TRANSCODING OF H.264/AVC TO SVC WITH MOTION DATA REFINEMENT

Jan De Cock, Stijn Notebaert, Kenneth Vermeirsch, Peter Lambert, and Rik Van de Walle

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

ABSTRACT

In this paper, we present motion-refined transcoding of H.264/AVC streams to SVC in the transform domain. By accurately taking into account both rate and distortion in the different layers on the one hand, and the SVC inter-layer motion prediction mechanisms on the other hand, the proposed transcoding architecture is able to improve rate-distortion performance over existing approaches. We propose a multi-layer control mechanism that trades off performance between the different layers, resulting in 0.5 dB gains in the output SVC base layer.

***Index Terms*—** transcoding, rewriting, H.264/AVC, SVC.

1. INTRODUCTION

Recently, the scalable extension of H.264/AVC, commonly referred to as SVC [1], was finalized. SVC makes it possible to encode scalable video bitstreams containing several quality, spatial, and temporal layers. By parsing and extracting, lower layers can easily be obtained, hereby providing different types of scalability in a flexible way.

Although the majority of the content nowadays is coded in a single-layer format, it is beneficial for broadcasters and content distributors to have scalable bitstreams at their disposal to allow easy adaptation of the video streams. To achieve conversion from single-layer streams to scalable streams, efficient techniques for migration of existing content to a scalable format are desirable. As a low-complexity technique, transcoding can be used. Transcoding is a popular technique for adaptation of video content that does not impose constraints on the original bitstream, i.e., the bitstream does not have to be scalable to allow transcoding [2]. One of the main goals in designing transcoding architectures is to obtain an architecture that performs adaptation at a computational cost significantly lower than decoding and re-encoding, while achieving rate-distortion performance close to that of the cascaded decoder-encoder.

Most existing techniques focus on residual data transcoding [3, 4], i.e., without taking into account the motion data in the bitstream. In these schemes, residual data is distributed among the different layers, but all motion data is concentrated in the base layer. For larger reductions of the base layer bit rate, however, it is beneficial to also adjust the motion parameters, such as motion vectors, macroblock partitioning, reference picture indices, etc. By doing this, coarser motion information is included in the base layer while enhancement layers contain further refinements of the motion data.

In [5], we introduced low-complexity transcoding techniques based on bitstream rewriting [6]. Here, we extend these techniques to include motion refinement. We propose an architecture which operates completely in the transform domain and avoids the time-consuming steps involved in decoding and re-encoding. Adjustment of the motion parameters needs to be performed in a prudent way, since changed values could lead to misprediction during motion compensation. This could lead to significant distortion and artifacts which could propagate and cause drift in the video stream. Because of these reasons it is important to be able to reliably estimate the distortion introduced by changing motion parameters, and to provide an accurate model for rate-distortion trade-off in order to improve overall R-D performance.

Although working in the transform domain somehow limits the freedom of adjustment of motion parameters (large adjustments would incur significant errors), we show that our model for motion refinement can lead to a further reduction of the bit rate without causing distortion that would result in a negative net rate-distortion result.

Further, we provide a multi-layer control model that allows to trade off base layer vs. enhancement layer R-D performance. Hence, we provide liberty for our implementation to distribute motion data bits among the most appropriate layer.

2. MOTION DATA REWRITING

While the original motion information is optimized for the bit rate of the incoming bitstream (or of the top layer of the outgoing SVC stream), this is not necessarily the case for the lower layers of the output SVC stream. When the quality gap between successive layers becomes larger, it is likely that rate-distortion efficiency in the lower layers will benefit from a change in motion parameters. To accomplish this, we examine the potential rate-distortion gain of tweaking motion information for these lower layers. Since a change in motion information induces a change in the motion-compensated prediction signal, a careful examination needs to be made of the change in both the rate and distortion.

2.1. Motion refinement

H.264/AVC allows a large degree of flexibility in macroblock partitioning, with (sub)macroblock partitions down to 4×4 pixels. In lower-rate bitstreams, larger block sizes become more dominant, and the amount of submacroblock partitions tends to decrease. Hence, the most natural way of refining mode decisions for lower bit rates is by merging partitions, if the distortion introduced by the merging operation is small enough.

We examine in successive steps if macroblock partitions can be merged together. If two merged (sub)macroblock partitions use the same motion vector and reference index, no loss is incurred during the merging operation. If the merged macroblock partitions contain different motion vectors (which is typically the case), however, a mismatch arises and the introduced distortion needs to be estimated.

