COMPLEXITY CONTROL OF HEVC FOR VIDEO CONFERENCING

Xin Deng[*]

Department of Electrical and Electronic Engineering
Imperial College London
x.deng16@imperial.ac.uk

Mai Xu[†]

School of Electronic Information Engineering
Beihang Univeristy
maixu@buaa.edu.cn

ABSTRACT

In this paper, we propose an effective complexity control approach for video conferencing scenarios on HEVC platform. A complexity control formulation is established to determine the number of depth-constrained largest coding units (LCUs) according to the target complexity. By limiting the maximum depths of different LCUs to different levels, the encoding complexity can be controlled with high accuracy. Different from other approaches, both the objective and perceptual-driven video quality are kindly preserved through taking both the objective and subjective weight maps into consideration when controlling the complexity. The experimental results demonstrate that our approach outperforms the state-of-the art approach with higher control accuracy. Also, despite of complexity reduction, our approach keeps the objective and perceptual-driven quality well.

***Index Terms*—** HEVC, Encoding complexity control, Video conferencing

1. INTRODUCTION

Video conferencing is a live and visual communication method, which aims to provide high-resolution images and high-fidelity audio signals for people from different places. The advent of customer services like Microsoft's Skype, Apple's Facetime and Cisco's Meeting server, makes video conferencing more and more ubiquitous in people's daily life. However, the encoding of high-resolution videos, e.g., 4K and 8K, is such a time-consuming job that the low-delay transmission need of video conferencing cannot be satisfied. Thus, it is quite necessary to control the encoding complexity of video conferencing.

Some works[1, 2, 3, 4, 5, 6, 7] have been done to control the encoding complexity of HEVC. Specifically, Correa *et.al* [1] designed a method controlling the encoding complexity of HEVC in Group of Pictures (GOP) level, through adjusting the operational configurations during encoding time. Deng *et.al* [2] proposed a HEVC complexity control method in largest coding unit (LCU) level. Relying on the concept of subjective weights, they reduce the coding depths of LCUs with smaller weights to achieve control by solving a distortion-complexity formulation. Recently, based on a set of early termination conditions, Moreno et al. [3] proposed a complexity control approach for HEVC. However, to our best knowledge, there is no work done on complexity control for video conferencing. Actually, by leveraging the property of video conferencing, further improvements in control accuracy and video quality can be achieved as shown in this paper.

The quardtree-based coding tree unit (CTU) partitioning scheme [8] is an advance in HEVC. However, most time-consuming components are included in it. In this scheme, each frame is divided into equal-sized blocks called LCUs. The size of LCU is designated by the encoder, default as 64×64. Another important parameter set by the encoder is the allowed maximum LCU splitting depth, or the maximum depth. It decides the size of the smallest coding unit (SCU), with the default depth as 3, indicating that 64×64 LCU can be split into 8×8 SCUs. Before the 64×64 LCU gets its optimal depth, the rate-distortion-optimization (RDO) process should be done 85 ($= 1 + 4 + 4^2 + 4^3$) times. Obviously, the larger the maximum depth is, the more encoding time is consumed. However, the optimal depth is not always equal to the maximum depth, which is actually highly content-dependent. For example, as we can see in Fig.1, the texture of the wall is quite homogenous, and the optimal depths of most LCUs in this region are 0, despite of their maximum depths being 3. Thus, the basic complexity reduction thought in our approach is to predict the optimal depths of LCUs and then reduce their maximum depths based on the predicted optimal depths. As long as the prediction is accurate, there exists no bit-rate increase or PSNR loss. The aim of our approach is to control the encoding complexity of HEVC, with the controlling mechanism based on the following observation: when the maximum depth is reduced to a fixed value, the encoding complexity takes a nearly same proportion despite of sequence content. Fig.2 shows the complexity proportion occupied by different maximum depths. Specifically, when maximum depth is reduced from 3 to 2, 1, and 0, the complexity proportion is decreased from 1.00 to 0.65, 0.38 and 0.20. We have tested sequences with different resolutions and find this relationship

[*]Thanks to CSC Imperial Scholarship for funding.

