Video Compression Complexity Reduction With Adaptive Down-Sampling

Diogo C. Garcia, Tiago A. da Fonseca
Department of Electrical Engineering
Universidade de Brasília, DF, Brasil
Email: {diogo,tiago}@image.unb.br

Ricardo L. de Queiroz
Department of Computer Science
Universidade de Brasília, DF, Brasil
Email: queiroz@ieee.org

***Abstract*—Encoding video sequences is a computation-demanding task in high-performance codecs. Optimizing this stage may result in a substantial encoding speed-up. In this paper, we propose a faster approach to encode sequences with the H.264/AVC codec, using mixed-resolution. By having some of the frames down-sampled, the overall computation is reduced, without greatly affecting the rate-distortion performance, which remains evaluated using the sequence's original resolution. In addition to this approach, a method which regards only the most frequent prediction modes of a frame is incorporated, so that the combined complexity reduction can be significant with negligible performance losses.**

I. Introduction

H.264/AVC is the current state-of-the-art video compression standard, presenting excellent rate-distortion (RD) performance over a wide range of applications. However, it demands a large computation load, due to its wide range of compression tools [1].

Previous studies have shown that the temporal prediction module (Inter prediction) can be responsible for as high as 90% of the complexity of a H.264/AVC baseline encoder [2]. Several studies have proposed faster Inter prediction methods to speed up this process [2]–[5]. Huang *et.al.* proposed a criteria to reduce the number of reference frames, while other studies have chosen to find the most probable partitions [3],[4]. Fonseca and de Queiroz [5] analysed the most frequent modes in a portion of frame macroblocks in order to reduce the overall number of tested prediction modes for the macroblocks in the rest of that frame.

On a related but different approach, distributed video coding (DVC) makes use of the Wyner-Ziv theorem to create much simpler encoders, leaving to the decoder the task of enhancing the quality of the transmitted sequence [6]. DVC targets at encoding devices where memory and computation are scarce, usually at the expense of worse RD performance. Within this context, Mukherjee *et.al.* introduced a spatial scalability framework [7] using down-sampling of some of the frames by factors of two.

There have also been approaches to use down-sampling prior to the coding process, mainly for image coding schemes. Basically, the image is down-sampled by an appropriate factor and encoded. At the decoder side, the compressed stream is decoded and upsampled back to its original resolution. Zeng and Venetsanopoulos [8] successfully applied this method to the JPEG standard and showed the advantages for low to medium bitrates. Since then, a number of researchers have explored better interpolation methods to suit this particular application, based on the JPEG and JPEG 2000 standards [9]–[11].

In this paper, we make use of the principles of Zeng and Venetsanopoulos [8] to code video sequences with the H.264/AVC standard, in order to reduce the encoding complexity, without significantly increasing the bitrate. To do so, we use a mixed-resolution approach, where some frames are coded in full-resolution and used to determine the down-sampling ratios for the low-resolution frames. This paper is organized as follows: in Section II, we present preliminary tests to show this is a viable framework for the H.264/AVC codec, and in Section III, we present the proposed mixed-resolution framework in more detail. Section IV presents our results for a series of sequences, and Section V contains our conclusions and final remarks.

II. Interpolative Coding

For most sequences, a simple down-sampling by two incurs in large quality reduction, even if it is encoded in a lossless manner. However, if smaller down-sampling factors are chosen, this maximum limit can be increased, and more viable quality values can be reached.

This behavior is illustrated in Fig. 1, which presents the RD performance curves for the third frame of the 720p50-resolution sequence *Mobcal*, encoded with the JPEG standard at different resolutions. The chosen methods for down- and up-sampling were DCT-based ones. For down-sampling, the full resolution type-II DCT of each frame was calculated, and the highest frequency coefficients were discarded until the desired resolution was achieved. Next, the smaller resolution inverse type-II DCT was calculated, and all coefficients were multiplied by a scaling factor. For up-sampling, the opposite path was taken, where the missing highest frequency coefficients were made null.