When merging macroblock partitions (8×8 and larger), special care has to be taken to avoid merging partitions that contain motion vectors pointing to different reference pictures. Reference picture indices can have a granularity down to 8×8 pixels (i.e., all submacroblock partitions within a single 8×8 block will refer to the same reference picture). If macroblock partitions with different reference indices would be merged, serious artifacts would arise in the decoded video stream, in particular when the temporal distance between the two reference pictures increases.

This problem is aggravated in B pictures, where different prediction directions can be used for each macroblock partition, i.e., reference pictures can be selected from different lists (forward prediction list, backward prediction list, or both). When bidirectional prediction is used, the partition is predicted based on a weighted sum of prediction signals.

Since merging partitions with different reference indices or prediction directions would cause artifacts in the transcoded bitstream, we avoid this situation, and only consider merging partitions with identical reference indices and prediction direction.

2.2. Rate calculation and distortion estimation

Different motion-related syntax elements in base and enhancement layer syntax contribute to the output motion data rate. For the base layer, the macroblock type and if necessary submacroblock types, reference picture indices, and motion vector differences need to be transmitted. If the macroblock is skipped, only a *macroblock skip run* (CAVLC entropy coding) or *macroblock skip flag* (CABAC) needs to be sent (one bit or less per skipped macroblock).

For the enhancement layer, a number of scenarios are possible. In case all motion information of a macroblock can be reused from the base layer, only the *base mode flag* is set and coded in the bitstream. If this is not the case, but a reliable approximation can be formed based on the base layer motion information, *motion prediction flags* can still be used to indicate that the reference indices can be copied from the base layer, and that a predictor can be formed based on the base layer. As an alternative, intra-layer motion vector prediction can be used to achieve the same result, and might result in improved coding efficiency in certain cases.

As shown in [7, 8], the distortion (D) introduced by motion vector variation can be estimated in the transform domain based on the picture power spectrum. We refer to [8] for the formulas, which can be obtained without additional overhead by approximating the FFT using the 4×4 integer transform coefficients in the input stream.

3. MULTI-LAYER CONTROL FOR H.264/AVC-TO-SVC REWRITING

During transcoding, we avoid loss of information, resulting in SVC streams which contain identical motion and residual data to the data available in the original bitstream. For the residual data, this is achieved by benefiting from the bitstream rewriting functionality in SVC [5]. For motion data, we use inter-layer motion prediction to efficiently redistribute the data over the different layers. For the top layer, the decoded motion data will be identical to the data found in the incoming single-layer H.264/AVC bitstream. This means that a reduction of the motion rate in a lower layer will lead to an increase of the bit rate in higher layers, resulting in a tradeoff between the different layers. The decision whether or not the evaluated refinement will be executed will depend on the impact of the rate and distortion in every layer. We use a multi-layer control mechanism which attaches a weight factor to every layer. The value of this weight factor depends on the scenario in which the rewriter is used. Based on the weight factors and the rate and distortion costs in every layer, we obtain formulas for joint optimization of both layers.

We examine the case for two layers, i.e., the base layer (indicated as *layer 0*) and one enhancement layer (*layer 1*); the discussion can readily be extended for three or more quality layers. *Base* layer coding decisions are made by minimizing

$$D_0(\boldsymbol{p_0}) + \lambda_0 R_0(\boldsymbol{p_0}),$$

where $\boldsymbol{p_i}$ encompasses the mode decisions m_i and motion vectors v_i for each layer i, respectively. This leads to the well-known functional used for rate-distortion optimized motion evaluation, as used for example in the JSVM encoder software. The Lagrangian multipliers λ_i are derived as in [9].

We additionally take into account the cost of the *enhancement* layer by also minimizing the enhancement layer distortion $D_1(\boldsymbol{p_1}|\boldsymbol{p_0})$ given the total bit rate $R_0(\boldsymbol{p_0}) +$

$R_1(\boldsymbol{p_1}|\boldsymbol{p_0})$ [10]. Weighting factor w is used to determine the trade-off between base layer and enhancement layer coding efficiency, leading to the cost functional

$$\min_{\boldsymbol{p_0},\boldsymbol{p_1}} (1-w)\cdot(D_0(\boldsymbol{p_0})+\lambda_0 R_0(\boldsymbol{p_0})) \\ + w\cdot(D_1(\boldsymbol{p_1}|\boldsymbol{p_0})+\lambda_1(R_0(\boldsymbol{p_0})+R_1(\boldsymbol{p_1}|\boldsymbol{p_0}))).$$

As mentioned, we examine the case where the motion information becomes identical to the information from the incoming bitstream when all layers are present in the SVC stream, i.e., no quality loss occurs after transcoding when no layers are dropped from the bitstream.

