A Scene Change and Noise Aware Rate Control Method for VVenC, An Open VVC Encoder Implementation

Christian R. Helmrich, Christian Bartnik, Jens Brandenburg, Valeri George, Tobias Hinz, Christian Lehmann, Ivan Zupancic, Adam Wieckowski, Benjamin Bross, and Detlev Marpe

Video Communication and Applications Group – Fraunhofer Heinrich Hertz Institute (HHI), Einsteinufer 37, 10587 Berlin, Germany

***Abstract*—Contemporary motion picture content, consisting of scenes with different amounts of visual complexity or camera noise, represents demanding input for video encoders operating in rate control (RC) modes. This paper presents improvements to the 2-pass RC method integrated into VVenC, an open VVC encoder implementation, outlined in previous publications. We specifically introduce three extensions to our RC solution: first, frame type adaptation operating near scene cuts, along with an associated simple detector; second, rate stabilization means to allow for more reliable lookahead based 2-pass RC operation in *on-the-fly* encoding applications; and third, a low-complexity approach for estimating the instantaneous intensity of camera noise or film grain to avoid large variations in bit consumption when encoding individual frames in the final RC pass. Experimental evaluation confirms that these extensions significantly improve both the objective (BD rate) and subjective (visual) RC performance of VVenC especially on challenging video content.**

***Index Terms*—Film grain, H.266, rate control, VVC, VVenC**

I. Introduction

Rate control (RC) methods are indispensable in video coding when compressing motion picture content for distribution over the air, e.g. in streaming applications, or via traditional delivery networks, e.g. in television broadcasting. A two-pass RC design based on a novel rate-quantization parameter (*R-QP*) model and a simple yet effective perceptual model was recently introduced [1] and thoroughly evaluated [2]. This RC scheme is integrated into VVenC [3], an open encoder implementation for the H.266/ Versatile Video Coding (VVC) standard [4, 5, 6]. It was initially devised for *sequence-wise* operation, in which the entire video input is analyzed and pre-encoded in a first pass and, using the picture and coding statistics of this first pass, encoded a second time with full featured rate-distortion encoding and specific rate constraints to closely match the user defined target rate R_{target}. In addition, the two-pass RC method was primarily used with a relatively short Intra-only coded frame (I-frame)[1] period of 1s.

One drawback of short I-frame periods is a limited encoding performance since I-frames, consuming most of the bits needed for encoding of a group of pictures (GOP) due to the lack of any efficient motion compensation techniques, occur quite frequently. Longer I-frame periods, however, lead to diminishing returns in efficiency around scene changes, as will be illustrated herein, and more obvious visual artifacts, especially around scene cuts, upon packet loss on unreliable channels like wireless networks. With regard to sequence-wise two-pass RC, it must be noted that such an approach is not possible with live sources or commonly used pipe based software processing chains, in which the video sequence is not available to the encoder in its entirety a priori. Even when sequence-wise two-pass encoding is feasible, it was observed that motion picture content comprising short scenes of strongly varying statistics – like intros, credits, historical or dark material, modern and bright scenes, and very regular or irregular motion – may lead to RC instabilities (and, thus, reduced visual quality) when scenes with very different content occur in fast succession [7]. This issue can be attributed primarily to changes in film grain or sensor noise level, or no noise, between scenes.

A. New Contributions, Paper Outline

To address the abovementioned drawbacks with sequence-wise two-pass encoding and large I-frame periods, three extensions to the two-pass RC scheme of [1, 2] are described herein:

- a frame type adaptation (FTA) for use with long Intra periods, inserting additional I-frames at key-frame locations based on the output of a corresponding detector—described in Sec. II,
- a redesign of the two-pass RC scheme of [1] to allow for *on-the-fly* operation, by employing a sliding analysis window in the first encoding pass with a picture lookahead of one GOP, and better second-pass QP constraints—discussed in Sec. III,
- a noise level estimator using independent minimum statistics evaluation in eight luminance regions, to track the (possibly time varying) picture noise levels and adjust the definition of the final-pass QP cascades accordingly—outlined in Sec. IV.

The results of experimental assessments of these extensions in the context of VVenC, documented in Sec. V, reveal substantial improvements in encoding performance and RC stability, especially in case of lookahead based *on-the-fly* two-pass encoding of varying input. Sec. VI summarizes and concludes the paper.