It can be seen that, as we increase the down-sampling ratio, the rate-distortion performance deteriorates, as expected. Nonetheless, there are bitrates where the curves for different resolutions are close to the full-resolution one. For example, the 1024×576 resolution has similar performance as the full-resolution curve from the lower bitrates up until around 1.25

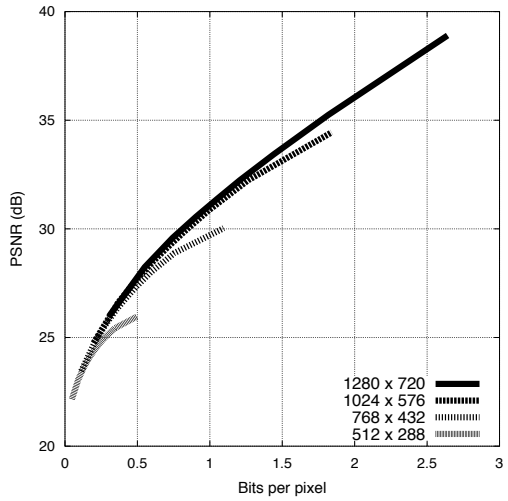

Fig. 1. JPEG rate-distortion performance for the third frame of the *Mobcal* sequence at full 720p-resolution and at smaller sizes - Y component.

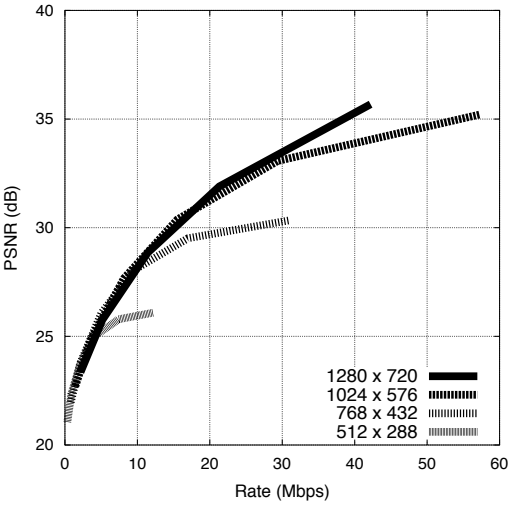

Fig. 2. H.264/AVC rate-distortion performance for the third and fourth frames of the *Mobcal* sequence at full 720p-resolution and at smaller sizes - Y component.

bits per pixel.

With this approach in mind, we carried some preliminary tests, to make sure these assumptions were valid for the H.264/AVC codec. First, we coded the same sequences in arbitrary resolutions besides their normal resolutions. Fig. 2 presents the RD performance for the third and fourth frames of the 720p-resolution sequence *Mobcal*, encoded with H.264/AVC in Inter mode, with one reference frame, with rate control and rate-distortion optimization disabled, and at different resolutions. The H.264/AVC reference software JM17.2 [13] was used for coding, and the same DCT-based down–/up-sampling methods were employed.

Figs. 1 and 2 show that the JPEG and H.264/AVC codecs present similar behavior, when using down-sampling prior to the encoding process. However, it is not clear how much of each frame's information should be disregarded under which circumstances. For instance, in both figures, lower resolutions can be chosen as the bitrate decreases. This indicates that the best down-sampling factor to be chosen depends on the amount of applied quantization (which controls the rate and the distortion), as well as on the sequence's information.

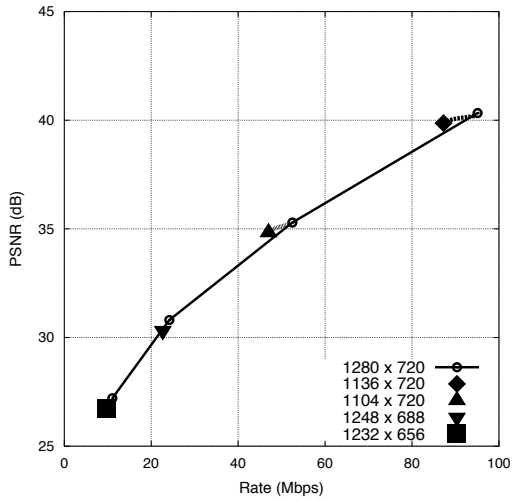

Fig. 3. Rate-distortion performance of ten frames of the *Parkrun* sequence at full 720p-resolution and at smaller sizes - Y component.