By following this approach, the distortion for the enhancement layer is eliminated, i.e., $D_1(\boldsymbol{p_1}|\boldsymbol{p_0}) = 0$, and the minimization problem becomes:

$$\min_{\boldsymbol{p_0},\boldsymbol{p_1}} (1-w)\cdot(D_0(\boldsymbol{p_0})+\lambda_0 R_0(\boldsymbol{p_0})) \\ + w\cdot\lambda_1(R_0(\boldsymbol{p_0})+R_1(\boldsymbol{p_1}|\boldsymbol{p_0})).$$

For $w = 0$, the functional reduces to the case where no joint optimization is performed, i.e.,

$$\min_{\boldsymbol{p_0}} D_0(\boldsymbol{p_0})+\lambda_0 R_0(\boldsymbol{p_0})$$

and only the base layer cost is minimized. In this case, base layer motion refinement will occur more frequently, since the cost of refinement bits is not taken into account. For $w = 1$, the expression

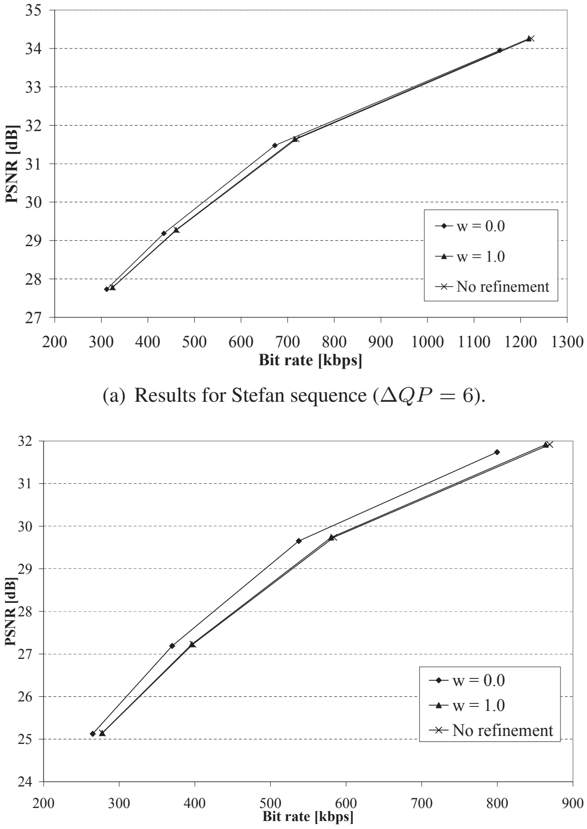
$$\min_{\boldsymbol{p_0},\boldsymbol{p_1}} R_0(\boldsymbol{p_0})+R_1(\boldsymbol{p_1}|\boldsymbol{p_0})$$

remains, under the side condition that reconstruction is identical when both layers are present in the bitstream. Typically, in this case, the optimum is achieved when all motion data is concentrated in the base layer, de facto corresponding to single-layer coding.

4. RESULTS

Several sequences were encoded using the Joint Model (single-layer) reference software, namely Foreman, Stefan, and Paris (CIF resolution). Hierarchical coding was used for the tests. We performed tests for two layers, i.e., one base layer and one enhancement layer. We used starting quantization parameters (QP_I) of 22, 27, 32, and 37. In order to cover typical use cases of SVC streams, we used ΔQP values of 6 and 12.

In Fig. 1(a), the rate-distortion results are shown for the base layer of the Stefan sequence, for $\Delta QP = 6$. By setting the enhancement layer weight to one (i.e., $w = 1.0$), the rate-distortion curve practically coincides with the curve without motion refinement. By setting the enhancement layer weight to zero ($w = 0.0$), rate-distortion performance is improved by approximately 5%, in particular in the lower bit rate range. For the highest rate point, a reduction of the bit rate is found (by 5.5%, from 1223 kbps to 1155 kbps) at a marginal gain in rate-distortion performance (the curve is located marginally higher for the higher bit rate range). These results correspond with the theoretical model and illustrate that although distortion increases somewhat by merging partitions, the motion refinement model only allows a merge if the rate reduction is large enough to improve overall rate-distortion efficiency.



(a) Results for Stefan sequence ($\Delta QP = 6$).

(b) Results for Stefan sequence ($\Delta QP = 12$).

Fig. 1. Base layer R-D results for Stefan sequence ($\Delta QP = 6$ and $\Delta QP = 12$).