[†]Mai Xu is the corresponding author. Thanks to NSFC under grant number 61537037 for funding.

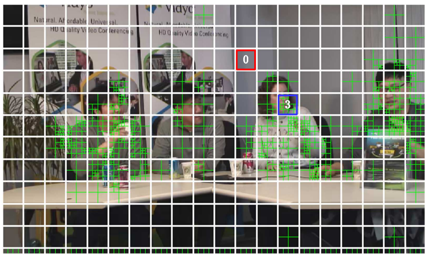


Fig. 1. The picture is the 12-th frame of *Fourpeople* which is video conferencing. The green lines indicate the optimal LCU partition results. The number in the red/blue box is the optimal depth for that LCU.

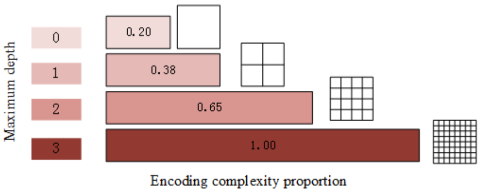


Fig. 2. Encoding complexity proportion occupied by different maximum depths.

applies with only a little difference.

In this paper, we propose a complexity control approach for video conferencing encoding using HEVC. The complexity is controlled by a proposed complexity control formulation. The basic idea is restricting the maximum depths of LCUs with low importance. The advantage of our approach is that in the process of complexity control, both the objective and subjective weight maps are considered, and thus the objective and subjective video quality can be preserved simultaneously.

2. PROPOSED METHOD

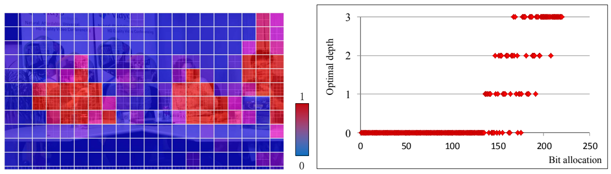
The basis of nearly all complexity control approaches is complexity reduction [9, 10, 11, 12]. In our complexity control approach, the core of complexity reduction is to reduce the maximum depths of LCUs with low importance. The importance of LCUs is measured from two aspects: objective and subjective.

2.1. Objective weight map

The objective weight map is used to preserve objective quality and coding efficiency. Here, we propose to use the bit-allocation map as the objective weight map, because we find that the bit-allocation map can tally with the optimal depth

Table 1. Average results of $\mathcal{P}(D|B)$ of *Fourpeople*

$\mathcal{P}(D\|B)$	D=0	D=1	D=2	D=3
$B < 20$	**99.89**	0.11	0.00	0.00
$20 \leq B < 40$	**99.43**	0.57	0.00	0.00
$40 \leq B < 60$	**98.61**	1.36	0.03	0.00
$60 \leq B < 80$	**61.57**	23.76	10.75	3.92
$80 \leq B < 100$	8.80	17.30	29.88	**44.02**



(a) (b)

Fig. 3. (a) is the objective weight map of 12-th frame of *Fourpeople*, i.e., bit allocation map of its previous frame. (b) shows the relationship between optimal depth and bit allocation. The horizontal axis in (b) is the ascending order of bits allocated to LCUs.

allocation well. As we can see in Figure 3-(a), the LCUs allocated with more bits tend to have larger optimal depths. Figure 3-(b) describe the relationship between bit allocation and optimal depth for 3-(a). Here, one significant observation is that LCUs with smaller bits have great chance being not split, i.e., the optimal depth is 0. Let b_j be the bits allocated to the j-th LCU, we can get the normalised objective weight of the j-th LCU, $W_o(j) = \frac{b_j}{b_{max}}$, where b_{max} is the largest bits among all LCUs in a frame.