To verify that, we ran further tests, based in the following algorithm. Given a quantization parameter (QP) in H.264/AVC, we encode the whole sequence at full resolution, obtaining rate and distortion values R_{QP} and D_{QP}. We define a threshold T_{QP} as the distortion value plus a small variation Δ_{QP}, $T_{QP} = D_{QP} + \Delta_{QP}$. Next, we slowly increase the down-sampling factor and encode the down-sampled sequence under the same H.264/AVC parameters, until the corresponding distortion (after decoding and up-sampling) D_{QP+DWN} is equal to or smaller than T_{QP}. When that happens, we continue down-sampling only with respect to the smallest dimension (width or height), encode and calculate the distortion D_{QP+DWN}, until $D_{QP+DWN} \leq T_{QP}$. Then, we repeat the process with respect to the remaining dimension, until the algorithm terminates. In this manner, we guarantee that we will encode the sequence in smallest resolution not surpassing the distortion threshold.

Fig. 3 presents the results for 720p-resolution sequence *Parkrun*, frames 3 to 12, with all frames in P mode except for the first, one reference frame, QP = $\{22, 27, 32, 37\}$, with rate control and rate-distortion optimization disabled, and at different resolutions. We chose $\Delta_{QP} = -0.5$ dB. Clearly, there are smaller resolutions in which the sequence can be encoded without significant distortion loss. Of course, we had to run a series of tests for each QP at different resolutions, which actually tremendously increases the computation and renders the method impractical for complexity reduction.

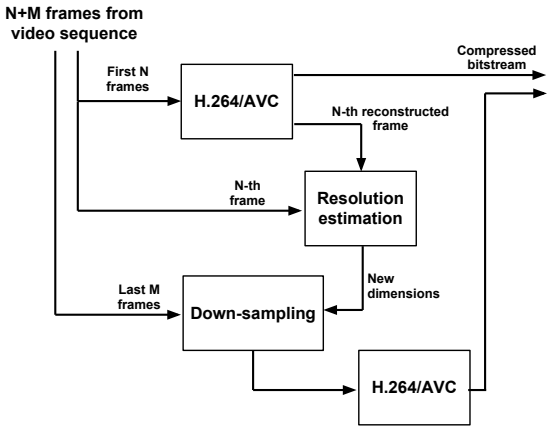

Fig. 4. Proposed mixed-resolution encoding framework

III. Proposed mixed-resolution framework

In order to estimate which resolution to use, we propose the following method. Video sequences are divided in groups of $N + M$ frames, and for these groups, we encode the first N frames at full resolution. Next, we pick the N–th reconstructed frame and compare it with its original version, from which we determine the resolution to code the next M frames. By using a previously coded frame, we can estimate the smallest resolution we can use, taking into account the sequence's information and the effect that quantization has over the sequence. By choosing adequate values for N and M, the system can adapt itself to small changes due to motion and to sudden overall changes, such as scene cuts. In order to obtain the reconstructed version of the encoded sequence, a regular H.264/AVC decoder is used for all frames, and the low-resolution frames are up-sampled to full resolution. Fig. 4 presents the proposed mixed-resolution encoding framework.

The resolution estimation method works as follows. After the first N frames of the $N + M$ group of frames have been encoded, the N-th decoded frame and its original version are chosen. We shall call these f'_N and f_N, respectively. Consider now that f'_N is down-sampled, creating frame f'_{ND}, and up-sampled back to full-resolution, creating frame f'_{NDU}. The estimated resolution for the M frames will be the smallest resolution of f'_{ND} such that $SSD(f_N, f'_{NDU}) \geq SSD(f_N, f'_N)$, where SSD is the sum of squared differences.

With this approach, we take into account the local characteristics of the input video sequence, as well as the effect of quantization over the quality of the decoded video. The resolution estimation analysis is done only on the N-th encoded frame in order to not extend the analysis too much, which would incur in an computation increase.

We used a DCT down-sampling method due to the fact that the DCT transform is an energy-preserving one. So, the SSD between a original frame and its decoded, reconstructed version is the same as the SSD between these two frames transformed by 2D DCTs, and so is the peak signal-to-noise ratio (PSNR). In this manner, we were able to work only in the transform domain, and save computation in the down–/up-sample process. It is also very convenient to resample images by fractional factors. For example, it is easy to downsize 720 rows to say 640 or 704 rows. If the number of rows or columns is K, we can test all down-sampling factors of K/S, S integer and $1 \leq S \leq K$.

