Optimal HEVC Encoding Based on GOP Configurations

Gangadharan Esakki, Venkatesh Jatla, and Marios S. Pattichis
{gesakki, venkatesh369, pattichi}@unm.edu
image and video Processing and Communications Lab (ivPCL)
Department of Electrical and Computer Engineering
The University of New Mexico, United States

***Abstract*—The emergence of HEVC software implementations allows for several different encoding options. Unfortunately, there is no established method for selecting optimal encoding configurations. The current paper proposes the use of a multi-objective optimization framework for selecting optimal encodings that can be subsequently used for solving constrained optimization problems in quality, bitrate, and encoding time. The proposed optimization framework is used to select optimal configurations from 3,600 possibilities based on GOP configurations and other relevant HEVC software options.**

We demonstrate our approach using the x265 encoder with examples from the UT LIVE video quality database and a standard 2K video example. The results demonstrate the success of the proposed approach by selecting optimal configurations and eliminating several suboptimal encodings.

***Index Terms*—HEVC, multi-objective optimization, Pareto front, GOP configurations.**

I. Introduction

The emergence of the HEVC standard has provided numerous encoding options that affect encoding times, video quality, and required bitrate [1]–[4]. In general, there are complex inter-dependencies between encoding time, video quality, and required bitrate. Furthermore, these complex relationships are strongly affected by video content. Beyond the standard use of rate-distortion theoretic methods, this paper introduces optimization methods based on a family of GOP configurations and other HEVC encoding parameters for jointly optimizing encoding time, video quality, and bitrate.

To formally define the multi-objective optimization framework, let $\mathtt{Q}$ denote a metric of video quality, $\mathtt{BPS}$ denote the number of bits per pixel, and $\mathtt{T}$ denote the required encoding time. An optimal video encoding configuration needs to simultaneously maximize image quality, minimize the required bitrate and encoding time. More compactly, in vector form, the multi-objective optimization framework requires that we solve:

$$\min_{\mathtt{c}} \left(-\mathtt{Q}(\mathtt{c}), \mathtt{BPS}(\mathtt{c}), \mathtt{T}(\mathtt{c})\right) \tag{1}$$

for the optimal encoding configuration $\mathtt{c}$. Here, we note that the negative sign for video quality comes from the fact that maximizing the video quality is equivalent to minimizing the negative of video quality. Furthermore, in what follows, we will drop the $\mathtt{c}$ argument from the objectives. In other words, we write $\mathtt{Q}, \mathtt{BPS}, \mathtt{T}$ with the understanding that they depend on the configuration $\mathtt{c}$.

The solution of the vector optimization problem given in (1) defines a Pareto front. The Pareto front is defined by the set of configurations for which no other configuration can be found that improves on all of the objectives $(\mathtt{Q}, \mathtt{BPS}, \mathtt{T})$ at the same time. Thus, a configuration $\mathtt{c}_{opt}$ is optimal if there is no way to find another configuration $\mathtt{c}_{other}$ that gives better image quality, lower bitrate, and requires less encoding time.

In order to select an optimal configuration, we also define optimal constrained optimization modes as (see related related work in [5]–[7]. Here, the goal is to find optimal solutions subject to realistic constraints on encoding ($\mathtt{T} \leq \mathtt{T}_{\mathtt{max}}$), bitrate ($\mathtt{BPS} \leq \mathtt{BPS}_{\mathtt{max}}$), and image quality ($\mathtt{Q} \geq \mathtt{Q}_{\mathtt{min}}$). We are primarily interested in optimal modes defined as [6]: (i) minimum encoding time mode, (ii) minimum bitrate mode, and (iii) maximum video quality mode, subject to opposing constraints from the two remaining objectives.

There are several challenges associated with the application of the multi-objective framework to HEVC encoding. First, we note that the Pareto-front will significantly vary from video to video, and even from GOP to GOP within each video. In [5], [6], the authors considered a bottom up approach that allowed the variation of DCT hardware cores and the quantization parameter (QP) for each image. In [7], in another bottom-up approach, the authors considered a multi-objective optimization approach that was applied to HEVC intra-coding.