[1] I-frames are needed as tune-in points in broadcasting or for seeking in file based playback

II. Frame Type Adaptation for Key Frames

Since the scalable extension of H.264 / Advanced Video Coding (AVC), hierarchical motion prediction structures with generally dyadic hierarchy stages [8] have been applied in all ITU / MPEG video coding standards. In such a picture coding structure, each GOP is segmented into different temporal levels $l \geq 0$, where an increase in l indicates a decrease in distance between successive pictures belonging to that temporal level. At the start (in coding order) or end (in display order) of each GOP, a *key frame* is encoded, as depicted in Fig. 1. Assuming that the Intra-only period I is an integer multiple of the GOP size G, every $I_G{}^{\text{th}}$ key frame, with $I_G = I/G$, will represent an I-frame, starting at the first encoded frame (here, at frame index $f_0 = 0$). All intermediate key frames will be predictively encoded P-frames, wherein motion compensation using the most recently encoded and decoded key frame as prediction source may be applied. To distinguish the P and I-frames, $l_P = 1$ and $l_I = 0$ shall be utilized for the former and latter, respectively. All other pictures at $l_B > 1$ are B-frames supporting bidirectional motion prediction, as also shown in Fig. 1. Due to the reduced distance between frames of higher temporal levels and their respective motion prediction sources (i. e., decoded reference pictures), and consequently more efficient motion compensation performance (i. e., prediction gain), high-l frames typically consumer fewer bits than low-l frames in video coding.

When scene changes characterized by camera switches (or simply *cuts*), fade-ins, fade-outs or cross-fades (or simply *fades*) are present between two key frames, the motion prediction performance in the later key frame will be marginal since the image statistics of the reference picture and the picture to be encoded will be very different. Although this may appear not to be much of a problem (since *Intra* coding may be used instead of motion predictive *Inter* coding by the encoder), the coding efficiency is often found to be suboptimal. Two reasons for this can be noted:

- the support for motion prediction causes signaling overhead, even–or especially–when such Inter coding is rarely or never used in a picture block (since, typically, it's used very often),
- the entropy coding tables for residuals of Inter coded blocks are generally not trained with videos including scene changes (as such rarely occurring events may deteriorate the training).

In addition, the configuration of the residual quantizer is usually quite different depending on whether the given frame is an Intra or Inter coded key frame, particularly with regard to the frame's overall quantization parameter QP_f and Lagrange parameter λ_f.

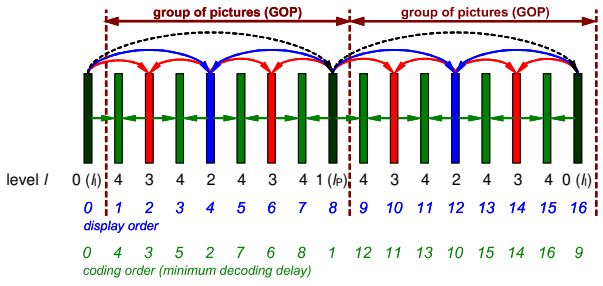

Fig. 1. Hierarchical coding structure for $G = 8$, $I_G = 2$, and 3 l_B values [8].

A. FTA Proposal, Algorithm Description

The previous discussion leads to the conclusion that, during a cut or fade between scenes of very different picture content, it may be advantageous to ensure that the first key frame after the scene change is an Intra-only I-frame. In spirit, such frame type adaptation (FTA), from type P to I, is similar to e.g. the approach in [9] but, in this paper, restricted to temporal levels $l \leq 1$ since, as mentioned, these low-l frames consume most of the bits. A simple detection algorithm for "very different picture content" is described hereafter. Of course, the FTA is only required when the first key frame after a scene change is not already an I-frame.