IV. Experimental results

The H.264/AVC codec works with 16×16 macroblocks. If the input sequence dimension is not a multiple of 16, the codec makes a padding of the input frames until the dimensions reach multiples of 16. So, our framework works with S as a multiple of 16.

Since the codec works with Intra and Inter coding, its quantization effect is different in each case, because the predictions are different. With this in mind, we chose values for N that would coincide with the period of Intra frames. For instance, if the sequence is coded with one Intra frame for every four Inter frames, we have an Intra frame period of 5, and we choose $N = 5$. In doing so, the N-th frame used in the resolution estimation process is the last Inter frame before a new Intra frame.

For the M frames coded at a smaller resolution, we chose smaller QP values than those used in the N full-resolution frames. This prevents down-sampling and quantization from degrading too much the group of M frames, thus avoiding a large quality difference between the two groups. A ΔQP value of -2 was chosen in all tests and, for simplicity, $N = M$.

The proposed method was implemented in the H.264/AVC reference software JM 12.3, except for the processes of resolution estimation and down-sampling. These were implemented in a separate code, which was not optimized for performance. The FFTW 3.2.2 library [14] was used for the DCT-based interpolation methods.

The first 60 frames of each sequence were tested, and RD-optimization was enabled. UMHexS motion estimation was employed to search four reference frames for an IPPP GOP, and the results were obtained by varying the QP over the range $\{22, 27, 32, 37\}$. Three 720p50-resolution video sequences were used: *Mobcal*, *Parkrun* and *Shields*. In order to better visualize the incurred penalties, we chose to present tabulated results from the relative performance differences. Table I shows the performance comparison between the adaptive down-sampling method and the regular encoder. ΔR stands for the popular Bjøntegaard metric [15] of average bitrate increase between the regular coder and the proposed framework. ΔC stands for the average relative computation complexity savings, where the net encoding time of the full framework is taken as a complexity measure. ΔC_{Max} is the maximum achieved complexity saving.

In order to take into account how the Intra period length affects the proposed method, periods of 5, 10 and 15 frames were chosen, and their performance can be compared by analysing Table I. The longer the period, the worse the complexity savings, mainly due to the fact that the down-sampling ratio has to be small to preserve the sequence's visual

TABLE I
ENCODER PERFORMANCE USING ADAPTIVE DOWN-SAMPLING.

Sequence	*Mobcal*	*Parkrun*	*Shields*
Rel. Diff. [%]	Intra Period **5**		
ΔR	6.33	1.26	3.14
ΔC	19.9	23.2	22.8
ΔC_{Max}	24.2	26.7	26.0
Rel. Diff. [%]	Intra Period **10**		
ΔR	8.06	**-0.71**	3.48
ΔC	12.1	11.8	12.5
ΔC_{Max}	14.8	14.5	15.4
Rel. Diff. [%]	Intra Period **15**		
ΔR	7.98	**-0.14**	4.20
ΔC	9.26	9.57	10.5
ΔC_{Max}	11.1	12.2	13.9

quality. It is interesting to note that in some cases the proposed approach, besides being faster than the regular encoder, has a slightly better RD performance than the regular encoder (highlighted values).

To increase the complexity savings, the M down-sampled frames were encoded using only *dominant modes* [5]. Results are presented in Table II. Basically, this approach improves the computation savings when compared to Table I, without incurring in significant bitrate increase. The *dominant modes* approach seems to reduce the Intra period influence over the complexity savings; however, shorter Intra periods still present better RD performance.

TABLE II
ENCODER PERFORMANCE USING ADAPTIVE DOWN-SAMPLING AND USING ONLY DOMINANT MODES.