Here, we take a top down approach where we consider the development of a unifying approach for all x265 HEVC configurations. Second, it is important to acknowledge that the current x265 HEVC configurations provide a very sparse sampling of the space of encoding time - video quality - bitrate. Unfortunately, such sparsity imposes fundamental limits to the usefulness of the proposed, multi-objective optimization framework [8]. Thus, to address this problem, the current paper introduces extended HEVC configurations in x265 that include new GOP configurations. This combination of new GOP configurations with the variation of QP, De-blocking filtering, and other parameters produces a large number of configurations that allows for significantly better sampling of the multi-objective space. Third, the use of extended HEVC configurations requires the compression of each video under each configuration and thus impose significant storage requirements. To address this issue, we introduce an offline approach that only stores the optimal configuration vectors

```
function OptEnc(V, Vc, ParetoFront, OptPars)
▷ Input: video V, Pareto front in ParetoFront,
▷       optimization mode specified in OptPars.
▷ Output: compressed video in Vc.

    ParetoEntry ← Find an optimal solution specified
            by OptPars that lies on ParetoFront.
    if (valid ParetoEntry has been found) then
        Vc ← Compress V using configuration
            (P, GOPconfig, ParVec)
            extracted from ParetoEntry.
    else
        ParetoEntry ← Search ParetoFront
            for an entry that violates the constraints by the
            least amount.
        Vc ← Compress V using configuration
            (P, GOPconfig, ParVec)
            extracted from ParetoEntry.
    end if
end function
```

Fig. 1. Optimal mode encoding using the Pareto front.

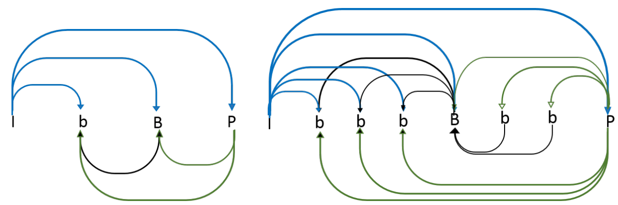
(without the compressed videos) associated with the Pareto front. Then, the optimal configuration is selected by solving the optimization problem associated with each optimization mode. The optimally compressed video is then re-produced by applying the optimal configurations with the x265 encoder.

In terms of related work, we also mention earlier research focused on the use of multiple objectives in hardware implementations, emergency video transmission, and intra encoding. We have the use of parallel cores for single-pixel processors in [9], the development of one-dimensional filtering in [10], and two-dimensional filter bank approaches in [11]. More recently, we have the development of scalable and fast architectures for the computation of the Discrete Periodic Radon Transform in [12]. In [13], we also have the development of adaptive HEVC compression methods for emergency scenery videos. Similarly, in [14], we have developed optimization methods that can be used for intra-coding video compression. The current paper fundamentally differs from [13], [14] in that there is a focus on general-purpose videos using new GOP configurations that account for most of the Pareto-optimal configurations (see section III).

The rest of the paper is organized into four sections. In section II, we provide the methodology used in the paper. We provide the results in III and give concluding remarks in IV.

II. Methodology

We summarize the proposed method in Figure 1. For each video, we pre-compute its Pareto front. As stated earlier, the resulting Pareto front is simply expressed in terms of a mapping from each optimal GOP configuration, HEVC profile, and related parameters to the three objective functions (video quality, encoding time, and bitrate requirements). For any given optimization mode, we select and apply the optimal encoding configuration as shown in Fig. 1.



(a) New GOP B2 configuration (b) New GOP B6 configuration

Fig. 2. New GOP configurations. (a) Extended GOP configuration by removing a **b** frame. (b) Extended GOP configuration by adding a **b** frame.

As stated earlier, efficient implementation of the optimization modes requires an extension of the standard GOP configurations. We present a diagram with some of the new GOP configurations in Fig. 2. We provide a detailed summary of the proposed GOP configurations in Table I.

From the Pareto front, we can select optimal configurations that can solve the following constrained optimization problems:

- ***Minimum encoding time mode:***

$$\min_{\mathtt{EP}} \mathtt{T} \quad \text{subject to} \quad (\mathtt{Q} \geq Q_{min}) \text{ and } (\mathtt{BPS} \leq \mathtt{BPS}_{\mathtt{max}}) \tag{2}$$

  In this mode, the goal is to minimize encoding time provided that the video can be communicated within the given bitrate and it is of sufficiently good quality.