In Fig. 1(b), the results are shown for the same sequence, but with a $\Delta QP = 12$ between the base and enhancement layer. As could be expected, a larger gap in quantization parameters (resulting in lower base layer bit rates) will lead to a higher degree of refined macroblocks in the stream. This leads to more potential for our motion-refined rewriting architecture, and gains of up to 0.5 dB. Overall bit rates are reduced by 5% for the lower bit rate range to 8% for higher bit rates. Similar results were obtained for the other sequences.

Results for the top layer are given in Fig. 2, showing the overhead of motion refinement. Note that, since reconstruction is perfect in all cases (when compared to the original single-layer stream), identical PSNR values are obtained for all RD points at a given QP. Hence, only the corresponding rate values are of interest in these charts. For $w = 1.0$, no overhead is incurred when compared to the case where no refinement is used and both curves practically coincide. On the contrary, the total bit rate is even somewhat reduced (but for all sequences $<$1%). This is caused by cases where inter-layer motion vector prediction is more efficient than regular H.264/AVC inter-layer motion vector prediction. When the weight of the enhancement layer diminishes, the total bit rate will slowly increase, leading to the curves of $w = 0.5$ and $w = 0.0$. This increase in bit rate corresponds with the rate-distortion model, which states that for low values of w, the base layer rate-distortion performance behavior is optimized without taking into account the overall bit rate. The more merging operations are performed in the base layer, the more information needs to be injected into the enhancement layer to reconstruct the original motion information. Since this introduces some redundancy in the bitstream (e.g., a macroblock type syntax element needs to be sent in both layers in case of refinement), the overall bit rate will start to increase.

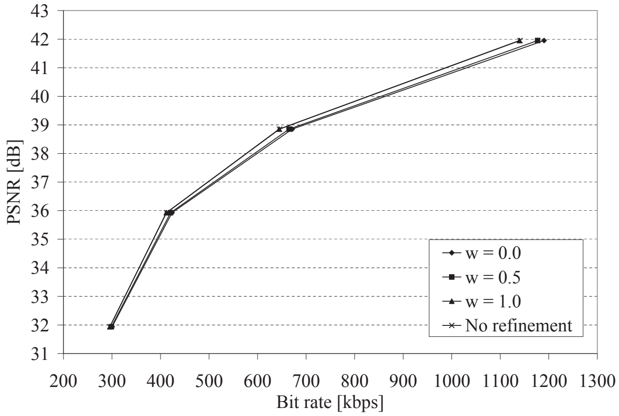


Fig. 2. Top-layer R-D results for Foreman sequence.

5. CONCLUSIONS

In this paper, we introduced a multi-layer transcoder control algorithm that provides a trade-off in rate and distortion between the considered layers. By setting the weight factors appropriately, the model allows rate-distortion performance to be improved for the desired layer(s). Even though operations are performed entirely in the transform domain, we have shown that distortion caused by motion refinement is accurately taken into account in the model. Although additional distortion is introduced due to changes in the motion data, our approach intelligently decides whether or not refinement in the motion data should occur, leading to an improvement in rate-distortion performance. In our implementation results, gains of up to 0.5 dB were obtained for the base layer.

6. ACKNOWLEDGEMENTS

The described research activities 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), the Fund for Scientific Research-Flanders (FWOFlanders), the Belgian Federal Science Policy Office (BFSPO), and the European Union.

7. REFERENCES

[1] ITU-T Rec. H.264 and ISO/IEC 14496-10 (MPEG-4 AVC), ITU-T and ISO/IEC JTC 1, *Advanced Video Coding for Generic Audiovisual Services*, Version 8 (including SVC extension): November 2007.

[2] A. Vetro, C. Christopoulos, and H. Sun, "Video transcoding architectures and techniques: an overview," *IEEE Signal Process. Mag.*, pp. 18–29, March 2003.

[3] A. Eleftheriadis and P. Batra, "Optimal data partitioning of MPEG-2 coded video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 10, pp. 1195–1209, October 2004.

[4] E. Barrau, "MPEG video transcoding to a fine-granular scalable format," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, September 2002.

[5] J. De Cock, S. Notebaert, P. Lambert, and R. Van de Walle, "Advanced bitstream rewriting from H.264/AVC to SVC," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, October 2008.

[6] A. Segall and J. Zhao, "Bit-stream rewriting for SVC-to-AVC conversion," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, October 2008.

[7] A. Secker and D. Taubman, "Highly scalable video compression with scalable motion coding," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1029–1041, August 2004.

[8] H. Shen, X. Sun, and F. Wu, "Fast H.264/MPEG-4 AVC transcoding using power-spectrum-based rate-distortion optimization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 6, pp. 746–755, June 2008.

[9] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 688–703, July 2003.

[10] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 533–545, Sept. 1994.