Let $f_P > G$ denote the index of the P-frame subjected to FTA detection and $f_M = f_P - G$ the index of the motion compensation source frame (i. e., reference picture index for the P-frame) with, consequently, $f_M > f_0$ and both f_P and f_M pointing to key frames. Before rate-distortion encoding, a minimal-complexity detector could simply subtract the uncoded input picture at f_M from that at f_P and calculate the ℓ^1 or ℓ^2 norm of the resulting sample-wise differences, which is then compared with a constant predefined threshold. If that threshold is exceeded, the frame type at index f_P is adapted from P to I and the quantizer configuration is modified accordingly before encoding the frame; otherwise, the type and quantizer are left unchanged, i. e., in a default P-type setting. It was observed, however, that FTA detection using a simplified picture difference is insufficiently reliable, especially when the level of image noise or amount of motion varies among scenes. Therefore, a more robust statistical measure of change, with still relatively low computational complexity, is desirable. Recently, a video quality assessment method called XPSNR was proposed [10,11] that uses a psychovisually inspired *perceptual sensitivity*

$$w_k = \sqrt{\frac{a_{\text{pic}}}{a_k}} \quad \text{with} \quad a_{\text{pic}} = D_1 \cdot 2^{2 \cdot D_2 - 9}, \quad D_1 = \sqrt{\frac{3840 \cdot 2160}{W \cdot H}}, \quad (1)$$

where k is the picture block index, W, H, and D_2 are the video width, height, and bit depth, respectively, and a_k, specified as

$$a_k = \max\left(a_{\text{min}}^2; \left(\frac{1}{4D_3} \sum_{[x,y] \in B_k} |h_s[x,y]| + 2|h_t[x,y]|\right)^2\right), \quad (2)$$

represents a spatiotemporal *visual activity* measure for the input block B (i. e., before encoding) of dimension D_3. The definitions of a_{min}, h_s, h_t are provided in [11] and omitted here for reasons of brevity. Note that h_t and h_s define temporal and spatial *partial activities*, respectively, for the given area of the picture at f. To improve performance, both are derived from spatially downsampled picture versions for UHD input [10]. Setting $k \overset{\text{def}}{=} f$ and $D_3 \overset{\text{def}}{=} W \cdot H$, an overall *FTA specific visual activity* for the luma picture at every f may be derived, particularly for both f_P and f_M:

$$a_t(f) = \max\left(a_{\text{min}}^2; \left(\frac{1}{4WH} \sum_{[x,y] \in B_f} |h_s[x,y]| + |h_f[x,y]|\right)^2\right). \quad (3)$$

Fig. 2 illustrates the ratio $a_t(f_P)/a_t(f_M)$ for all key frames of two HD sequences used by JVET [12] on a logarithmic scale. A ratio above T or below $\frac{1}{T}$ may be an indicator for a scene change (cut or fade) and, thus, FTA. Notice how, with a threshold of $T \approx 2^{1.5}$, both scene changes (marked) in the videos are detected reliably,

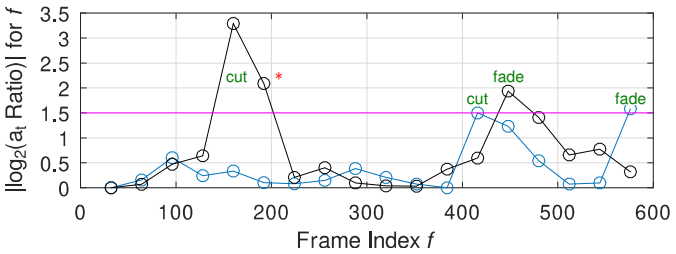

Fig. 2. $a_t(f)$ ratio for key frames in JVET HD sequences (–) *MarketPlace* and (–) *RitualDance* at GOP size of 32; (–) desired FTA threshold.

as long as detections in successive key frames are forbidden (see *). The assessed signal h_f in (3) represents a temporal high-pass that, compared with $h_t[x,y] \stackrel{\text{def}}{=} h_{t_f}[x,y] = s_f[x,y] - s_{f-1}[x,y]$ in (2) and [11], is downsampled by G in temporal direction of f:

$$h_f[x,y] = 2\big(s_f[x,y] - s_{f-G}[x,y]\big), \tag{4}$$

where s holds the luma picture samples at the given frame index and x, y are the horizontal, vertical sample indices, respectively. This 1:G downsampling allows for relatively stable detection of short scene fades in addition to cuts, which is not possible with a visual activity measure like the one of (2) used by the XPSNR.