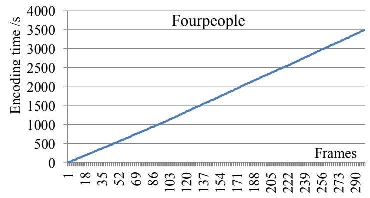
In order to accurately analyse the dependency between the optimal depth and bit allocation, $\mathcal{P}(D|B)$ is adopted, where D denotes the event that the optimal depth is 0, 1, 2 or 3, and B is the bit ascending order. For example, $\mathcal{P}(D = 0|B < 20)$ indicates the probability of event that the optimal depth of LCU is 0 when its allocated bit is ordered less than 20%. Table 1 shows the average results of $\mathcal{P}(D|B)$ of *Fourpeople*. We can see that when the bit order is less than 20%, the probability of the event that LCUs do not split (i.e., D=0) is pretty high, i.e., 99.89. Thus, by setting the maximum depths of these LCUs to be 0, the encoding complexity can be saved with little quality and coding efficiency loss. Finally, since the bit allocation information of the current frame can only be obtained after encoding, we use the bit allocation map of its previous frame as the map for the current frame. This assumption is reasonable because there are few scene changes in video conferencing, and the experimental results also verify the effectiveness.

2.2. Subjective weight map

The subjective weight maps aim to protect the perceptual-driven video quality. Here, we adopted the method in [13] to generate the subjective weight maps, because it is very fast. [13] can predict the saliency value of each pixel in frames. Let $\{p_q\}_{q=1}^{Q}$ denote the saliency values of all Q pixels in j-th LCU, then the subjective weight of j-th LCU is $W_s(j) = \sum_{q=1}^{Q} p_q/Q$. The subjective weight maps can highlight the regions attracting people's attention most when they are watching videos. Intuitively, we hope to preserve the video quality of regions with large subjective weights.

However, the subjective weight has lower relevance with optimal depth. For example, many LCUs have large subjec-

Fig. 4. Illustration of complexity control algorithms for different target levels.



(a) *Fourpeople*

Fig. 5. Relationship between frames and sum encoding time.

tive weights but their optimal depths are pretty small. Relying only on subjective weights to determine the maximum depths of LCUs may incur objective quality and coding efficiency loss. By comparison, the objective weights can protect the objective quality, but may impair the perceived quality. Thus, in order to keep a balance between the objective and perceived quality, we take both the objective and subjective weights into consideration when deciding the maximum depths of LCUs in Section 2.3.

2.3. Complexity control formulation

In our approach, complexity is controlled by adjusting the number of LCUs with different constrained maximum depths. Based on proportions in Fig. 2, the complexity control formulation is established as

$$\min_{\{N_i\}_{i=0}^{3}} \left| \frac{1}{J} \sum_{i=0}^{3} P_i N_i - T_c \right| \quad \text{s.t.} \quad \sum_{i=0}^{3} N_i = J, \tag{1}$$

where N_i is the number of LCUs with maximum depth being i, and J is the total number of LCUs in each frame. P_i is the complexity proportion occupied by maximum depth being i. T_c is the target complexity. We divide T_c into three levels: high (T_c is from 1.00 to 0.65), medium (from 0.65 to 0.45), and low (less than 0.45). Based on the T_c level, (1) is solved using different ways.

High. The target complexity is so high that there is no need to reduce the maximum depths to 0 and 1. N_0 and N_1 in (1) are set to 0, and then (1) can be turned to

$$\min_{\{N_2,N_3\}} \left| \frac{1}{J} \sum_{i=2}^{3} P_i N_i - T_c \right| \quad \text{s.t.} \quad \sum_{i=2}^{3} N_i = J. \tag{2}$$

Medium. The maximum depths of LCUs can be selected from {0,1,2,3}. However, there are some constraints on the selections. The LCUs with bits ordered less than 20% should select 0, and LCUs with bits ordered from 80% to 100% should select 3 as their maximum depths. The other LCUs can select between 1 and 2. As we have explained in Section 2.1, setting the maximum depths of LCUs whose bits order is less than 20% to 0 has little effect on the coding efficiency and objective quality:

$$\min_{\{N_1,N_2\}} \left| \frac{1}{J - N_0 - N_3} \sum_{i=1}^{2} P_i N_i - T_c \right| \text{s.t.} \sum_{i=1}^{2} N_i = J - N_0 - N_3, \tag{3}$$

where N_0 and N_3 are both $J \times 20\%$.

