Efficient Multi-Rate Video Encoding for HEVC-Based Adaptive HTTP Streaming

Damien Schroeder, *Student Member, IEEE*, Adithyan Ilangovan, Martin Reisslein, *Fellow, IEEE*, and Eckehard Steinbach, *Fellow, IEEE*

***Abstract*— Adaptive HTTP streaming requires a video to be encoded at multiple representations, that is, different qualities. Encoding these multiple representations is a computationally complex process, especially when using the recent High Efficiency Video Coding (HEVC) standard. In this paper, we consider a multi-rate HEVC encoder and identify four types of encoding information that can be reused from a high-quality reference encoding to speed up lower quality-dependent encodings. We show that the encoding decisions from the reference cannot be directly reused, as this would harm the overall rate-distortion (RD) performance. Thus, we propose methods to use the encoding information to constrain the RD optimization of the dependent encodings so that the encoding complexity is reduced while the RD performance is kept high. We additionally show that the proposed methods can be combined, leading to an efficient multi-rate encoder that exhibits high RD performance and substantial complexity reduction. Results show that the encoding time for 12 representations at different spatial resolutions and signal qualities can be reduced on average by 38%, while the average bitrate increases by less than 1%.**

***Index Terms*— Adaptive HTTP streaming, HEVC, multi-rate encoding, video.**

I. Introduction

ADAPTIVE HTTP streaming is now widely used to deliver video over the Internet for both on-demand and live streaming [1]. The compressed video content is made available on an HTTP server at different bitrates (and thus different qualities) called representations. The video streaming clients request the representations based on a client-side adaptation mechanism, e.g., to match their current throughput.

In this paper, we focus on *multi-rate video encoding* for adaptive HTTP streaming, where a video is directly encoded at multiple bitrates, where each one is independently decodable [2]. Specifically, we consider the recent High Efficiency Video Coding (HEVC) standard [3], which offers a higher rate-distortion (RD) performance than its predecessor H.264/AVC at the cost of an increased encoding complexity. Encoding multiple representations of a video with HEVC is a computationally complex process, which limits the number of representations that can be encoded if computational resources or time are limited.

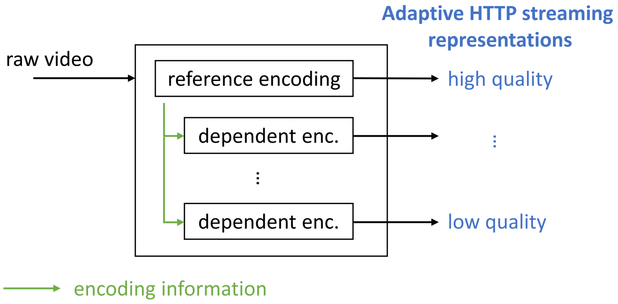


Fig. 1. Schema of a multi-rate encoder. Encoding information is passed from a reference encoding to reduce the complexity of dependent encodings.

In multi-rate encoding, the redundancy of encoding the same video at different bitrates is exploited in order to reduce the overall encoding complexity. Fig. 1 shows a schema of a multi-rate encoder. The representation with the highest quality is encoded as *reference*. Encoding information from the reference is passed to lower quality *dependent* encodings, where it is then used to decrease the encoding complexity. The main goal of our work is to determine information that can be reused and how this information can be reused to decrease the overall complexity, with the constraint that the RD performance is the least degraded.

This paper builds on our preliminary work in [4] and [5], where we proposed a multi-rate encoding method that reuses the coding unit (CU) structure information from the reference encoding in a single-resolution case and a multiple-resolutions case, respectively. As our main added contributions, we identify the following from a reference encoding that can also be reused for dependent encodings:

1) the prediction mode;
2) the intra mode;
3) the motion vectors (MVs).

For each information type, we observe similarities among representations at different qualities. We show that the encod-

Manuscript received January 18, 2016; revised April 28, 2016 and July 5, 2016; accepted August 3, 2016. Date of publication August 10, 2016; date of current version January 5, 2018. This work was supported in part by Technische Universität München and in part by a Friedrich Wilhelm Bessel Research Award from the Alexander von Humboldt Foundation. This paper was recommended by Associate Editor Y. Wang.

D. Schroeder and E. Steinbach are with the Chair of Media Technology, Technische Universität München, 80333 Munich, Germany (e-mail: damien.schroeder@tum.de; eckehard.steinbach@tum.de).

A. Ilangovan is with Nomor Research GmbH, 81541 München, Germany (e-mail: ilangovan@nomor.de).

M. Reisslein is with the School of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe, AZ 85287-5706 USA (e-mail: reisslein@asu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2016.2599028

ing decisions from the reference encoding cannot be directly reused in the dependent encodings as they would substantially decrease the RD performance. Thus, we propose methods to reuse the information such that the RD optimization (RDO) in the dependent encodings is constrained, both in a single-resolution case across the SNR dimension and in a multiple-resolutions case across different spatial resolutions. We evaluate the impact of the different reuse methods both on the RD performance and on the encoding time. In a last step, we combine the different methods to leverage all the possible encoding time reductions and evaluate the outcome of the proposed multi-rate encoder both for a representations set with fixed quantization parameter (QP) encoding and for a set using rate-control-based encoding.

This paper is organized as follows. The related work is presented in Section II. Common settings for this paper are introduced in Section III. In Section IV, we evaluate the potential of the different information reuse methods. Sections V–VII present the information reuse methods both in the single-resolution case and in the multiple-resolutions case for the prediction mode, the intra mode, and the MVs, respectively. We summarize the CU structure reuse methods from [4] and [5] in Section VIII. Finally, we propose the combined multi-rate encoder in Section IX, and Section X concludes this paper.

II. Related Work

In order to avoid buffer underflow in video streaming, the bitrate of the compressed video stream has to be adapted to the communication channel. The bitrate of a compressed video is influenced by the spatial resolution, the temporal resolution (i.e., the frame rate), and the signal fidelity (i.e., level of distortion introduced by lossy compression). A video can be compressed at a target bitrate using rate control. Different rate control methods have recently been proposed for HEVC (see [6] and [7]). Another possibility is to transcode (or transrate) an already encoded video to another bitrate [8]–[10]. The drawback of transcoding is that the RD performance is decreased due to requantization. Both rate control and transcoding are designed to target one specific representation.

On the other hand, a video can be encoded to provide inherent scalability. Scalable video coding [11] encodes a video into a base layer and several enhancement layers. Decoding an enhancement layer requires the availability of the base layer at the decoder. The major drawbacks of scalable video coding are the increased decoding complexity compared with a single-layer encoding with no decoding dependencies and the decreased RD performance. For example, the scalable extension of HEVC increases the bitrate by at least 14.3% [12] compared with a single-layer HEVC. In [13], Xu *et al.* propose a method to reduce the encoding complexity by performing the motion estimation and mode decision only for the base layer, but by considering the highest quality enhancement layer for temporal prediction, they show that this can improve the coding efficiency due to less motion signaling. In an HTTP streaming scenario with scalable coding, a client needs to send a request per layer, which leads to inefficient multiple requests to obtain a high-quality representation. Due to the different drawbacks, scalable coding is not expected to be widely deployed for adaptive HTTP streaming, contrary to the single-layer HEVC.

Finally, a video can be encoded simultaneously at different qualities and thus bitrates to form a set of independently decodable representations. This multi-rate encoding is especially well suited for adaptive HTTP streaming. In this context, Finstad *et al.* [2] proposed a multi-rate VP8 encoder that directly reuses RDO decisions from a reference encoding. This direct reuse leads to a severe degradation of the RD performance of the dependent encodings. More recently, De Praeter *et al.* [14] proposed a multi-rate system for HEVC. The CU structure of the dependent encodings is predicted based on features from the reference encoding using machine learning trained on the first frames of the video. The proposed system achieves an encoding time reduction of around 67% at the cost of a relatively high bitrate increase of around 5%.

Typically, adaptive HTTP streaming deployments offer 10 to 15 representations (see [15], [16]), which are chosen to accommodate different devices and different connection types, and thus have to span across different spatial resolutions and over a wide range of bitrates. Toni *et al.* [17] have proposed a method to define a set of representations that maximizes the user satisfaction by taking into account network characteristics and video content. However, [17] is based on H.264/AVC, and there have been no studies on optimal sets of representations for HEVC so far.

III. Common Settings

In this section, we present the settings that are used throughout this paper.

A. HEVC Encoder

We use the reference HEVC encoder HM 16.5 [18] compiled with *gcc* 4.8.4 as software encoder throughout this paper. The unmodified encoder is used to gather observations and as a baseline for comparison with our method. Our proposed methods are implemented based on HM 16.5 in order to allow for fair comparisons. Furthermore, we follow the common test conditions and software reference configurations from [19]. For adaptive HTTP streaming, the video representations must be segmented in the time domain into individually decodable segments, that is, the segments need to start with an I-frame. As we are focusing on adaptive HTTP streaming, we use the *random access, main* profile defined in [19], which provides periodic I-frames in the encoding structure.

B. Video Sequences

We use a set of ten different sequences with an original spatial resolution of 1920 × 1080 pixels (1080p). Two sequences *Kimono* (24 fps) and *ParkScene* (24 frames/s) are from [19] and the eight other sequences *BlueSky* (25 frames/s), *CrowdRun* (50 frames/s), *DucksTakeOff* (50 frames/s), *ParkJoy* (50 frames/s), *PedestrianArea* (25 frames/s), *Riverbed* (25 frames/s), *RushHour* (25 frames/s), and *Sunflower* (25 frames/s) are from [20]. For

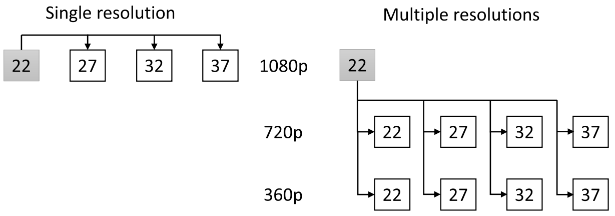


Fig. 2. Reference encoding (gray) and dependent encodings (white) with QPs used for the single-resolution and the multiple-resolutions case.

the multi-rate system with multiple spatial resolutions, the original 1080p uncompressed sequence is downsampled to 1280 × 720 (720p) and 640 × 360 pixels (360p).