III. GOP-wise Two-Pass RC with Sliding Window

To address the infeasibility of sequence-wise two-pass encoding in *on-the-fly* applications, the RC scheme of [1] can be operated in a GOP-wise fashion, i. e., with the target rate representing an *instantaneous* second-pass constraint. In other words, an instantaneous set of first-pass encoding statistics, determined inside a sliding temporal analysis window A, may be evaluated for each new GOP to derive the frame-wise overall QP_f and λ_f values for second-pass rate-distortion optimized encoding of that GOP. To stabilize the RC, the size for A is chosen such that it extends at least one GOP into the past (except at the beginning of the video sequence) and one or two GOPs into the future (i.e., towards the newly incoming picture data, except at the end of the sequence). The latter range of frames in A represents the RC's lookahead.

A. Modifications to Two-Pass RC Algorithm

The lookahead based RC integrated into VVenC, following the considerations above, works as follows, where only differences to the sequence-wise two-pass RC of [1, 2] are described. First, a second-pass average base QP required for e. g. loop filter initialization and QP delta-coding, named QP''_{base}, is estimated:

$$QP''_{\text{base}} = \text{round}\left(QP'_{\text{base}} + c_{\text{high}} \cdot \max(0; 24 - QP'_{\text{base}})\right) \tag{5}$$

with $QP'_{\text{base}} = \text{round}\big(40 - 1.5D_1\sqrt{R_{\text{target}}/500000} - 0.5\log_2 I_G\big)$ and D_1 as in (1). Using only R_{target}, in bps, and D_1 in the empirically devised (5) is necessary since no sequence-mean first-pass data are available when setting QP''_{base} at the start of the RC pass.

Next, each new GOP is encoded, in a very fast configuration [2], in the analysis pass using the same QP_{base} as in [1], yielding first-pass QP_f and rate r_f statistics for all f in the lookahead range (a lookahead of only one GOP was chosen to limit memory use). Then, the first-pass statistics for all $f \in A$ are collected, where a size for A of $\min(8G; I) + G$ frames is used. Hence, the statistics of the last Intra period (limited to at most $8G$) of previously encoded frames are added to the GOP of lookahead statistics, and with typical $I \leq 8G$, the sliding window will cover $I_G + 1$ GOPs.

Finally, an instantaneous first-pass rate R_A (from frame-wise bit counts r_f) and a target rate R'_A (from R_{target} and frame rate fps in Hz) are derived from all frames in A. Using these two values, frame-wise second-pass target bit counts r'_f are determined as in [1, Sec. IV.A] and, with the help of the first-pass frame (or *slice*) QP values QP_f, the second-pass frame QP values QP''_f for rate-distortion optimized encoding of each frame are obtained therefrom, using the two-step *R-QP* model introduced in [1, Sec. II].

B. Improved Limiting of Changes in QPs

The *R-QP* based sequence-wise RC implementation of [1] in VVenC was found to work quite reliably on statistically homogeneous video signals. Mixed material with varying amounts of camera or content motion, sensor noise or film grain, and other visual activity features such as contrast, can destabilize the RC's operation. This problem, occurring especially when the features change abruptly upon scene changes, is much more pronounced in the lookahead restricted RC design of Sec. III.A, occasionally leading to severe visual artifacts due to high QP''_f (and therefore, very coarse second-pass quantization) after some scene changes.

To address this issue, the RC method in VTM [13, 14] limits the amount by which the QP and λ parameters may vary between successively encoded frames. Specifically, in each new frame f, the QP''_f may change by at most ±10 compared to the previously encoded (in coding order) frame's QP and by at most ±3 relative to the QP of the previously encoded frame at the same temporal level l (where equal values of l_P and l_I are assumed). Likewise, the λ_f may vary by at most a factor of $2^{\pm 10/3}$ or $2^{\pm 3/3}$, respectively.

For the sequence- and GOP-wise two-pass RC in VVenC, the above change constraints were refined and extended as follows:

- QP''_f at level l shall lie in range $[QP''_{\text{curL}} \pm c_{\text{curL}}]$, where QP''_{curL} is the QP of the last coded frame at the same l; $c_{\text{curL}} = 5 + I_G$ during a scene change and $c_{\text{curL}} = \max\big(3; 6 - \lfloor \frac{l}{2} \rfloor\big)$ otherwise.
- QP''_f at level l shall be larger than QP''_{prevL}, the QP of the last coded frame one level below l (i. e., at level $l-1$), with $l > 1$.
- QP''_f at level $l \leq 1$ shall lie in range $[1 + QP_{\text{avg}}/2, QP_{\max}]$, with QP_{avg} holding the mean of all past QP''_f in A and $QP_{\max} = 63$.
- QP''_f at level l shall lie in range $[l + QP''_{\text{base}}/2, QP_{\max}]$, with the second-pass base QP estimate QP''_{base} defined as in (5).