- ***Minimum bitrate mode:***

$$\min_{\mathtt{EP}} \mathtt{BPS} \quad \text{subject to} \quad (\mathtt{Q} \geq Q_{min}) \text{ and } (\mathtt{T} \leq \mathtt{T}_{\mathtt{max}}). \tag{3}$$

  In this mode, the goal is to minimize bandwidth requirements provided that the video is of sufficient quality and we do not spend a large amount of time encoding it.

- ***Maximum video quality mode:***

$$\max_{\mathtt{EP}} \mathtt{Q} \quad \text{subject to} \quad (\mathtt{BPS} < \mathtt{BPS}_{\mathtt{max}}) \text{ and } (\mathtt{T} < \mathtt{T}_{\mathtt{max}}). \tag{4}$$

  Here, the goal is to reconstruct the video with the highest possible video quality that does not require more bandwidth that is available and within reasonable encoding time.

III. Results

For testing our approach, we consider optimal encoding for videos as shown in Figs. 3(a), 3(b), and 3 (c) [15]–[18]. For measuring the encoding time, we run the x265 ver 1.4 reference software [4] on a Windows 8 64-bit platform with 64GB RAM using an Intel(R) Xeon(R) CPU E5-2630v3 microprocessor with 8 cores (16 threads) running at 2.40 GHz. Furthermore, we only consider the use of PSNR for evaluating video quality although our approach can also be applied to other metrics provided that they can be computed fast.

TABLE I

ENCODER GOP CONFIGURATION SETUP BROKEN INTO TWO GROUPS. GROUP A PRESETS ARE EXTENSIONS OF GOP B4 INTO NEW GOP B2,B6 AND CONSIST OF: ULTRA FAST (U), SUPER FAST (S), VERY FAST (V), FASTER (FR), FAST (F), MEDIUM (M), AND SLOW (S). GROUP B PROFILES ARE EXTENSIONS OF DEFAULT GOP B8 INTO NEW GOP B6,B10 AND CONSISTS: SLOWER (SL), VERY SLOW (VS) AND PLACEBO (P). THERE IS A TOTAL OF 3600 POSSIBLE CONFIGURATIONS.

Parameter	Profile Group A	Profile Group B
Presets	U, S, V, Fr, F, M, S	Sl, Vs, P
GOP	AI, **B2**, B4, **B6**, ZL	AI, **B6**, B8, **B10**, ZL
GOP Str	Open/Close	Open/Close
QP	22, 27, 32, 37, 42	22, 27, 32, 37, 42
SAO	On/Off	On/Off
DBF	On/Off	On/Off
Tuning	PSNR, ZL, FD	PSNR, ZL, FD
Configs.	360 per profile	360 per profile

TABLE II

OPTIMAL GOP CONFIGURATIONS. THE NEW GOP CONFIGURATIONS ARE SHOWN IN BOLD.

GOP conf.	Optimal Configurations (%)		
	Jockey	Pa	Rb
AI	63 (14.38%)	101 (11.46%)	34 (59.64%)
B2	145 (33.1%)	208 (23.6%)	5 (8.77%)
B4	25 (5.7%)	179 (20.31%)	3 (5.2%)
B6	33 (7.53%)	136 (15.43%)	1 (1.75%)
B8	6 (1.36%)	27 (3.06%)	0 (%)
B10	0 (0%)	17 (1.92%)	0 (%)
ZL	166 (37.89%)	213 (24.17%)	14 (24.56%)
Pareto	**438** (100%)	**881** (100%)	**57** (100%)

Overall, as we document in Fig. 3, we find that we can generate relatively dense Pareto fronts for the Jockey and Pedestrian video examples. On the other hand, the very complex motions of the riverbed example generated a relatively sparse Pareto front. Furthermore, we note that the new GOP configurations contributed (i) 40.64 % of the optimal 438 configurations for the UHD video, and (ii) 40.97 % of the optimal 881 configurations for the Pedestrian video. Refer to Table II for more details.