C. Metrics

The RD performance and the encoding time are the two metrics used to compare our proposed methods with the unmodified HM encoder. The RD performance difference is measured using the Bjøntegaard delta rate (BD-rate), which expresses the average bitrate difference in percent over the considered Peak Signal-to-Noise Ratio (PSNR) interval, and the Bjøntegaard delta PSNR (BD-PSNR), which expresses the average PSNR difference in decibels over the considered bitrate interval [21], [22]. In order to calculate BD-rate and BD-PSNR, four RD points are needed. Thus, for each spatial resolution, we encode a sequence at four different qualities (fixed QPs 22, 27, 32, and 37) [19]. The encoding time difference (ΔT) is then measured for each resolution as the difference of the total encoding time for the four representations. Fig. 2 shows the reference encoding and the dependent encodings for both the single-resolution case and the multiple-resolutions case. As the reference is always at 1080p, the encoding time for the reference encoding is included in the encoding time difference for the single-resolution case (1080p) but not in the multiple-resolutions case (720p and 360p).

IV. Information Reuse

The main goal of the multi-rate video encoder is to reduce the overall encoding time for multiple representations. The RDO accounts for a major part of the encoding time in HEVC [23]. Thus, we focus on simplifying the RDO for dependent encodings in this entire study. We do this using the information of the reference encoding to constrain the set of RDO options to be tested.

A. Background on RDO

The RDO minimizes the RD cost $J = D + \lambda R$, where D is the distortion, R is the bitrate, and λ is a Lagrange multiplier that determines the tradeoff between distortion and rate. Each coding decision, for example, a certain block size and a prediction mode, is associated with a distortion and a bitrate. Thus, the RDO is equivalent to finding the encoding parameters that minimize the RD cost J.

To understand the RDO, it is necessary to know the set of possible encoding options. A major novelty of HEVC compared with its predecessor H.264/AVC is the new block partitioning structure based on a quadtree [24]. The coding tree unit (CTU) is the basic block and is subdivided into CUs, which correspond to the nodes of the quadtree. Each CU can be encoded with one of two prediction modes: either intra encoding (spatial prediction) or inter encoded (temporal prediction). A CU contains one, two, or four prediction units (PUs), and the prediction options are determined at PU level. In the case of intra encoding, the intra-prediction mode can be one out of 35 intra-prediction modes in HEVC [25]. There are 33 angular prediction modes plus a planar prediction mode and a DC prediction mode. In the case of inter encoding, a block is predicted from another frame using motion estimation with a quarter-sample accuracy. The MVs describe the displacement between the blocks to be predicted and the prediction blocks.

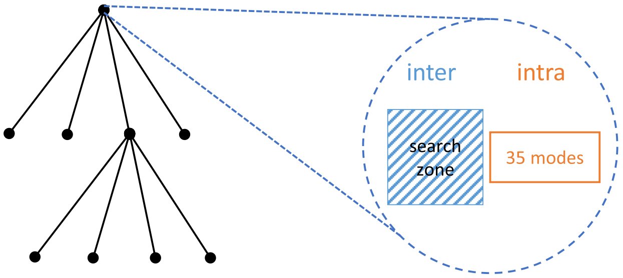


Fig. 3. Schema of the RDO in HEVC: traversal of the CTU quadtree to analyze each CU.

B. RDO Complexity

Fig. 3 schematically represents the different parts of the RDO in an HEVC encoder. Due to the quadtree structure of the CTUs, the RDO consists of a tree traversal. In the most general case, each node, that is, each possible CU has to be analyzed, that is, checked for intra or inter prediction, which means that the CUs are split until the smallest possible CU size is reached. For inter prediction, the complexity of the motion estimation is mainly due to the estimation algorithm and the size of the search zone. For intra prediction, the complexity is due to the number of intra modes to be checked. After the quadtree has been traversed, the combination of CU structure and prediction mode that leads to the lowest RD cost is chosen by the encoder for further processing (such as entropy encoding).

In [4] and [5], we showed that reusing the CU structure information from a reference encoding can substantially speed up the encoding of dependent representations by constraining the quadtree that is traversed during the RDO. In this paper, we identify that additionally to the block structure, the prediction mode, the intra mode, and the MVs can potentially be reused in order to further speed up the RDO of dependent encodings.

C. Preliminary Study

The potential encoding time reductions differ depending on the information that is reused in the multi-rate system. Indeed, reusing the CU structure information for a dependent encoding is equivalent to skipping the analysis of certain nodes in the quadtree. Thus, the prediction mode decision and the

TABLE I

AVERAGE ANALYSIS TIME FOR DIFFERENT BLOCK SIZES (ms)

depth	CU	intra prediction	inter prediction	intra mode	MV
0	17.35	3.03	10.75	1.85	2.25
1	5.76	0.90	3.70	0.49	0.61
2	1.94	0.28	1.23	0.13	0.20
3	0.54	0.36	0.29	0.05	0.06

underlying intra direction or MV search are skipped as well. For example, this means that we expect the CU structure reuse to lead to larger time gains than the prediction mode decision reuse.

We illustrate this by performing time measurements of the individual RDO steps. While complexity assessment is a topic in its own right (see [23]), a time measurement is a good measure of the underlying complexity of a software encoder. Although the exact value of the time measurements is not relevant because the encoding time depends on the computer configuration, the relative times give insight into the relative complexities.

Table I shows the average analysis time (ms) of different steps of the RDO at different CU depths. The average is taken over 12 240 CTUs from different sequences. As expected, the time to analyze an entire CU (first column) is the largest, as it includes analysis of intra and inter prediction. The results also indicate that the inter-prediction part of the RDO takes longer than the intra-prediction part, mainly due to the various possible PU structures of inter prediction, such as asymmetric partitioning. The sum of the intra and inter time does not have to be smaller than the CU time (e.g., at depth 3), because the HM encoder implements early decision algorithms and does not always analyze all the possibilities. Finally, the times to determine the intra mode for intra prediction and the MVs for inter prediction are the shortest times.

V. PREDICTION MODE REUSE

A. Observations

In order to identify the similarities in the prediction mode across different qualities, we encode ten videos at different QPs ranging from 22 to 40. We gather the intra/inter decision at every node of the quadtree during the RDO, that is, for every possible CU depth. We first examine in Fig. 4 the percentage of inter-predicted blocks in inter-predicted frames (i.e., I-frames are omitted). We observe that, on average, the percentage of inter blocks increases with increasing QP, independently of the block's depth.

We next investigate if the decision for a node in a low-quality encoding (QP from 24 to 40) is the same as the decision in the reference encoding (QP 22). Fig. 5 shows what percentage of intra-encoded blocks in the reference encoding (QP 22) is still intra encoded in the lower quality encodings, as a function of the QP. At QP 22, we have 100% of intra blocks in common, as expected, as we compare an encoding with itself. As the QP increases to 24, only 80% of the blocks intra encoded at QP 22 are still intra encoded. This means that if we were to directly reuse the intra-encoding information from the reference at QP 22, we would make a suboptimal decision for 20% of the blocks. The percentage of intra blocks in common decreases further as the QP increases. We conclude from these results that we cannot reuse the intra-encoding information from a high-quality reference encoding and skip the inter-analysis part, as this would lead to a large number of suboptimal decisions and thus to a decreased RD performance.

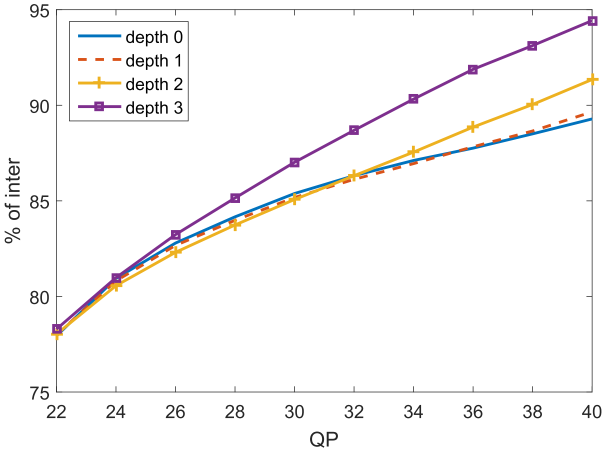


Fig. 4. Percentage of inter blocks in inter-predicted frames as a function of the QP.

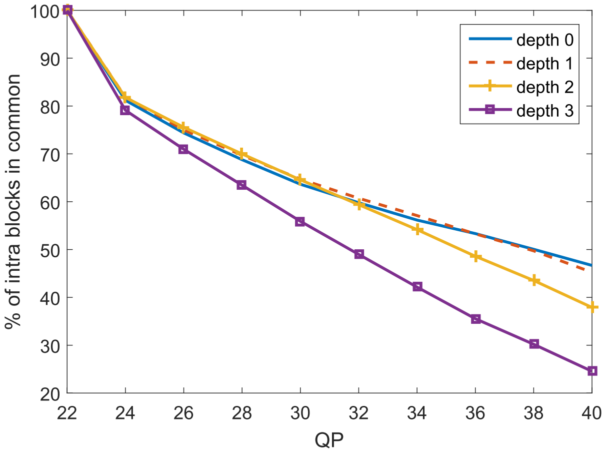


Fig. 5. Percentage of intra blocks in common with the reference at QP 22 at different depths.

Fig. 6 shows the percentage of inter-encoded blocks in the reference encoding (QP 22) that is still inter encoded in the lower quality encodings, as a function of the QP. Unlike the intra case, a very high percentage of inter blocks in common can be observed across the range of QPs, with a minimum around 97.6% at QP 24 and depth 3. These results indicate that a block that is encoded in inter mode in the reference encoding will be inter encoded in a lower quality representation with a very high probability.

We also study if a low-quality reference (QP 40) could alternatively be used to speed up higher quality-dependent encodings (QP 22 to 38). Figs. 7 and 8 show that in that case, the percentage of inter and intra blocks in common with the reference can be as low as 83% and 80%, respectively.

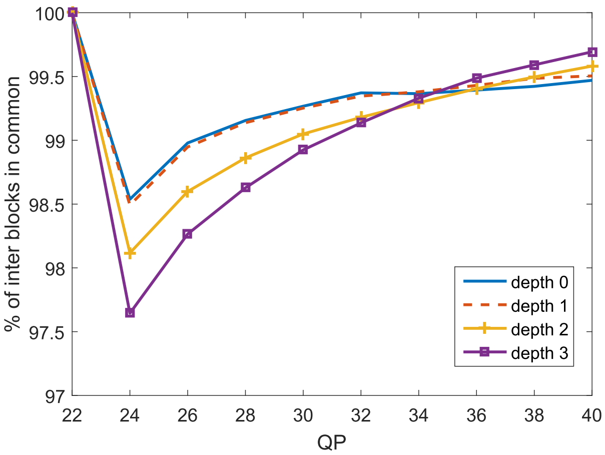


Fig. 6. Percentage of inter blocks in common with the reference at QP 22 at different depths.