In addition, VVenC's functionality to better match the target bit rate as the encoding progresses, as described in [1, Sec. IV.A], was attenuated in the first encoded GOP (d was reduced to ¼). Along with the above five QP limiters, implemented in VVenC function clipTargetQP( ) [4], these constraints stabilize both the sequence-wise and GOP-wise RC in case of highly variant input statistics. Moreover, they outperform the simpler QP, λ clipping operations in VTM's RC algorithm [13], as illustrated in Fig. 3.

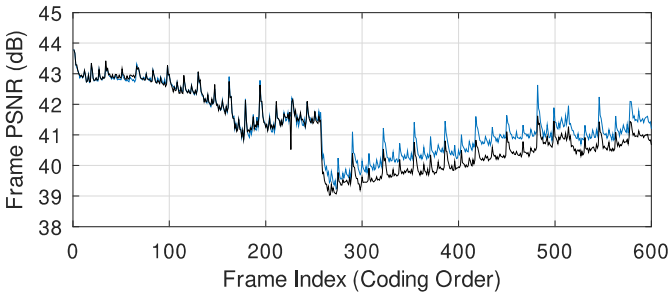

Fig. 3. GOP-wise two-pass RC encoding of HHI 4K sequence *Oberbaum* [15] with VVenC and QP change limits of (–) Sec. III.B, (–) VTM [13]. R_{target} = 813 kbps, VVenC preset = 'slow', G = 32, and I = 64.

IV. Noise Level Estimation and Limiting of QPs

The first pass in two-pass RC encodings is, as already noted in Sec. III, operated in a very fast configuration. More specifically, to acquire the QP and rate statistics, this analysis pass performs rate-distortion encoding with a relatively high average QP of [1]

$$QP_{\text{base}} = \text{round}\big(40 - D_1 \cdot \sqrt{R_{\text{target}}/500000}\big) \qquad (6)$$

and, thus, coarse residual quantization. In addition, CU size and coding tool constraints (CTU size of 64×64 samples, only 32×32 or larger CUs allowed, and most VVC tools and loop filters except for MCTF [16] disabled) are enforced. While this approach lowers the first-pass runtime to a fraction of the second-pass RC runtime, it occasionally destabilizes the RC due to unexpectedly high frame bit counts occurring in isolated frames in the second encoding pass. The reason for such spikes in bit consumption is a specific combination of quantization step-size (as specified by the QP) and noise level within the picture (as a result of camera sensor noise or film grain) at a given frame index f. Specifically, the RC encoding may apply, in a picture or CTU at f, a step-size leading to non-zero quantization of picture noise, thus resulting in excessive rate due to the high entropy–and unpredictability–of such random components. Due to higher QP_f and step-sizes, however, the analysis pass may not have "seen" such r_f bloating.

Countermeasures against this effect are to increase D_1 in (6), thus reducing the first-pass base QP, or to perform MCTF on all frames, thus denoising all f in the video a priori, but both options cause higher computational complexity and runtime during RC encoding. The following alternative solution, adapting the residual quantizer as the origin of the issue, was therefore pursued.

Again, let k indicate the CTU index within some frame f. If the sensor noise or film grain level L in video block B_k is known after the first pass, or at least estimated with sufficient accuracy, the CTU's second-pass QP'_k and, thereby, quantization step-size Δ'_k may be limited proportionally to L to prevent large bit counts:

$$QP''_k = \max(QP^*_k; QP'_k) \quad \text{s.t.} \quad \Delta''_k = \max(L; \Delta'_k). \qquad (7)$$

Second-pass quantization applied this way ensures that undesirably high SNR of noisy spatiotemporal picture regions will not occur; the SNR in these regions will not exceed a few dB. More efficient reconstruction of noisy "background" signals in images can, in any case, be realized through parametric coding [17, 18].