Sequence	*Mobcal*	*Parkrun*	*Shields*
Rel. Diff. [%]	Intra Period **5**		
ΔR	6.54	1.61	3.55
ΔC	33.3	36.6	34.2
ΔC_{Max}	36.6	38.7	38.1
Rel. Diff. [%]	Intra Period **10**		
ΔR	8.52	**-0.12**	4.49
ΔC	30.6	29.7	29.1
ΔC_{Max}	33.1	30.8	31.9
Rel. Diff. [%]	Intra Period **15**		
ΔR	8.65	*0.56*	5.69
ΔC	29.9	28.3	28.9
ΔC_{Max}	33.6	30.6	33.1

V. CONCLUSION

In this paper, we propose a complexity reduction framework for H.264/AVC based on spatial down-sampling. Instead of reducing the frame dimensions in powers of two, we use a rational down-sampling ratio. Such a factor is found adaptive to the video sequence content, so as not to greatly distort input frames, while reducing the amount of data to be analysed. Results show that the RD performance is little affected by the method for 720p50-resolution sequences, and there are cases where the approach could even surpass the original encoder RD-performance. The merge to *dominant modes* mode decision framework improved the complexity savings without affecting too much the incurred penalties. The proposed framework can benefit from more adequate down-sampling methods, which could even improve the original encoder performance.

REFERENCES

[1] T. Wiegand, G. J. Sullivan, G. Bjontergaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560-576, July 2003.

[2] Y.-Y. Huang, B.-Y. Hsieh, S.-Y. Chien, S.-Y. Ma, and L.-G. Chen, "Analysis and complexity reduction of multiple reference frames motion estimation in H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 4, pp. 507-522, April 2006.

[3] T.-Y. Kuo and C.-H. Chan, "Fast variable block size motion estimation for H.264 using likelihood and correlation of motion field," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 10, pp. 1185-1195, October 2006.

[4] B. Kim, S.-K. Song, and C.-S. Cho, "Efficient inter-mode decision based on contextual prediction for the P-Slice in H.264/AVC video coding," *IEEE International Conference on Image Processing*, pp. 1333-1336, September 2006.

[5] T. A. Fonseca and R. L. de Queiroz, "Complexity reduction techniques for the compression of high-definition video," *Journal of Communications and Information Systems*, Vol. 24, No. 1, pp. 1-10, Apr. 2009.

[6] B. Girod, A. Aaron, S. Rane and D. Rebollo-Monedero, "Distributed video coding," *Proceedings of the IEEE, Special Issue on Video Coding and Delivery*, vol. 93, no. 1, pp. 71-83, January 2005.

[7] D. Mukherjee, B. Macchiavelo, and R. L. de Queiroz, "A simple reversed-complexity Wyner-Ziv video coding mode based on a spatial reduction framework," *Proc. IS&T/SPIE*, Vol. 6508, pp. 65081Y1-65081Y12, Jan. 2007.

[8] B. Zeng, A.N. Venetsanopoulos, "A JPEG-based interpolative image coding scheme," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol.5, no., pp.393-396 vol.5, 27-30 Apr 1993.

[9] Y. Tsaig, M. Elad, P. Milanfar, G.H. Golub, "Variable projection for near-optimal filtering in low bit-rate block coders," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.15, no.1, pp. 154-160, Jan. 2005

[10] W. Lin, L. Dong, "Adaptive downsampling to improve image compression at low bit rates," *IEEE Transactions on Image Processing*, vol.15, no.9, pp.2513-2521, Sept. 2006.

[11] A.-M. Sevcenco, W.-S. Lu, "Combined Adaptive and Averaging Strategies for JPEG-Based Low Bit-Rate Image Coding," *Canadian Conference on Electrical and Computer Engineering*, vol., no., pp.168-171, 22-26 April 2007.

[12] X. Wu, X. Zhang, X. Wang, "Low Bit-Rate Image Compression via Adaptive Down-Sampling and Constrained Least Squares Upconversion," *IEEE Transactions on Image Processing*, vol.18, no.3, pp.552-561, March 2009.

[13] JM H.264/AVC reference software: http://iphome.hhi.de/suehring/tml/

[14] M. Frigo, S. Johnson, "The Design and Implementation of FFTW3," *Proceedings of the IEEE, Special issue on Program Generation, Optimization, and Platform Adaptation*, vol. 93, no. 2, pp. 216–231, Feb. 2005.

[15] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves.", *VCEG-M33*, Apr. 2001.