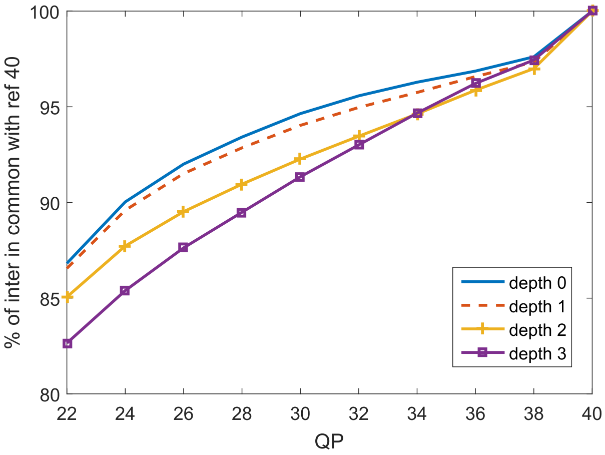


Fig. 7. Percentage of inter blocks in common with the reference at QP 40 at different depths.

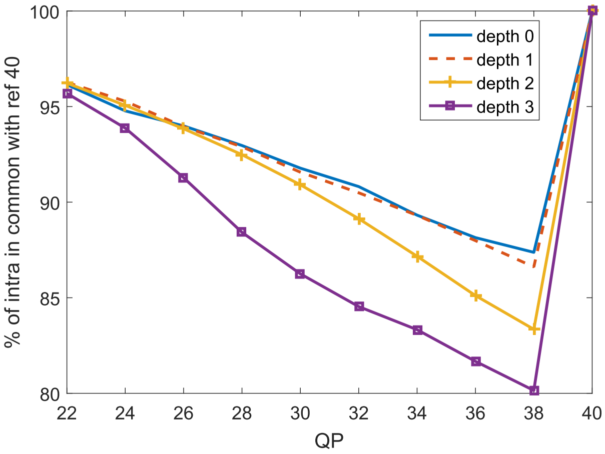


Fig. 8. Percentage of intra blocks in common with the reference at QP 40 at different depths.

TABLE II

COMPARISON OF ENCODING WITH PREDICTION MODE REUSE VERSUS CONVENTIONAL ENCODING

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.12%	−0.005 dB	−0.58%
CrowdRun	0.13%	−0.005 dB	−3.04%
DucksTakeOff	0.02%	−0.0001 dB	−1.96%
Kimono	0.09%	−0.003 dB	−2.17%
ParkJoy	0.07%	−0.003 dB	−2.92%
ParkScene	0.08%	−0.003 dB	−1.21%
PedestrianArea	0.64%	−0.020 dB	−1.28%
RiverBed	0.07%	−0.003 dB	−0.49%
RushHour	0.30%	−0.007 dB	−1.56%
Sunflower	0.02%	−0.002 dB	−0.78%
Average	**0.15%**	**−0.005 dB**	**−1.60%**

This indicates that a substantial number of suboptimal decisions would be made if a low-quality reference was to be used as a reference. This means that a low-quality reference is a worse choice than a high-quality reference in terms of the overall RD performance. Thus, in the following, we concentrate on a high-quality reference.

B. Information Reuse

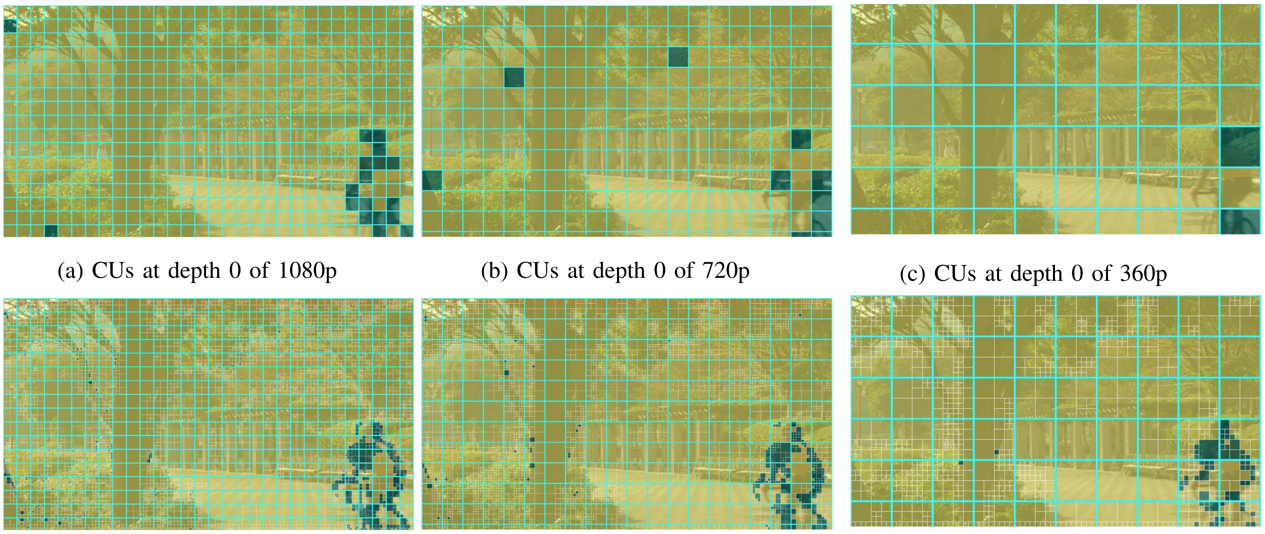
Given the preceding observations, we propose to reuse information about the prediction mode from a high-quality reference encoding in order to speed up the RDO of lower quality-dependent encodings. Specifically, we gather the prediction mode decision at every node of the quadtree during the RDO of the reference encoding. That is, we store one decision at depth 0, four decisions at depth 1, 16 decisions at depth 2, and 64 decisions at depth 3. If the decision from the reference encoding is inter mode, we do not check intra prediction in the dependent encodings, because the decision for the node will be inter with a very high probability. This information reuse scheme leads to a suboptimal decision with a small probability. The few blocks with the suboptimal decision will contribute to a small decrease in the RD performance. However, we cannot skip the inter-analysis part if we have an intra-encoded CU in the reference encoding, because this would lead to numerous suboptimal decisions and, thus, substantially harm the overall RD performance.

C. Preliminary Results

We implement the proposed method to assess the impact of reusing only the prediction mode decision on a multi-rate system. We compare our HM-based implementation with the original HM encoder, and the results are listed in Table II. On average, our proposed method shows a BD-rate increase of approximately only 0.15%, while the overall encoding time over four representations is reduced by 1.60%. The average time gain is relatively small, which is due to the fact that we do not skip the inter-analysis part, which would have resulted in higher time gains (see Table I).

D. Multiple Resolutions

We next apply the idea of reusing the prediction mode decision to the case where multiple spatial resolutions of the video have to be encoded. Again, we first analyze the similarities in the prediction modes at different depths. As an example, Fig. 9 shows the prediction mode decision of the 55th frame of the *ParksScene* sequence at different resolutions,



(a) CUs at depth 0 of 1080p (b) CUs at depth 0 of 720p (c) CUs at depth 0 of 360p

(d) CUs at depth 3 of 1080p (e) CUs at depth 3 of 720p (f) CUs at depth 3 of 360p

Fig. 9. Inter- (green) and intra- (yellow) mode decision for the 55th frame of the *ParkScene* sequence encoded at QP 22.

TABLE III

PREDICTION MODE DECISION FOR MULTIPLE RESOLUTIONS ACCORDING TO INTER-PREDICTION PERCENTAGE p

Condition	Prediction mode
$p \geq \theta$	*inter*
$p < 100 - \theta$	*intra*
else	*no reuse*

both for depth 0 CUs and depth 3 CUs. Similarities can be seen across resolutions, and were also observed at depths 1 and 2. A quantitative analysis similar to the one for different QPs has indicated that both intra and inter decisions can be reused across resolutions.

However, as explained in [5], the challenge of reusing information from a high-resolution reference encoding comes from the fact that there is no direct correspondence (i.e., overlap) between blocks at different resolutions if the downsampling factor is not a multiple of 2.

Therefore, we propose an algorithm to determine the prediction mode at the CU level at each depth for the dependent low-resolution encodings: For a CU of the low-resolution video at depth i, we select the corresponding area A of the reference at the same depth i. We measure the percentage p of A which is encoded with inter prediction. We then determine the prediction mode of the current CU, depending on the value of p according to Table III. As a parameter, the threshold θ can take a value between 50 and 100.

If a CU is mapped to inter mode, the intra-analysis part is skipped during RDO. Similarly, if a CU is mapped to intra, the inter-analysis part is skipped during RDO. Finally, for the *no reuse* case, we do not skip any analysis part.

As we are interested in keeping the multi-rate system RD performance close to the original HEVC encoder, after preliminary evaluations for different θ values, we select a value of $\theta = 80$ for the following. As an example, Table IV shows the effect of a varying threshold on the encoding performance of the *Kimono* sequence at 720p.

TABLE IV

ENCODING PERFORMANCE FOR THE *KIMONO* SEQUENCE AT 720p FOR DIFFERENT THRESHOLD VALUES θ

θ	**60**	**70**	**80**	**90**	**100**
BD-rate	1.01%	0.91%	0.74%	0.70%	0.69%
ΔT	−8.97%	−8.68%	−8.48%	−7.89%	−7.84%

TABLE V

COMPARISON OF ENCODING RESULTS FOR 720p BASED ON A 1080p REFERENCE. THE PREDICTION MODE IS MAPPED USING A THRESHOLD OF $\theta = 80$

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.13%	−0.01 dB	−2.68%
CrowdRun	0.56%	−0.03 dB	−7.82%
DucksTakeOff	1.77%	−0.06 dB	−19.65%
Kimono	0.74%	−0.03 dB	−8.48%
ParkJoy	0.47%	−0.02 dB	−9.29%
ParkScene	0.24%	−0.01 dB	−3.22%
PedestrianArea	0.61%	−0.03 dB	−13.75%
Riverbed	0.22%	−0.01 dB	−52.11%
RushHour	0.49%	−0.02 dB	−7.72%
Sunflower	0.53%	−0.02 dB	−2.85%
Average	**0.58%**	**−0.02 dB**	**−12.76%**

Comparison results for 720p and 360p using a 1080p reference are shown in Tables V and VI. The average BD-rate increase is kept low at 0.58% and 0.84% for 720p and 360p, respectively. The average time reduction is higher than in the single-resolution case, which is mainly due to the fact that the intra-encoding information is reused and thus the inter-analysis parts can be skipped in the multi-resolution case.