A. QP Noise-Limiter, Algorithm Description

Noise level adaptive block-wise QP limiting prior to second-pass residual quantization can be implemented as follows. First, a time-varying estimator for L is required since different scenes in a video could exhibit different amounts of noise. A *minimum statistics* (MS) estimator [19] applied in a GOP-wise fashion and in 8 separate luminance regions was found to work well for the use case at hand. In the RC's analysis pass, the MS estimator is reinitialized in the starting (in coding order) key frame of every new GOP (i. e., lookahead) and, once that GOP has been coded, the eight values of L are provided to the second-pass quantizer. The CTU-wise update of each noise estimate $L(Y)$, $0 \le Y < 8$, is carried out via (2) by obtaining, for all k in the GOP, the minima

$$n_k = \tfrac{1}{4D_3} \cdot \min\big(\textstyle\sum_{[x,y]\in B_k} |h_{\text{s}}[x,y]|\,;\ \sum_{[x,y]\in B_k} |h_{\text{t}}[x,y]|\big). \qquad (8)$$

Note that noise level estimate n_k for block B_k is readily available from (2) without additional sample-wise operations. With index

$$u_k = \Big\lfloor 8 \cdot \tfrac{\mu_k}{2^{D_2}} \Big\rfloor = \Big\lfloor \tfrac{2^{3-D_2}}{D_3} \cdot \textstyle\sum_{[x,y]\in B_k} s[x,y] \Big\rfloor \qquad (9)$$

parametrizing the mean luma value in B_k, L can now be updated: if $L(u_k) > n_k$, then $L(u_k) \overset{\text{def}}{=} n_k$, otherwise $L(u_k)$ is left as is.

Then, for each block B_k encoded in the second RC pass, a QP limit QP^*_k corresponding to step-size $\Delta^*_k = L$ can be determined:

$$QP^*_k = \big\lfloor \alpha + 6 \cdot \log_2\big(L'(u_k)\big) \big\rfloor, \qquad (10)$$

where α is a D_2 and VVC specification dependent constant and luminance index u_k is defined as in the first-pass (9) above (note that these values may be transferred from the first RC pass). For more reliable results, L' is used instead of L in (10) and given by

$$L'(Y) = L^*(Y) \ \text{ if } \ L(Y) > L^*(Y), \ \ L'(Y) = L(Y) \ \text{ otherwise, } \qquad (11)$$

where $L^*(Y) = \max\big(L(Y-1); L(Y+1)\big)$ if $0 < Y < 7$, otherwise $L^*(Y) = L(Y)$. Using (10) and (11) with u_k, the QPs used during block-wise second-pass quantization, QP''_k, are obtained via (7). Parameter α in (10) can be used to control the resulting SNR in noise-like picture regions. This is an aspect left for further study.

V. Performance Evaluation on top of VVenC

Two encoding experiments were conducted to evaluate the three contributions described in this paper. The first, performed using fixed-QP coding without RC, assesses the merit of the proposed FTA of Sec. II whereas the second, carried out in GOP-wise RC configuration, intends to verify stable behavior of the lookahead based RC design as well as the benefit of its extensions of Secs. III and IV. The baseline software for this study is VVenC 1.6.0 [4], the version into which the contributions were integrated, in preset "slow". As in [1], both experiments were run in a random access (RA) setup with G = 32 frames, D_2 = 10 bit, and the non-normative MCTPF tool on [16], i. e., in accordance with JVET's common test conditions for SDR video [12, 13]. To reflect more realistic use, however, the sequence duration in UHD class A was extended to 10 s and HHI's *Berlin* sequences [15] were added.

TABLE I. BD-rate comparison of VVenC run with different Intra periods I.

Resolution	**Increase of I**	**Activation of (F)TA**	**Overall Change**
Class	$I = 1$ s vs. $I = 4$ s	$I = 4$ s vs. $I = 4$ s + F	$I = 1$ s vs. $I = 4$ s + F
UHD A½	– 2.07%	– 0.00%	– 2.07%
UHD HHI	–11.86%	– 0.00%	–11.86%
HD B	– 1.88%	– 1.93%	– 3.95%
HD HHI	–18.38%	– 0.00%	–18.38%
SD C	– 4.46%	0.06%	– 4.40%
Overall CTC	– 2.64%	– 0.63%	– 3.32%
Overall HHI	–15.12%	– 0.00%	–15.12%
MarketPlace	**8.24%**	– 9.27%	– 1.74%

