman sequence, for which the rate-distortion performance is shown in Fig. 4, and for the Paris sequence.

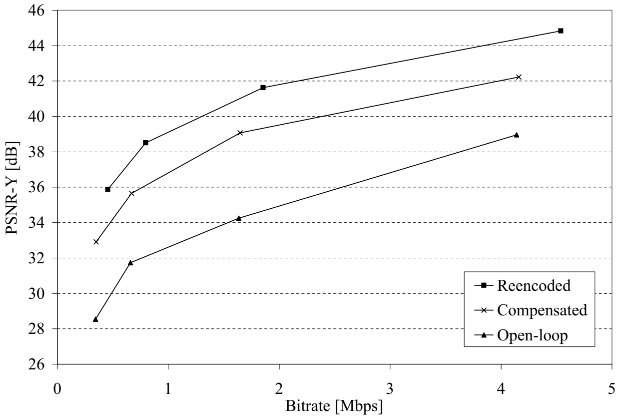

Fig. 4. Rate-distortion performance (Foreman sequence).

Similar results are observable when decoding the base layer only. The quality after transcoding with compensation approximates the rate-distortion optimal decoder-encoder cascade within 1 to 2 dB, as can be seen in Fig. 5.

In Fig. 6, the 64 first frames of the Stefan sequence are shown with their corresponding PSNR values after reencoding, transcoding, and open-loop bitstream rewriting. It is clear (particularly in the first intra period), that compensation is required in order to obtain reliable images and to restrain drift, and that the open-loop architecture as such is not applicable in H.264/AVC-to-SVC transcoding.

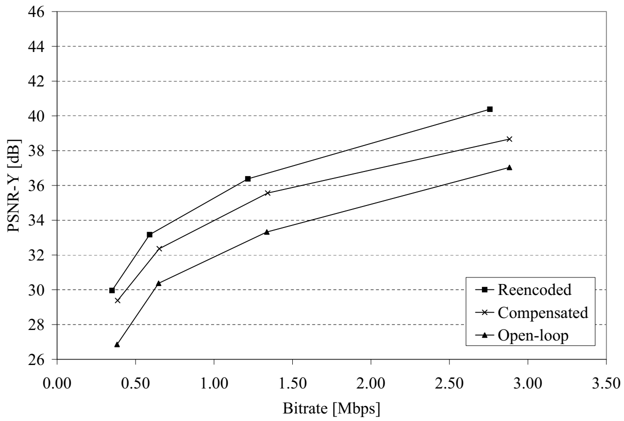

Fig. 5. Rate-distortion performance for base layer (Stefan sequence).

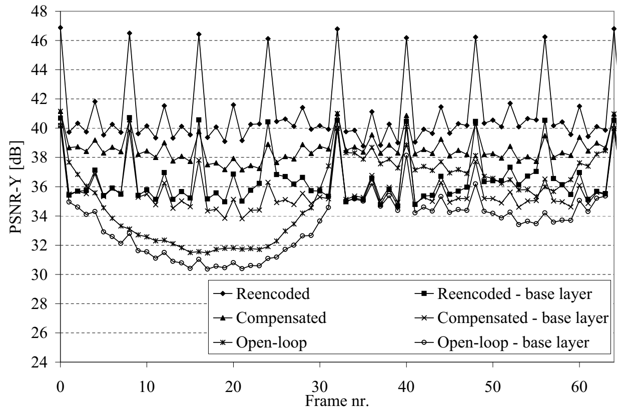

Fig. 6. Stefan sequence (64 frames).

As an indication of the computational complexity, it should be noted that a reduction in execution time of 90-95% was observed compared to the decoder-encoder cascade using the JSVM software. Further optimization, however, is possible for both the transcoder and the JSVM software.

5. CONCLUSIONS

In this paper, we have shown that it is possible to efficiently transcode single-layer H.264/AVC bitstreams to SNR scalable SVC streams with CGS layers. Implementation results were provided that show that the rate-distortion-optimal reencoder is approached within 1 to 2 dB. The open-loop architecture was shown not to be applicable in H.264/AVC-to-SVC transcoding, resulting in unpredictable video quality behavior due to drift, and overall losses of 2 to 4 dB when compared to the transcoding solution with compensation.

6. ACKNOWLEDGEMENTS

The research activities that have been described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT-Flanders), the Fund for Scientific Research-Flanders (FWO-Flanders), and the European Union.

7. REFERENCES

[1] Anthony Vetro, Charilaos Christopoulos, and Huifang Sun, “Video Transcoding Architectures and Techniques: an Overview,” *IEEE Signal Processing Magazine*, pp. 18–29, March 2003.

[2] J. Xin, C.-W. Lin, and M.-T. Sun, “Digital video transcoding,” *Proceedings of the IEEE*, vol. 93, no. 1, pp. 84–97, January 2005.

[3] Thomas Wiegand, Gary Sullivan, Julien Reichel, Heiko Schwarz, and Mathias Wien, *Joint Draft 10*, Joint Video Team, Doc. JVT-W201, San Jose, CA, USA, April 2007.

[4] Stijn Notebaert, Jan De Cock, Koen De Wolf, and Rik Van de Walle, “Requantization Transcoding of H.264/-AVC Bitstreams for Intra 4x4 Prediction Modes,” in *Proceedings of PCM (Pacific-rim Conference on Multimedia)*, 2006.

[5] Jan De Cock, Stijn Notebaert, Peter Lambert, Davy De Schrijver, and Rik Van de Walle, “Requantization Transcoding in Pixel and Frequency Domain for Intra 16x16 in H.264/AVC,” in *Proceedings of ACIVS (Advanced Concepts for Intelligent Vision Systems)*, 2006.

[6] Jan De Cock, Stijn Notebaert, and Rik Van de Walle, “A Novel Hybrid Requantization Transcoding Scheme for H.264/AVC,” in *Proceedings of ISSPA 2007 (International Symposium on Signal Processing and its Applications)*, 2007.

[7] Andrew Segall, *CE8: SVC-to-AVC Bit-stream Rewriting for Coarse Grain Scalability*, Joint Video Team, Doc. JVT-V035, Marrakech, Morocco, January 2007.

[8] Julien Reichel, Heiko Schwarz, and Mathias Wien, *Joint Scalable Video Model (JSVM) 10*, Joint Video Team, Doc. JVT-W202, San Jose, CA, USA, April 2007.

[9] Heiko Schwarz, Detlev Marpe, and Thomas Wiegand, *Overview of the Scalable Extension of the H.264/MPEG-4 AVC Video Coding Standard*, Joint Video Team, Doc. JVT-W132, San Jose, CA, USA, April 2007.