VI. INTRA MODE REUSE

A. Observations

An intra PU is characterized by its intra-prediction mode, which can be planar prediction (mode 0), DC prediction (mode 1), or one of 33 angular predictions (modes 2 to 34), which sums up to 35 different possible intra-prediction modes

TABLE VI

COMPARISON OF ENCODING RESULTS FOR 360p BASED ON A 1080p REFERENCE. THE PREDICTION MODE IS MAPPED USING A THRESHOLD OF $\theta = 80$

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.06%	0.00 dB	−3.14%
CrowdRun	0.56%	−0.03 dB	−5.41%
DucksTakeOff	3.45%	−0.17 dB	−18.72%
Kimono	0.87%	−0.04 dB	−7.20%
ParkJoy	0.54%	−0.03 dB	−7.61%
ParkScene	0.23%	−0.01 dB	−2.67%
PedestrianArea	0.87%	−0.05 dB	−11.52%
Riverbed	0.97%	−0.04 dB	−51.66%
RushHour	0.37%	−0.02 dB	−5.90%
Sunflower	0.50%	−0.03 dB	−2.63%
Average	**0.84%**	**−0.04 dB**	**−11.65%**

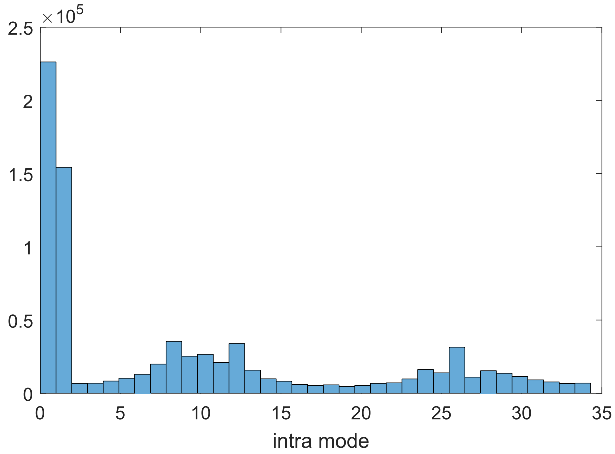


Fig. 10. Histogram of the luma intra mode at depth 2 for ten videos at QP 22.

in HEVC [25]. An intra CU at depth between 0 and 2 always contains only one PU, whereas a CU at depth 3 can contain one PU or four square PUs [24]. From a PU perspective, this last partitioning is equivalent to a depth 4. We encode ten frames of ten videos with the *intra, main* profile [19] (that is, only I-frames) with QP ranging from 22 to 40, in order to assess the similarities in intra mode from the luma component across videos with different qualities. As an example, the distribution of the intra modes at depth 2 for the QP 22 videos is shown as a histogram in Fig. 10. The easiest way to reuse the intra mode information in a multi-rate system would be to directly reuse the intra mode of a PU from the reference for the same PU in a low-quality encoding.

Fig. 11 shows the intra mode of PUs at QP 30 and depth 2, which were intra mode 10 (horizontal prediction) in the reference encoding at QP 22. Intra mode 10 is still the intra mode with the most elements; however, it accounts for only 32% of all PUs, which means that directly reusing the intra mode 10 from the reference at QP 22 for a low-quality encoding at QP 30 would lead to 68% of suboptimal decisions at depth 2 for these PUs. Fig. 12 shows what percentage of PUs that are intra mode 10 at QP 22 are still intra mode 10 at other QPs and different depths. The values range between 10% and 50%. We have observed similar trends with other intra modes. We infer from these observations that the intra mode cannot be reused directly. However, the intra mode from the reference can still be considered a "good candidate" with a probability between 10% and 50%.

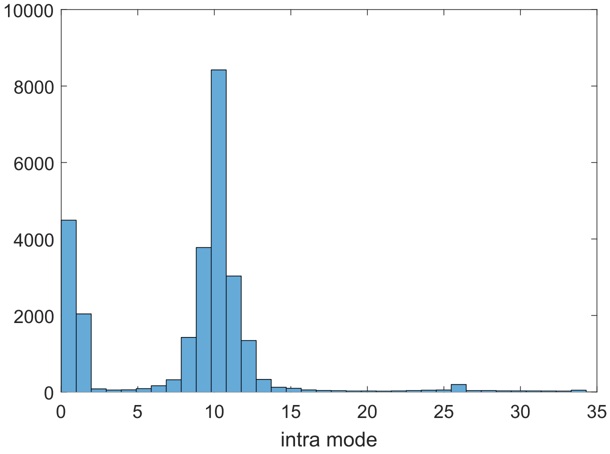


Fig. 11. Histogram of the luma intra mode at depth 2 for ten videos at QP 30 for PUs, which were intra mode 10 at QP 22.

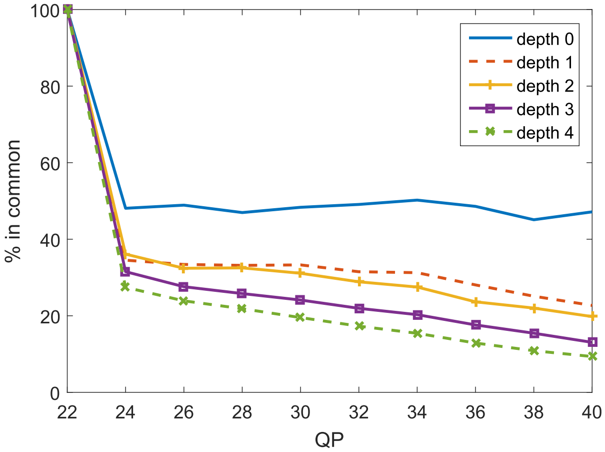


Fig. 12. Percentage of PUs with intra mode 10 at QP 22, which are still intra mode 10 at lower quality QPs at different depths.

B. Information Reuse

Calculating the full RD costs for all the 35 intra modes is too complex to be practical. Thus, HM implements a suboptimal fast intra algorithm that first evaluates an approximated cost for all the 35 modes and then makes a candidate list ψ with the best three or eight candidates (depending on the PU size), which are in turn fully analyzed [25]. Based on the observation that the intra mode from the reference is a "good candidate," we propose to reduce the candidate list to three for all the PU sizes and then check if the reference intra mode is in this list. If it is not, then the reference intra mode is added to this short list, which then contains four candidates to be fully analyzed. The choice of not to reduce the list down to less than three candidates comes from the fact that the approximated cost is sensitive to the three *most probable intra modes* defined in HEVC [25].

TABLE VII

COMPARISON OF ENCODING WITH INTRA MODE REUSE VERSUS CONVENTIONAL ENCODING

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.08%	−0.005 dB	−14.16%
CrowdRun	0.28%	−0.02 dB	−13.60%
DucksTakeOff	0.05%	−0.0001 dB	−14.18%
Kimono	−0.10%	0.004 dB	−14.33%
ParkJoy	0.17%	−0.01 dB	−13.97%
ParkScene	0.12%	−0.005 dB	−13.41%
PedestrianArea	−0.02%	0.001 dB	−13.53%
Riverbed	−0.06%	0.002 dB	−13.67%
RushHour	−0.15%	0.004 dB	−13.49%
Sunflower	−0.05%	0.003 dB	−13.49%
Average	**0.03%**	**−0.003 dB**	**−13.78%**

C. Preliminary Results

We implement the proposed method to assess the impact of reusing only the intra mode information on a multi-rate system. The first results with the *random access, main* profile lead to an average encoding time reduction of 0.88% and to a BD-rate increase of 0.004%. The low encoding time reduction comes from the high number of inter-encoded frames, where the intra mode reuse does not have a big impact. We now encode the videos with the *intra, main* profile in order to focus on I-frames only, where the intra mode reuse method will have the highest impact, and the results are presented in Table VII. The average time gain of almost 14% comes at the expense of a very small BD-rate increase of 0.03%. We further observe that the RD performance is actually improved compared with the original HM encoder for the *Kimono*, *PedestrianArea*, *Riverbed*, *RushHour*, and *Sunflower* sequences. This confirms that the intra mode information from a high quality can be considered a "good candidate." Although our proposed method is specific to the HM encoder, we believe that similar reuse schemes can be explored for various HEVC encoders.

D. Multiple Resolutions

As stated in Section V-D, the challenge in the case of multiple resolutions is that there is no direct correspondence between blocks from different resolutions. Thus, for a PU at a low resolution, there is not necessarily a single corresponding PU at a reference high resolution, and thus there may be multiple different intra modes in the area of the low-resolution PU. As there is no indication about which one would make the most sense and because adding all these possible intra modes to the list of candidates to be fully checked would increase the complexity, we need to focus on a different method.

We propose to merge the candidate lists ψ_k from the high-resolution PUs k, which overlap the area of the considered low-resolution PU into a multiset $\psi_{\text{merge}} = \uplus_k \psi_k$. To obtain the final candidate list ψ for the low-resolution PU, we pick the three elements with the highest multiplicity (that is, the elements that occur most often in the multiset). Ties are resolved randomly. We give a numerical example shown in Fig. 13. The area of the reference encoding corresponding to the current PU overlaps four PUs with candidate lists $\psi_1 = \{0, 1, 21\}, \psi_2 = \{0, 11, 25\}, \psi_3 = \{0, 1, 26\}, \psi_4 = \{1, 19, 21\}$. The multiset is then $\psi_{\text{merge}} = \{0, 0, 0, 1, 1, 1, 11, 19, 21, 21, 25, 26\}$, and the final candidate list $\psi = \{0, 1, 21\}$.

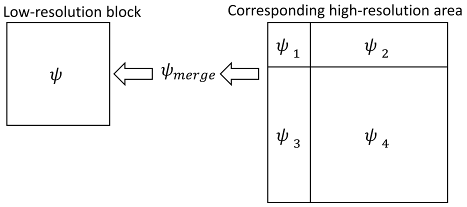


Fig. 13. Example of merging and clipping of candidate lists from a high-resolution reference.