The Bjøntegaard Delta-rate (BD-rate) results of the first experiment, obtained according to [20] on 6:1:1 YUV averaged PSNR statistics, are provided in Tab. I. They indicate substantial gains in compression efficiency when increasing the Intra period (here rounded to the nearest integer multiple of G for each sequence), except on class B. The culprit of this phenomenon, the *MarketPlace* video with its two scene changes (see also Fig. 2), exhibits a BD-rate loss of 8.2% when moving to $I = 4$ s. Fortunately, the FTA proposal of Sec. II fully eliminates this problem, yielding a BD-rate gain of 9.3% on that sequence and, thereby, ensuring that $I = 4$ s is never worse—but often much better—than $I = 1$ s. The moderate improvement on class A is a topic for future study.

The BD-rates for the second experiment, derived from YUV averaged XPSNR data for VVenC as that encoder was run with XPSNR based block-wise perceptual QP adaptation [11, 21], are listed in Tab. II. As in [1], the sequence-wise rates produced by fixed-QP CTC compliant RA encoding with the respective software (VVenC or VTM) were used as R_{target} in the RC. c_{high} in (5) and all other previously undefined parameters were chosen as in [1]. For the HHI videos [15], the VTM results in [1] were taken. Tab. II indicates a notable performance advantage of the present lookahead RC proposal over the single-pass RC design adopted in VTM, the only other *on-the-fly* RC solution for $G = 32$ frames known to the authors [22]. In fact, the lookahead RC approaches the performance of the sequence-wise RC, which is remarkable.

VI. Summary and Conclusion

Three extensions to the two-pass rate control method in the open VVC encoder VVenC were described and assessed. Frame type adaptation allows for efficient usage of long Intra periods while noise level and lookahead aware GOP-wise operation, using the proposed improved second-pass QP constraints, enables the use of VVenC in practical *on-the-fly* video encoding applications.

TABLE II. BD-rate performance of RC modes in VVenC 1.6 and VTM 14.0. $I = 4$ s for VVenC; BD-rate and runtime references: fixed-QP runs.

Resolution	**VTM 14, no QPA**		**VVenC, GOP-wise**		**VVenC, seq.-wise**	
Class	$I = 1$ s	Runtime	III.B off, **on**	Runtime	IV.A off	Runtime
UHD A½	**10.04%**	98.1%	2.40%, **0.65%**	102%	–0.46%	104%
UHD HHI	**9.13%**	101%	5.18%, **3.46%**	103%	4.43%	105%
HD B	**5.45%**	105%	4.60%, **1.31%**	104%	0.63%	105%
HD HHI	**14.0%**	109%	7.79%, **4.52%**	103%	6.26%	105%
SD C	**4.02%**	102%	5.47%, **2.10%**	101%	0.91%	103%
Overall CTC	**6.91%**	102%	3.95%, **1.26%**	102%	0.27%	104%
Overall HHI	**11.6%**	105%	6.48%, **3.99%**	103%	5.34%	105%

References

[1] C. R. Helmrich, I. Zupancic, J. Brandenburg, V. George, A. Wieckowski, and B. Bross, "Visually optimized two-pass rate control for video coding using the low-complexity XPSNR model," in *Proc. IEEE Int. Conf. VCIP,* Munich, Germany, 2021. www.ecodis.de/ratecontrol.htm.

[2] I. Zupancic, C. R. Helmrich, J. Brandenburg, V. George, C. Bartnik, A. Wieckowski, B. Bross, and D. Marpe, "Visually Optimized Rate Control Methods for a Practical VVC Encoder Implementation," submitted to *IEEE Trans. Circuits and Syst. for Video Technol.*, Apr. 2022.

[3] A. Wieckowski *et al.*, "VVenC: An Open and Optimized VVC Encoder Implementation," in *Proc. IEEE ICMEW*, Shenzhen, China, July 2021.

[4] Fraunhofer HHI, "Fraunhofer Versatile Video Encoder (VVenC)," GitHub repository, July 1, 2022. https://github.com/fraunhoferhhi/vvenc.