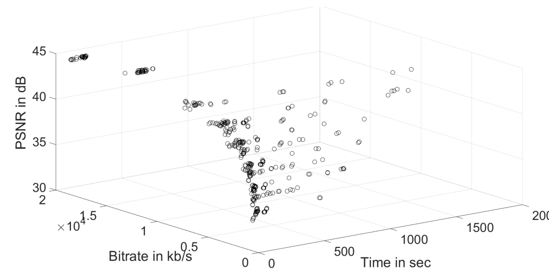
The relatively dense Pareto fronts for the Jockey and Pedestrian videos allow fine optimization of the modes described by equations (2), (3), and (4). We present three DRASTIC mode optimization examples in Table III. For the examples, all of the constraints have been met.

Also, as expected, the optimal mode result from finding solutions that are close to the bounds required by at-least one of the constraints. To see this, we consider the maximum quality mode in Table III that requires that $\mathtt{T}_{\mathtt{max}} < 5$ seconds and $\mathtt{BPS}_{\mathtt{max}} < 5000$ bps. Then, the maximum quality mode requires 4.8 seconds of total encoding time that is close to the upper bound of 5 seconds. On the other hand, we note that there was a lot more bitrate that could have been used. Yet, the optimization method maximized image quality to a level 42.8 dB using less bitrate.

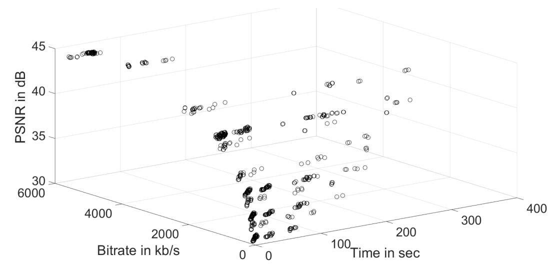

(a) Jockey [18].


(b) Pedestrian [15], [16].

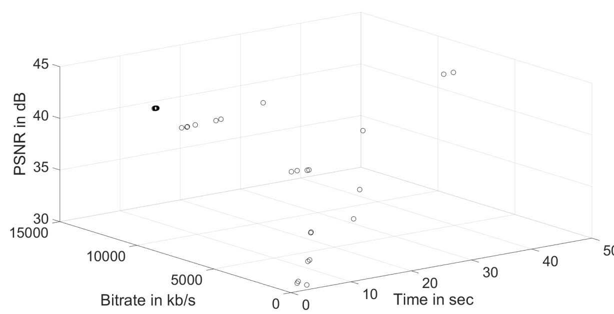

(c) Riverbed [15], [16].



(d) Pareto front for UHD video: Jockey (1920x1080, 30 fps, 150 frames).



(e) Pareto Front for Pedestrian video (768x432, 25 fps, 250 frames).



(f) Pareto Front for Riverbed video 768x432

Fig. 3. Test videos and resulting Pareto fronts. (a) UHD video with strong predictable, translational motions. (b) Pedestrian video with multiple, yet predictable, translational motions. (c) Riverbed video with very complicated motions created by the flowing water. (d) Pareto front for UHD video demonstrating a relatively dense front. (e) Pareto front for Pedestrian video demonstrating a relatively dense front. (f) Pareto front for Riverbed video with fewer optimal points on pareto front.

TABLE III
MODE OPTIMIZATION. WE MEASURE BITRATE IN BITS PER SECOND, PSNR IN dB, AND TIME IN SECONDS WE USE BR FOR BITRATE, Q FOR IMAGE QUALITY, AND T FOR ENCODING TIME. IN EACH CASE, WE PRESENT THE QUANTITY THAT IS OPTIMIZED IN BOLD. REFER TO TABLE I FOR ABBREVIATIONS. REFER TO (2), (3), AND (4) FOR DEFINTIONS OF THE MODES AND THE CONSTRAINTS. NOTE THAT ALL OF THE CONSTRAINTS HAVE BEEN MET IN THESE EXAMPLES.

2KJockey 1920x1080 @30 FPS, 150 frames

Mode	GOP	Profile	Time	Bitrate	PSNR
Max Q	B2	SF	4.8	4167.3	**42.8**
Constraints			5.0	5000.0	
Min T	B2	M	**6.9**	1049.2	39.1
Constraints				1300.0	39.0

Pedestrian 768x432 @25 FPS, 250 frames

Mode	GOP	Profile	Time	Bitrate	PSNR
Min BR	ZL	Fr	2.3	**147.0**	31.9
Constraints			3.0		31.0

IV. CONCLUSION AND FUTURE WORK

In this paper, we have presented a unifying framework that allows us to jointly optimize for encoding time, bitrate, and image quality. The approach eliminates several suboptimal configurations. Furthermore, except of the rare case of irregular motions, a set of new GOP configurations can be used to generate dense samplings of the Pareto front. The new GOP configurations also enable fine optimization methods that can be used to minimize encoding time, maximize quality, or reduce bandwidth requirements.