TABLE VIII

PERCENTAGE OF CANDIDATE LISTS CONTAINING THE OPTIMAL INTRA MODE, AS GIVEN BY THE ORIGINAL ENCODER

Resolution	PUs at depth				
	0	1	2	3	4
720p	93.7%	79.7%	61.0%	64.3%	57.8%
360p	96.9%	82.4%	64.4%	64.2%	58.7%

TABLE IX

COMPARISON OF ENCODING RESULTS FOR 720p BASED ON A 1080p REFERENCE. THE INTRA MODE CANDIDATE LIST IS REUSED UNTIL DEPTH 1

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.19%	−0.01 dB	−7.95%
CrowdRun	0.07%	−0.01 dB	−7.32%
DucksTakeOff	0.41%	−0.02 dB	−6.34%
Kimono	0.65%	−0.03 dB	−8.97%
ParkJoy	0.09%	−0.01 dB	−7.56%
ParkScene	0.25%	−0.01 dB	−7.79%
PedestrianArea	1.50%	−0.07 dB	−9.68%
Riverbed	0.41%	−0.02 dB	−8.40%
RushHour	1.97%	−0.09 dB	−9.87%
Sunflower	1.25%	−0.07 dB	−9.86%
Average	**0.68%**	**−0.03 dB**	**−8.37%**

We assess how accurate the merging and clipping is by checking if the best intra mode found in the original encoding is in the derived candidate list. Table VIII shows the percentage of derived candidate lists containing the best intra mode. This percentage decreases with increasing depth at both 720p and 360p. We propose to reuse the derived candidate list information until depth 1, in order to avoid making too many suboptimal decisions. In the case of the *random access, main* profile, the average encoding time reduction is 0.77% and 0.76%, while the BD-rate increase is 0.58% and 0.30% for 720p and 360p, respectively. Preliminary results for the *intra, main* profile are shown in Tables IX and X. The average encoding time decrease is 8.37% and 7.54% for 720p and 360p, respectively, which is lower than the results achieved in the single-resolution case. This can be explained on the one hand by the fact that here the time gain comes from skipping the evaluation of the approximated costs, while in the single-resolution case, the full RD cost calculation is shortened. On the other hand, only the approximated costs at depths 0 and 1 are skipped here, whereas all the depths are affected in the single-resolution case.

TABLE X

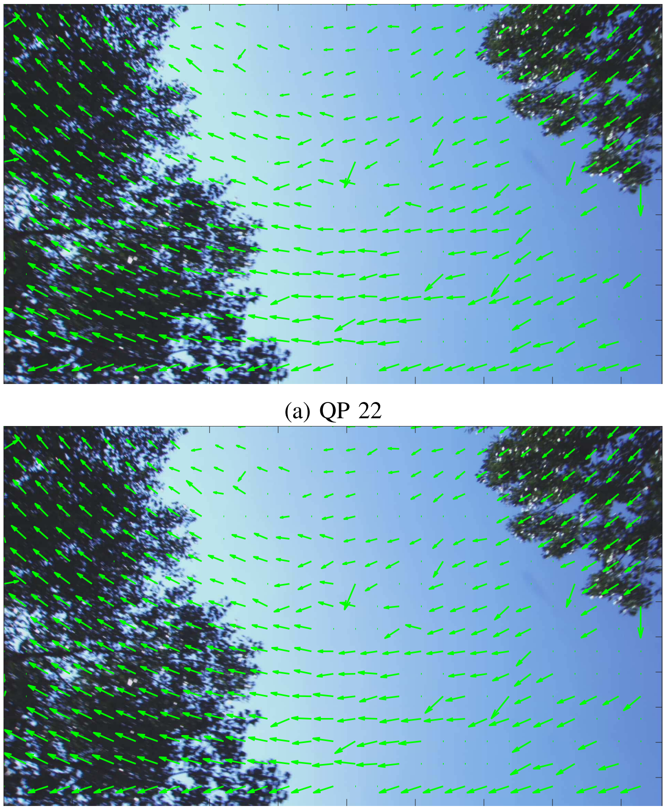
COMPARISON OF ENCODING RESULTS FOR 360p BASED ON A 1080p REFERENCE. THE INTRA MODE CANDIDATE LIST IS REUSED UNTIL DEPTH 1

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.07%	−0.01 dB	−8.88%
CrowdRun	0.06%	0.00 dB	−5.88%
DucksTakeOff	0.19%	−0.01 dB	−5.94%
Kimono	0.30%	−0.02 dB	−8.74%
ParkJoy	0.16%	−0.01 dB	−7.24%
ParkScene	0.23%	−0.01 dB	−7.65%
PedestrianArea	0.92%	−0.06 dB	−7.34%
RiverBed	0.18%	−0.01 dB	−7.35%
RushHour	0.69%	−0.05 dB	−8.42%
Sunflower	0.50%	−0.04 dB	−8.02%
Average	**0.33%**	**−0.02 dB**	**−7.54%**

VII. MOTION VECTOR REUSE

A. Observations

Inter-predicted frames rely on a motion-compensated prediction based on previously encoded frames. An inter CU contains either one PU (called 2N×2N), two, or four PUs [24]. Each PU is characterized by a two-dimensional MV that points to the predictor block in a specified reference frame. The *random access, main* profile has an encoding structure with B-frames, that is, frames can be predicted from two reference frames. The reference frames are listed in two lists, L0 and L1. We examine the MVs of a video at different qualities. Therefore, we compare the MVs of the 2N × 2N PU at each CU depth found during the inter-analysis process for ten videos encoded with a QP ranging from 22 to 40. As an example, Fig. 14(a) and (b) shows the MVs at depth 0 and list L0 from the second frame of the *BlueSky* sequence encoded at QP 22 and 24, respectively. Blocks with no displayed MVs are intra predicted at depth 0. The MVs are scaled uniformly for better visualization. Strong similarities can be observed, and we have observed comparable similarities at different qualities and for other videos. However, there is always a small MV difference. Thus, we cannot directly reuse the MVs from the reference encoding. To quantify the similarity, we determine if the difference vector of an MV with the corresponding MV in the reference at QP 22 has a norm smaller than 4 pixels. Fig. 15 shows the percentage of PUs that have an MV difference with the corresponding reference MV smaller than 4 pixels in the case of the L0 list. This percentage is around 95% at depth 0, whereas it can go down to 85% at depths 2 or 3. Results for the L1 list are very similar.


(a) QP 22

(b) QP 24

Fig. 14. MVs at depth 0 and list L0 for the second frame of *BlueSky*.

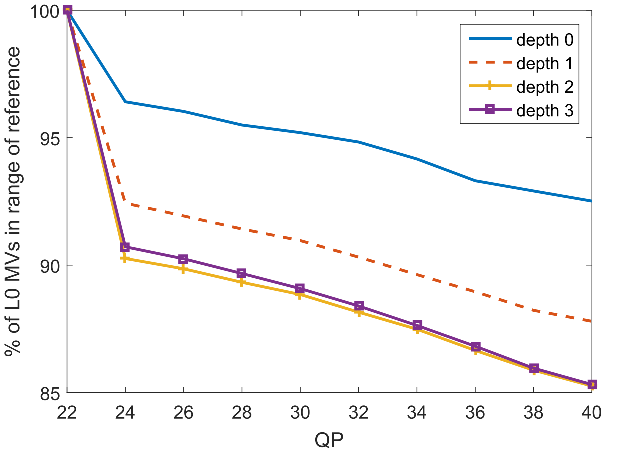


Fig. 15. Percentage of PUs that have an MV difference with the reference MV smaller than 4 pixels for the L0 list.

B. Information Reuse

Based on the insight that the MVs at lower quality encodings are very similar to the MVs from the high-quality reference encoding, we propose to restrict the motion estimation to the vicinity of the reference MV in the dependent encodings. For this, we first collect the MV of the 2N × 2N PU in the reference encoding for each possible CU (i.e., at each node in the quadtree) for both the L0 and L1 lists. We do not collect the index of the reference frame in the lists. The HM encoder implements a test zone (TZ) search algorithm [26], [27] (combination of diamond search and raster search) with a default search range of 64 pixels in the *random access, main* profile and up to 2 reference frames in both lists L0 and L1. The TZ search algorithm is initialized with either the zero vector or with a vector predicted from neighboring blocks. In our proposed method, we initialize the TZ search algorithm with the MV from the reference encoding for the corresponding list. We restrict the search range to 4 pixels, and we deactivate the raster search part of the TZ search. As we do not have the reference frame information, the motion estimation is still run for all the possible reference frames

TABLE XI
COMPARISON OF ENCODING WITH MV REUSE VERSUS CONVENTIONAL ENCODING

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.07%	−0.003 dB	−4.17%
CrowdRun	−0.08%	0.003 dB	−4.89%
DucksTakeOff	−0.09%	0.002 dB	−5.60%
Kimono	−0.06%	0.002 dB	−6.25%
ParkJoy	−0.20%	0.008 dB	−5.71%
ParkScene	−0.05%	0.002 dB	−2.73%
PedestrianArea	−0.19%	0.006 dB	−8.75%
Riverbed	−0.05%	0.002 dB	−11.58%
RushHour	−0.09%	0.002 dB	−6.16%
Sunflower	−0.52%	0.016 dB	−6.25%
Average	**−0.12%**	**0.004 dB**	**−6.21%**

in each list. The dependent encoder is not restricted to the 2N × 2N PU partitioning and uses the reference MV from the 2N × 2N PU to initialize the motion estimation of the different possible PUs.

C. Preliminary Results

Table XI shows the comparison results of our implementation of the proposed MV information reuse method with the original HM encoder. On average, our method can reduce the encoding time over four representations by 6.21%. Interestingly, the proposed reuse method improves the RD performance, as the average BD-rate is decreased by 0.12% and the average BD-PSNR is increased by 0.004 dB. On the one hand, this is due to the fact that the original HM encoder does not perform a full-search as motion estimation, and thus, does not always find the optimal MV in the RD sense. On the other hand, it shows that initializing the motion estimation with a good guess is beneficial in terms of RD performance, even if only one reference MV per CU is used, although there can be multiple PUs, and even if the search range is drastically reduced, as in our proposed method. Although we have examined the proposed method in the context of the HM encoder, we believe that the reuse of MV information is also beneficial for both the RD performance and the encoding time of other encoders that do not rely on a full-search motion estimation.