[5] ITU-T H.266 and ISO/IEC 23090-3, "Versatile Video Coding," Apr. 2022 (and subsequent editions). https://www.itu.int/rec/T-REC-H.266.

[6] B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J.-R. Ohm, "Overview of the Versatile Video Coding (VVC) Standard and Its Applications," *IEEE Trans. Circuits and Syst. for Video Techn.*, vol. 31, no. 10, pp. 3736–3764, Oct. 2021.

[7] "Rate control destroys quality towards end of video," VVenC issue 82, GitHub, Aug. 2021. https://github.com/fraunhoferhhi/vvenc/issues/82.

[8] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B pictures and MCTF," in *Proc. IEEE ICME*, Toronto, Canada, Jul. 2006.

[9] D. Raveena Judie Dolly, G. Josemin Bala, and J. Dinesh Peter, "Adaptation of frames for GOP using NSEW affine translation for video compression," in *Proc. IEEE ICECS*, Coimbatore, India, Feb. 2014.

[10] C. R. Helmrich, M. Siekmann, S. Becker, S. Bosse, D. Marpe, and T. Wiegand, "XPSNR: A Low-Complexity Extension of the Perceptually Weighted Peak Signal-to-Noise Ratio for High-Resolution Video Quality Assessment," in *Proc. IEEE ICASSP,* Barcelona/online, May 2020.

[11] C. R. Helmrich, S. Bosse, H. Schwarz, D. Marpe, and T. Wiegand, "A Study of the Extended Perceptually Weighted Peak Signal-to-Noise Ratio (XPSNR) for Video Compression with Different Resolutions and Bit Depths," *ITU Journal: ICT Discoveries*, vol. 3, no. 1, May 2020. http://handle.itu.int/11.1002/pub/8153d78b-en.

[12] F. Bossen, X. Li, V. Seregin, K. Sharman, and K. Sühring, "VTM and HM common test conditions and software reference configurations for SDR 4:2:0 10-bit video," ITU/ISO/IEC doc. JVET-Y2010, Feb. 2022.

[13] JVET and Fraunhofer HHI, "VVCSoftware_VTM," Gitlab repository, May 2022. https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM.

[14] G. Ren, J. Jia, J. Wang, and Z. Chen, "AHG10: An improved rate control scheme," ITU doc. JVET-Y0105, Jan. 2022. www.jvet-experts.org.

[15] B. Bross, H. Kirchhoffer, C. Bartnik, M. Palkow, and D. Marpe, "AHG4 Multiformat Berlin test sequences," ITU doc. JVET-Q0791, Jan. 2020.

[16] A. Wieckowski, T. Hinz, C. R. Helmrich, B. Bross, and D. Marpe, "An optimized temporal filter implementation for practical applications," submitted to *IEEE Picture Coding Symposium*, San Jose, USA, 2022.

[17] A. Norkin and N. Birkbeck, "Film grain synthesis for AV1 video codec," in *Proc. IEEE Data Compress. Conf.*, Snowbird, USA, Mar. 2018.

[18] C. R. Helmrich, S. Bosse, P. Keydel, H. Schwarz, D. Marpe, and T. Wiegand, "A spectrally adaptive noise filling tool for perceptual transform coding of still images," *Proc. IEEE ICCE*, Berlin, Germany, Sep. 2018.

[19] R. Martin, "An Efficient Algorithm to Estimate the Instantaneous SNR of Speech Signals," in *Proc. EuroSpeech*, Berlin, Germany, Sep. 1993. www.isca-speech.org/archive_v0/eurospeech_1993/e93_1093.html.

[20] ITU-T HSTP-VID-WPOM and ISO/IEC TR 23002-8, "Working practices using objective metrics for evaluation of video coding efficiency experiments," 2021. https://www.itu.int/pub/T-TUT-ASC-2020-HSTP1.

[21] C. R. Helmrich, S. Bosse, M. Siekmann, H. Schwarz, D. Marpe, and T. Wiegand, "Perceptually optimized bit-allocation and associated distortion measure for block-based image or video coding, in *Proc. IEEE Data Compress. Conf.*, Snowbird, USA, pp. 172–181, Mar. 2019.

[22] F. Liu, Z. Liu, Y. Li, and Z. Chen, "AHG10: Extension of RC to support RA configuration with GOP size of 32," doc. JVET-T0062, Oct. 2020.