Current research is focused on modeling the Pareto front so as to eliminate the need to compute all possible configurations. Furthermore, we are also investigating methods for dynamically adjusting the constrained optimization modes based on video content.

V. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under NSF AWD CNS-1422031.

REFERENCES

[1] G.J. Sullivan, J. Ohm, Woo-Jin Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *, IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.

[2] J. Ohm, G.J. Sullivan, H. Schwarz, Thiow Keng Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards x2014;including high efficiency video coding (hevc)," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1669–1684, Dec 2012.

[3] F. Bossen, B. Bross, K. Suhring, and D. Flynn, "Hevc complexity and implementation analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1685–1696, Dec 2012.

[4] MulticoreWare Inc, "X265 software documentation," https://x265.readthedocs.org/en/default/, 2015.

[5] Yuebing Jiang and MariosS. Pattichis, "A dynamically reconfigurable architecture system for time-varying image constraints (drastic) for motion jpeg," *Journal of Real-Time Image Processing*, pp. 1–17, 2014.

[6] Yuebing Jiang and M. Pattichis, "Dynamically reconfigurable dct architectures based on bitrate, power, and image quality considerations," in *19th IEEE International Conference on Image Processing (ICIP)*, Sept 2012, pp. 2465–2468.

[7] Yuebing Jiang, G. Esakki, and M. Pattichis, "Dynamically reconfigurable architecture system for time-varying image constraints (drastic) for hevc intra encoding," in *2013 Asilomar Conference on Signals, Systems and Computers*, Nov 2013, pp. 1112–116.

[8] Gangadharan Esakki, "Dynamic switching of gop configurations in high efficiency video coding (hevc) using relational databases for multi-objective optimization," M.S. thesis, Electrical and Computer Engineering UNM, jul 2014.

[9] D. Llamocca and M. Pattichis, "A dynamically reconfigurable pixel processor system based on power/energy-performance-accuracy optimization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 3, pp. 488–502, March 2013.

[10] Daniel Llamocca, Marios Pattichis, and G. Alonzo Vera, "Partial reconfigurable fir filtering system using distributed arithmetic," *Int. J. Reconfig. Comput.*, vol. 2010, pp. 4:1–4:14, Feb. 2010.

[11] Daniel Llamocca and Marios Pattichis, "Dynamic energy, performance, and accuracy optimization and management using automatically generated constraints for separable 2d fir filtering for digital video processing," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 7, no. 4, pp. 4:1–4:30, Dec. 2014.

[12] C. Carranza, D. LLamocca, and M.S. Pattichis, "Fast and scalable computation of the forward and inverse discrete periodic radon transform," *IEEE Transactions on Image Processing*, accepted.

[13] Z. Antoniou, A.S. Panayides, and M.S. Pattichis, "Adaptive emergency scenery video communications using hevc for responsive decision support in disaster incidents," in *37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015, pp. 173–176.

[14] Y. Jiang, C. Zong, and M.S. Pattichis, "Scalable hevc intra frame complexity control subject to quality and bitrate constraints," in *3rd IEEE Global Conference on Signal and Information Processing*, 2015, In press.

[15] Kalpana Seshadrinathan, Rajiv Soundararajan, Alan C Bovik, and Lawrence K Cormack, "A subjective study to evaluate video quality assessment algorithms," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2010, pp. 75270H–75270H.

[16] K. Seshadrinathan, R. Soundararajan, A.C. Bovik, and L.K. Cormack, "Study of subjective and objective quality assessment of video," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1427–1441, June 2010.

[17] The University of Texas at Austin, "Laboratory for image and video engineering," http://live.ece.utexas.edu/research/Quality/index.htm, 2015.

[18] Tampere University of Technology, "Ultra video group," http://ultravideo.cs.tut.fi/#testsequences/, 2015.