VIII. CODING UNIT STRUCTURE REUSE

A. Observations

We encode ten videos at varying QPs between 22 and 40. We observe similarities in the block structure at the CU level. To quantify the similarity, we evaluate the percentage of the frame area where the CUs have the same depth (i.e., the same size) as in the reference encoding at QP 22. Fig. 16 shows the percentage of the frame area of ten videos where the CU depth is greater, identical, or lower than the depth from the reference encoding at QP 22. The percentage of CUs with the same depth as in the reference encoding decreases as the QP increases (down to approximately 45% at QP 40). Thus, we cannot directly reuse the CU structure information for lower quality encodings, as numerous suboptimal CU size decisions would be made. We also observe that a large majority (>90%) of the area in the lower quality encodings has CUs with depths either lower or equal to the depth in the reference encoding.

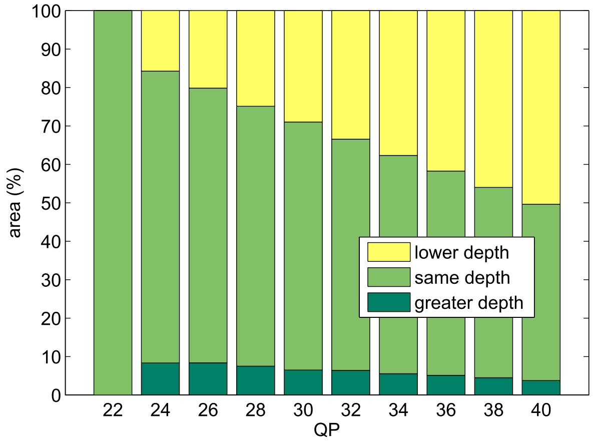


Fig. 16. Average percentage of the area of ten videos with CU depth greater, identical, or lower than the reference encoding at QP 22.

TABLE XII
COMPARISON OF ENCODING WITH CU STRUCTURE REUSE VERSUS CONVENTIONAL ENCODING

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.37%	−0.014 dB	−39.23%
CrowdRun	0.51%	−0.022 dB	−19.23%
DucksTakeOff	0.29%	−0.008 dB	−25.53%
Kimono	0.74%	−0.025 dB	−39.71%
ParkJoy	0.33%	−0.014 dB	−26.13%
ParkScene	0.63%	−0.021 dB	−35.39%
PedestrianArea	1.00%	−0.031 dB	−35.94%
Riverbed	0.37%	−0.015 dB	−40.89%
RushHour	0.14%	−0.002 dB	−35.02%
Sunflower	0.91%	−0.017 dB	−39.04%
Average	**0.53%**	**−0.017 dB**	**−33.61%**

B. Information Reuse

Based on the preceding observations and on the fact that the RDO consists of a quadtree traversal starting with the root (i.e., largest possible CU size), we propose to stop the RDO in the dependent encodings at the depth given by the reference encoding. In fact, the probability to find the optimal CU size is high, as the CU size in a lower quality encoding will be larger or the same (i.e., the CU depth will be lower or the same) with a high probability. A suboptimal decision is made in the few cases where an area in the lower quality encoding should have a smaller CU size than in the reference encoding. As the general goal is to keep the RD performance high, the choice of the highest quality representation as a reference is supported by the fact that the highest quality representation has the blocks with the largest depth on average.

C. Preliminary Results

Table XII shows the encoding results of our implementation of the proposed CU structure reuse method based on HM 16.5 compared with the unmodified encoder. On average, the encoding time is decreased by 33.61% for four representations, while the average BD-rate is increased by 0.53%.

TABLE XIII

COMPARISON OF ENCODING RESULTS FOR 720p BASED ON A 1080p REFERENCE WHEN THE CU STRUCTURE IS REUSED

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.60%	−0.03 dB	−50.86%
CrowdRun	0.57%	−0.03 dB	−19.48%
DucksTakeOff	0.13%	0.00 dB	−22.91%
Kimono	1.02%	−0.04 dB	−44.10%
ParkJoy	0.41%	−0.02 dB	−25.57%
ParkScene	0.82%	−0.03 dB	−34.47%
PedestrianArea	1.39%	−0.06 dB	−40.01%
Riverbed	0.36%	−0.02 dB	−46.15%
RushHour	1.27%	−0.04 dB	−43.04%
Sunflower	1.00%	−0.04 dB	−59.64%
Average	**0.76%**	**−0.03 dB**	**−38.62%**

TABLE XIV

COMPARISON OF ENCODING RESULTS FOR 360p BASED ON A 1080p REFERENCE WHEN THE CU STRUCTURE IS REUSED

Sequence	BD-rate	BD-PSNR	ΔT
BlueSky	0.59%	−0.04 dB	−44.43%
CrowdRun	0.45%	−0.02 dB	−16.61%
DucksTakeOff	0.17%	−0.01 dB	−22.80%
Kimono	1.70%	−0.07 dB	−33.57%
ParkJoy	0.36%	−0.02 dB	−20.78%
ParkScene	0.55%	−0.02 dB	−27.73%
PedestrianArea	2.11%	−0.11 dB	−30.85%
RiverBed	0.46%	−0.02 dB	−35.25%
RushHour	2.38%	−0.10 dB	−34.03%
Sunflower	0.55%	−0.03 dB	−52.03%
Average	**0.93%**	**−0.04 dB**	**−31.81%**

D. Multiple Resolutions

We also observe similarities in the CU structure of a video at different resolutions, in the sense that homogeneous regions tend to be coded with large CUs, whereas highly detailed frame regions tend to be coded with small blocks. If we want to reuse the CU structure information from a high-resolution encoding, we need to match the CU structure at a high resolution to the CU structure at a low resolution.

However, there is no direct correspondence between CUs at different resolutions if the downsampling factor is different from a multiple of 2.

Thus, we propose an algorithm that extracts CU structure information from a high-resolution encoding. We compute a CU structure mask for the low-resolution CTUs as follows. First, the CU at depth 0 from the first CTU is selected in the low-resolution video. Then, we select the corresponding area A in the reference encoding. Next, we measure the percentage p_0 of A encoded at depth (less than or equal to) 0. In general, the percentage p_i with $i \in \{0, 1, 2\}$ is defined as the percentage of the corresponding area in the reference encoding with depth less than or equal to i. If p_0 is greater than or equal to a threshold τ, the current CU is not split, and the process moves on to the next CTU. On the contrary, if p_0 is less than τ, then the current CU is split into four smaller CUs at depth 1. The process is recursively repeated for all the CUs in order to traverse the quadtree. The threshold τ determines how conservative the algorithm is. A method to set τ based on the video content is proposed in [5].

During the encoding of the dependent low-resolution representations, we propose to stop the RDO process at the depth given by the computed CU structure mask, similar to the single-resolution case.

The preliminary results for the CU structure reuse for multiple representations are presented in Tables XIII and XIV for the case of one reference encoding at 1080p and QP 22. The average time reduction of 38.62% and 31.81% for 720p and 360p, respectively, is comparable with the time reduction in the single-resolution case. The RD performance is slightly worse, with a BD-rate increase of 0.76% and 0.93%, respectively.[1]

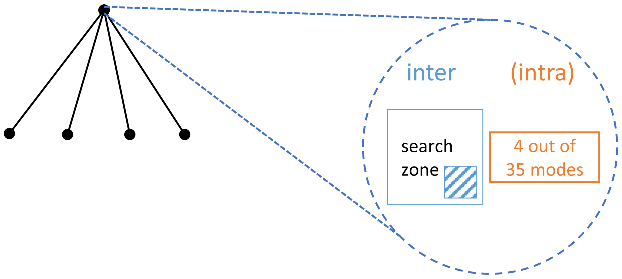


Fig. 17. Conceptual schema of the constrained RDO in the proposed multi-rate encoder. Compared with the original RDO (see Fig. 3), the quadtree traversal is shortened, the intra analysis is potentially skipped, and fewer intra modes and a smaller inter-prediction MV search zone are considered.

IX. MULTI-RATE ENCODING SYSTEM

A. Combined Proposed Methods

So far, we have investigated different possible information reuse methods separately and have examined their individual effect on the RD performance and encoding time. From the previous sections, we have seen that the CU structure reuse leads to the highest encoding time reduction, as expected from Section IV (see Table I). Comparatively, the prediction mode reuse leads to smaller encoding time reductions, while the RD performance loss is small as well (BD-rate increase of 0.15% in the single-resolution case and BD-rate increase of 0.58% and 0.84% in the multiresolution case). The intra mode reuse leads to encoding time reductions around 10% in the case of all intra encoding, but its effect is limited in an encoding configuration that also uses inter prediction. Finally, the MV reuse leads to a relatively low encoding time reduction (on average 6.21%). However, the method is interesting as it improves the RD performance compared with the original HM encoder.

In this section, we consider a multi-rate encoding system that cumulatively implements the different proposed methods, in order to leverage all possible encoding time reductions. Fig. 17 conceptually shows how the RDO in the proposed multi-rate encoder is constrained. The quadtree to be analyzed is shortened with the CU structure reuse method. The intra-analysis part can be skipped with the prediction mode reuse method. In the intra mode reuse method, the number of intra modes to be checked is reduced. Finally, in the MV reuse method, the size of the search zone in the motion estimation is reduced.

[1] These results differ from the results in [5], where four references at 1080p and QP 22, 27, 32, and 37, respectively, were used.

TABLE XV

COMPARISON OF ENCODING RESULTS FOR A FIXED QP REPRESENTATIONS SET

Sequence	**BD-rate (%)**			**BD-PSNR (dB)**			**ΔT (%)**			**ΔT_{12} (%)**
	1080p	720p	360p	1080p	720p	360p	1080p	720p	360p	
BlueSky	0.68	0.62	0.63	−0.027	−0.032	−0.039	−47.72	−53.25	−47.01	−49.35
CrowdRun	0.44	1.16	1.11	−0.018	−0.055	−0.058	−23.91	−27.04	−22.83	−24.76
DucksTakeOff	−0.004	1.83	2.43	0.0001	−0.063	−0.115	−25.27	−39.53	−40.02	−30.19
Kimono	0.70	1.65	1.71	−0.023	−0.066	−0.063	−43.95	−48.78	−38.63	−45.00
ParkJoy	0.16	0.86	1.04	−0.007	−0.039	−0.050	−29.21	−33.90	−28.24	−30.54
ParkScene	0.56	0.99	0.77	−0.018	−0.037	−0.033	−36.33	−37.52	−30.47	−36.27
PedestrianArea	1.09	2.13	2.06	−0.036	−0.099	−0.112	−43.40	−50.29	−40.80	−45.30
Riverbed	0.49	0.80	1.54	−0.021	−0.037	−0.073	−45.45	−78.05	−72.71	−57.27
RushHour	0.89	1.71	2.39	−0.023	−0.058	−0.102	−42.15	−48.01	−38.81	−43.67
Sunflower	0.50	1.27	1.01	−0.018	−0.051	−0.053	−53.02	−61.26	−54.07	−55.58
Average	**0.55**	**1.30**	**1.47**	**−0.019**	**−0.053**	**−0.070**	**−39.04**	**−47.76**	**−41.36**	**−41.79**
	1.11			**−0.047**			**−42.72**			

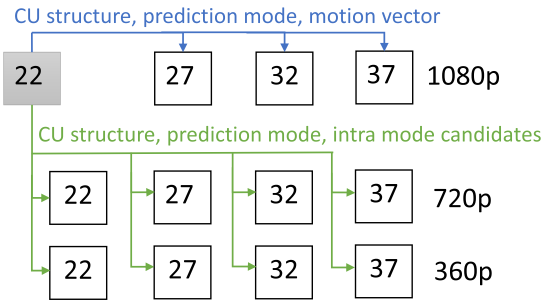

CU structure, prediction mode, motion vector
22
27
32
37
1080p
CU structure, prediction mode, intra mode candidates
22
27
32
37
720p
22
27
32
37
360p

Fig. 18. Schema of the multi-rate encoding system with the reference encoding (gray) and dependent encodings (white) and the corresponding QPs.

So far, we have separately considered a single-resolution system and a multiple-resolutions system. We now consider a system with 12 representations, spanning both multiple resolutions and different SNR qualities, as shown in Fig. 18. The encoding at 1080p and QP 22 is used as reference encoding. For the three 1080p dependent encodings, we implement the proposed CU structure reuse, prediction mode reuse, and MV reuse. We found that incorporating the proposed intra mode reuse method leads to a lower RD performance, without achieving a significant time reduction. This is due to the fact that the intra mode reuse impacts mainly I-frames, which only account for a small fraction of the overall encoding time. For the low-resolution dependent encodings, we implement the proposed CU structure reuse and prediction mode reuse.

B. Results

The encoding results compared with the original HM encoder are presented in Table XV. The BD-rate and BD-PSNR are measured separately for the different resolutions. The overall encoding time difference ΔT_{12} is now measured over the 12 representations, which includes the reference encoding. The RD performance loss is smallest for the 1080p representations with an average BD-rate increase of 0.55%. Averaged over the three resolutions, the BD-rate is increased by 1.11%, and the BD-PSNR is reduced by 0.047 dB. The average encoding time reduction is 41.79%.

C. Rate-Control Encoding

In practical deployments, rate control is generally used instead of fixed QP encoding. To show the effect of our proposed methods for rate control deployments, we now determine a set of 12 representations based on spatial resolution and target bitrate, instead of spatial resolution and fixed QP. To determine the bitrates, we use an average of the bitrates of the videos encoded at QPs 22, 27, 32, and 37. As we have videos at 24, 25, and 50 fps, we determine two bitrates sets: one for the videos at 24 and 25 fps and the other for the videos at 50 fps. The target bitrates are listed in Table XVI.

TABLE XVI

TARGET BITRATES FOR RATE-CONTROL ENCODING (kb/s)

24 fps and 25 fps			**50 fps**		
1080p	**720p**	**360p**	**1080p**	**720p**	**360p**
7,500	5,200	2,300	55,000	45,000	20,000
3,500	2,500	1,100	25,000	20,000	9,000
1,500	1,200	540	10,000	9,000	4,500
800	600	260	5,000	4,500	2,000

We run the multi-rate system with 12 representations where the 1080p representation at the highest bitrate is the reference encoding. We use the default HM rate control. Table XVII shows the encoding results compared with the original HM encoder. The results are comparable with the results from the fixed QP representations set. Again, the 1080p representations show the least RD performance loss with a BD-rate increase of 0.46%. The overall average encoding time reduction is 37.92%, which is slightly less as in the fixed QP case. However, the overall RD performance is also slightly better than in the fixed QP case with an overall average BD-rate increase of 0.96%, instead of 1.11%.

Fig. 19 shows an example of RD curves for the *ParkScene* sequence, both for the original HM encoder and for our proposed multi-rate encoder for the rate-control case. Fig. 20 shows the corresponding encoding time with the original encoder and our proposed multi-rate encoder.

D. Alternative Spatial Resolutions

In addition to the set of 1080p videos used throughout this paper, we assess the impact of our proposed multi-rate system on video sequences with different reference resolutions. We use the sequences *PeopleOnStreet* and *Traffic* with an original resolution of 2500 × 1600 pixels [19], and

TABLE XVII
COMPARISON OF ENCODING RESULTS FOR THE SET BASED ON RATE CONTROL

Sequence	BD-rate (%)			BD-PSNR (dB)			ΔT (%)			ΔT_{12} (%)
	1080p	720p	360p	1080p	720p	360p	1080p	720p	360p	
BlueSky	0.47	1.12	0.98	−0.017	−0.051	−0.057	−41.57	−46.53	−39.63	−42.98
CrowdRun	0.44	0.72	1.09	−0.018	−0.035	−0.074	−24.20	−25.80	−24.93	−24.78
DucksTakeOff	−1.07	0.24	0.73	0.027	−0.010	−0.053	−29.83	−33.98	−32.07	−31.39
Kimono	0.80	0.91	2.31	−0.027	−0.039	−0.103	−42.80	−46.45	−36.84	−43.47
ParkJoy	0.30	0.62	1.86	−0.012	−0.031	−0.144	−29.83	−31.34	−28.38	−30.18
ParkScene	0.65	0.96	1.39	−0.021	−0.037	−0.065	−37.86	−38.72	−32.62	−37.68
PedestrianArea	1.17	1.73	2.00	−0.036	−0.070	−0.108	−41.06	−38.67	−30.85	−39.45
Riverbed	0.47	0.43	0.01	−0.020	−0.018	−0.007	−45.49	−51.39	−45.87	−47.38
RushHour	0.53	1.51	2.71	−0.010	−0.045	−0.116	−38.65	−41.38	−33.07	−39.06
Sunflower	0.87	1.01	1.76	−0.016	−0.028	−0.073	−42.10	−45.26	−38.47	−42.79
Average	**0.46**	**0.93**	**1.48**	**−0.015**	**−0.036**	**−0.080**	**−37.34**	**−39.85**	**−34.27**	**−37.92**
	0.96			**−0.044**			**−37.15**			

TABLE XVIII
COMPARISON OF ENCODING RESULTS FOR VIDEOS WITH ALTERNATIVE SPATIAL RESOLUTIONS

Sequence	BD-rate (%)			BD-PSNR (dB)			ΔT (%)			ΔT_{12} (%)
	1600p	1080p	720p	1600p	1080p	720p	1600p	1080p	720p	
PeopleOnStreet	1.00	0.89	0.75	−0.046	−0.045	−0.043	−26.05	−20.29	−17.91	−23.40
Traffic	0.56	0.51	0.70	−0.020	−0.022	−0.037	−39.96	−37.67	−36.17	−38.88
	480p	360p	240p	480p	360p	240p	480p	360p	240p	
BQMall	0.68	0.71	0.27	−0.030	−0.038	−0.015	−31.41	−31.59	−28.14	−31.01
PartyScene	0.42	0.73	0.44	−0.019	−0.038	−0.022	−26.19	−28.74	−26.44	−26.97
Average	**0.67**	**0.71**	**0.54**	**−0.029**	**−0.035**	**−0.029**	**−30.90**	**−29.57**	**−27.17**	**−30.07**
	0.64			**−0.031**			**−29.21**			

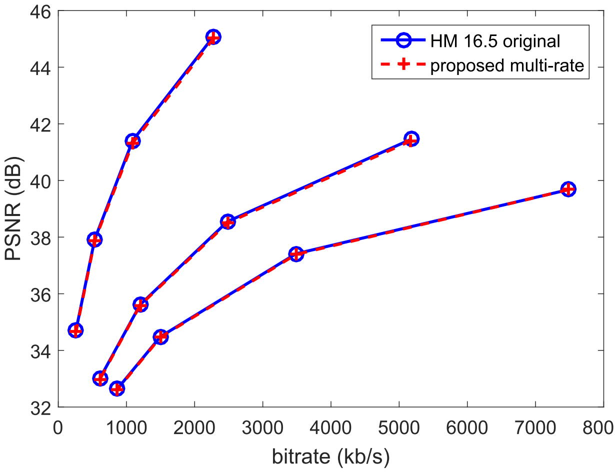


Fig. 19. RD curves for the *ParkScene* sequence at 1080p, 720p, and 360p.

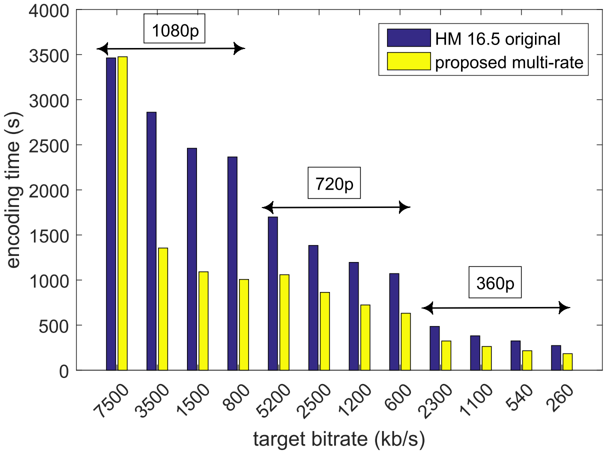


Fig. 20. Encoding time of the 12 representations of the *ParkScene* sequence.

we define two dependent resolutions of 1728 × 1080 and 1124 × 720 pixels. For lower resolutions, we use the sequences *BQMall* and *Partyscene* with an original resolution of 832 × 480 pixels, and define two dependent resolutions of 624 × 360 and 416 × 240 pixels.

Table XVIII shows the encoding results compared with the original HM encoder. The videos are encoded with a fixed QP (22, 27, 32, and 37) at each resolution. On average, the encoding time for 12 representations can be decreased by 30.07%, while the BD-rate is increased by 0.64%. The results are comparable with the results for the 1080p set, although the encoding time reduction is lower, but the BD-rate increase is lower as well.

E. Comparison With Previous Work

De Praeter *et al.* [14] proposed a multi-rate encoding system for HEVC, where the CU structure of the dependent encodings is predicted with machine learning based on the CU structure of the reference encoding. The reference encoding does not have to be the encoding with the highest quality. In their results, they encode five videos *BasketballDrive*, *BQTerrace*, *Cactus*, *Kimono*, and *ParkScene* [19] at three different resolutions (1920 × 1080, 1280 × 720, and 960 × 536 pixels) and different fixed QPs. The results are provided in [14] for different possible references. In order to be consistent, we show their results based on the reference encoding that leads to the lowest BD-rate increase. We encode the same

TABLE XIX
COMPARISON WITH PREVIOUS WORK

Sequence	proposed		De Praeter et al. [14]	
	BD-rate	ΔT	BD-rate	ΔT
BasketballDrive	0.78%	−41.94%	6.4%	−59.6%
BQTerrace	0.36%	−31.91%	5.6%	−70.7%
Cactus	0.79%	−35.74%	6.5%	−59.4%
Kimono	1.12%	−46.02%	4.7%	−57.8%
ParkScene	0.77%	−36.51%	4.8%	−57.4%
Average	**0.76%**	**−38.42%**	**5.6%**	**−61.0%**

video sequences at the same three resolutions using the same fixed QPs.

Table XIX shows the average BD-rate and the overall time reduction over all the representations for the method in [14] and for our proposed method. Although our method achieves a lower overall time reduction, the average BD-rate increase of our proposed method is less than 0.8%, which is very close to the original performance of HEVC. In contrast, the average BD-rate increase with the method in [14] is 5.6% on average, which may be prohibitive for video providers, as storage and transmission costs increase.

X. CONCLUSION

In order to reduce the complexity of encoding multiple representations for an adaptive HTTP streaming scenario, we have considered an HEVC multi-rate encoder in this paper. We have identified four possible encoding information types from a high-quality reference encoding that can be reused in order to speed up lower quality-dependent encodings. We have observed that a direct reuse of the encoding decisions is generally not possible if the RD performance of the multi-rate encoder has to be high. Thus, we have proposed methods that reuse the encoding information to constrain the RDO of the dependent encodings, both in the case of a single-spatial resolution and in the case of multiple resolutions. We have evaluated the impact of these individual methods on the RD performance and on the overall encoding time. Finally, we have shown that the different methods can be combined, and we have proposed a multi-rate encoder that cumulatively leverages the different proposed methods. Results show that the overall encoding time is reduced on average by 38% for 12 representations in the case of rate-control-based encoding, while the BD-rate increases by less than 1% compared with the reference software HM. The proposed methods are in principle also applicable to other HEVC encoders.

REFERENCES

[1] T. Stockhammer, “Dynamic adaptive streaming over HTTP—Standards and design principles,” in *Proc. ACM Conf. Multimedia Syst. (MMSys)*, Santa Clara, CA, USA, Feb. 2011, pp. 133–144.

[2] D. H. Finstad, H. K. Stensland, H. Espeland, and P. Halvorsen, “Improved multi-rate video encoding,” in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dana Point, CA, USA, Dec. 2011, pp. 293–300.

[3] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, “Overview of the High Efficiency Video Coding (HEVC) standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[4] D. Schroeder, P. Rehm, and E. Steinbach, “Block structure reuse for multi-rate High Efficiency Video Coding,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Quebec City, QC, Canada, Sep. 2015, pp. 3972–3976.

[5] D. Schroeder, A. Ilangovan, and E. Steinbach, “Multi-rate encoding for HEVC-based adaptive HTTP streaming with multiple resolutions,” in *Proc. IEEE Int. Workshop Multimedia Signal Process. (MMSP)*, Xiamen, China, Oct. 2015, pp. 1–6.

[6] S. Wang, S. Ma, S. Wang, D. Zhao, and W. Gao, “Rate-GOP based rate control for High Efficiency Video Coding,” *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1101–1111, Dec. 2013.

[7] B. Li, H. Li, L. Li, and J. Zhang, “λ domain rate control algorithm for High Efficiency Video Coding,” *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3841–3854, Sep. 2014.

[8] Y. Chen, Z. Wen, J. Wen, M. Tang, and P. Tao, “Efficient software H.264/AVC to HEVC transcoding on distributed multicore processors,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 6, pp. 1423–1434, Aug. 2015.

[9] A. J. Díaz-Honrubia, J. L. Martínez, P. Cuenca, J. A. Gamez, and J. M. Puerta, “Adaptive fast quadtree level decision algorithm for H.264 to HEVC video transcoding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 154–168, Jan. 2016.

[10] L. P. Van, J. De Cock, A. J. Díaz-Honrubia, G. Van Wallendael, S. Van Leuven, and R. Van de Walle, “Fast motion estimation for closed-loop HEVC transrating,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Paris, France, Oct. 2014, pp. 2492–2496.

[11] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the scalable video coding extension of the H.264/AVC standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[12] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, “Overview of SHVC: Scalable extensions of the High Efficiency Video Coding standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 20–34, Jan. 2016.

[13] M. Xu, Z. Ma, and Y. Wang, “One-pass mode and motion decision for multilayer quality scalable video coding,” *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4250–4262, Nov. 2015.

[14] J. De Praeter *et al.*, “Fast simultaneous video encoder for adaptive streaming,” in *Proc. IEEE Int. Workshop Multimedia Signal Process. (MMSP)*, Xiamen, China, Oct. 2015, pp. 1–6.

[15] Apple. *Using HTTP Live Streaming*, accessed on Jan. 18, 2016. [Online]. Available: http://goo.gl/fJIwC

[16] Netflix. *Per-Title Encode Optimization*, accessed on Jan. 18, 2016. [Online]. Available: http://techblog.netflix.com/2015/12/per-title-encode-optimization.html

[17] L. Toni, R. Aparicio-Pardo, K. Pires, G. Simon, A. Blanc, and P. Frossard, “Optimal selection of adaptive streaming representations,” *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 11, no. 2s, Feb. 2015, Art. no. 43.

[18] (2015). *HEVC Reference Software HM 16.5*, accessed on Jan. 18, 2016. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_hevcsoftware/tags/hm-16.5/

[19] F. Bossen, *Common HM Test Conditions and Software Reference Configurations*, document JCTVC-L1100, Jan. 2013.

[20] Xiph.org Foundation. (2015). *Xiph.org Video Test Media*, accessed on Jan. 18, 2016. [Online]. Available: http://media.xiph.org/video/derf/

[21] G. Bjøntegaard, *Calculation of Average PSNR Differences Between RD-Curves*, document VCEG-M33 ITU-T Q6/16, Austin, TX, USA, Apr. 2001.

[22] G. Bjøntegaard, *Improvements of the BD-PSNR Model*, document VCEG-AI11, ITU-T SG16/Q6, 2008, p. 35.

[23] F. Bossen, B. Bross, K. Sühring, and D. Flynn, “HEVC complexity and implementation analysis,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1685–1696, Dec. 2012.

[24] I.-K. Kim, J. Min, T. Lee, W.-J. Han, and J. Park, “Block partitioning structure in the HEVC standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1697–1706, Dec. 2012.

[25] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, “Intra coding of the HEVC standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1792–1801, Dec. 2012.

[26] F. Bossen, D. Flynn, K. Sharman, and K. Sühring. (2015). *HM Software Manual*, accessed on Jan. 18, 2016. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.5/doc/software-manual.pdf

[27] N. Purnachand, L. N. Alves, and A. Navarro, “Improvements to TZ search motion estimation algorithm for multiview video coding,” in *Proc. 19th Int. Conf. Syst., Signals Image Process. (IWSSIP)*, Vienna, Austria, Apr. 2012, pp. 388–391.

Damien Schroeder (S'12) received the Dipl.-Ing. degree in electrical engineering and information technology from the Technische Universität München (TUM), Munich, Germany, in 2011 and the Diploma degree from Supélec, Gif-sur-Yvette, France. He is currently pursuing the Ph.D. degree with TUM.

He joined the Chair of Media Technology, TUM, in 2012, where he is currently a member of the Research and Teaching Staff. His current research interests include video coding, adaptive HTTP streaming, and video quality.

Adithyan Ilangovan received the bachelor's degree from IIT Hyderabad, Telangana, India, and the M.Sc. degree in communications engineering from the Technische Universität München (TUM), Munich, Germany, in 2015.

He was involved in High Efficiency Video Coding-based multi-rate encoding with the Chair of Media Technology. TUM, during his master thesis. He is currently a Software Engineer for future multimedia delivery systems with Nomor Research GmbH, Munich. His current research interests include adaptive streaming over HTTP, video quality, and multimedia delivery formats.

Martin Reisslein (A'96–S'97–M'98–SM'03–F'14) received the Ph.D. degree in systems engineering from the University of Pennsylvania, Philadelphia, PA, USA, in 1998.

He is currently a Professor with the School of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe, AZ, USA.

Prof. Reisslein served as the Editor-in-Chief of the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS from 2003 to 2007 and as an Associate Editor of the IEEE/ACM TRANSACTIONS ON NETWORKING from 2009 to 2013. He currently serves as an Associate Editor of the IEEE TRANSACTIONS ON EDUCATION, *Computer Networks*, and *Optical Switching and Networking*.

Eckehard Steinbach (M'96–SM'08–F'15) received Dipl.-Ing. degree in electrical engineering from Karlsruhe Institute of Technology, Karlsruhe, Germany, the University of Essex, Colchester, U.K., and ESIEE Paris, Marne-la-Vallée, France.

He was a member of the Research Staff with the Image Communication Group, the University of Erlangen–Nuremberg, Erlangen, Germany, from 1994 to 2000, from which he received the Engineering Doctorate degree in 1999. From 2000 to 2001, he was a Post-Doctoral Fellow with the Information Systems Laboratory, Stanford University, Stanford, CA, USA. In 2002, he joined the Department of Electrical Engineering and Information Technology, Technische Universität München, Munich, Germany, where he is currently a Full Professor of media technology. His current research interests include audio-visual-haptic information processing and communication, as well as networked and interactive multimedia systems.