Low. The target complexity is so low that most LCUs can only select their maximum depths between 0 and 1. N_2 is set to 0. However, in order to guarantee the video quality, like the Medium, the LCUs with bits ordered from 80% to 100% are given optimal depths as 3. Then, (1) can be turned to

$$\min_{\{N_0,N_1\}} \left| \frac{1}{J - N_3} \sum_{i=0}^{1} P_i N_i - T_c \right| \quad \text{s.t.} \sum_{i=0}^{1} N_i = J - N_3. \tag{4}$$

For each complexity level, following the above formulations, it is easy to calculate and obtain $\{N_i\}_{i=0}^{3}$. Then, in each frame, after $\{N_i\}_{i=0}^{3}$ is obtained, the j-th LCU can get its maximum depth based on its objective weight $W_o(j)$ and subjective weight $W_s(j)$. Before that, we need to sort the objective and subjective weights of all LCUs in a frame. Let $\{\lambda_p\}_{p=0}^{2}$ be the thresholds of LCU numbers with limited depths, $\lambda_p = \sum_{i=0}^{p} N_i$. Then, the thresholds of objective and subjective weights corresponding to λ_p are denoted by $\mathcal{O}(\lambda_p)$ and $\mathcal{S}(\lambda_p)$, respectively. Table 2 presents the overall algorithm in determining the maximum depth D_j for the j-th LCU in a frame.

The target complexity for current frame can be updated based on the encoding time of its previous frames, to further increase the control accuracy. Here, the target complexity of the first M frames is set to 1.00, and their encoding time can be used to predict the total encoding time of the sequence:

$$E_f = \frac{F}{M} E_M, \tag{5}$$

where E_M is the encoding time of the first M frames and F is the frame number of sequence. As can be seen from Fig. 5, the encoding time is directly proportional to the frame number. Thus, it is reasonable to predict total encoding time using (5). The target encoding time per frame t_{frame} is obtained

$$t_{frame} = \frac{E_f}{F} \times T_c. \tag{6}$$

From the (M+1)-th frame on, the average encoding time per frame is denoted by t_{actual}. T_c is updated as follows: if $t_{actual} < \alpha t_{frame}$, T_c of current frame is updated to $T_c + a$; if $t_{actual} > \beta t_{frame}$, T_c is updated to $T_c - b$. Here, we empirically set α and β to 0.95 and 1.05, set a and b to 0.05, and M to 48, i.e., the first 12 GOPs.

Table 2. The Overall Algorithm of Our Approach

– **Input:** The target complexity T_c.
– **Output:**The maximum depth D_j for j-th LCU in each frame.

- Initialize F to the number of frames to be encoded.
- Initialize J to the number of LCUs in a frame.
- Initialize M to the number of frames without complexity control.
- **For** $k = 1$, $k \leq M$, k++
  Calculate t_{frame} using (6).
  End
- **For** $k = M + 1$, $k < F$, k++
  1 Calculate $\{N_i\}_{i=0}^{3}$ by (2), (3), and (4), based on the target complexity level. set $\lambda_p = \sum_{i=0}^{p} N_i$.
  2 **For** $j = 0$, $j < J$, j++
  Calculate $W_o(j)$ and $W_s(j)$ for the j-th LCU.
  If $W_o(j) < \mathcal{O}(\lambda_0)$, D_j=0
  Else If $W_o(j) < \mathcal{O}(\lambda_1)$&&$W_s(j) < \mathcal{S}(\lambda_1)$, D_j=1
  Else If $W_o(j) > \mathcal{O}(\lambda_2)$, D_j=3
  Else D_j=2
  End
  3 Update T_c.

End

Table 3. Test Sequences

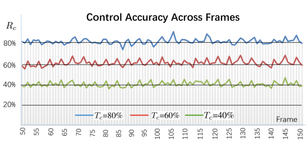
Sequences	Resolution	Frames
Johnny	1280 × 720	600 @60fps
KristenAndSara	1280 × 720	600 @60fps
Fourpeople	1280 × 720	600 @60fps
Vidyo_1	1280 × 720	600 @60fps
Vidyo_3	1280 × 720	600 @60fps
Vidyo_4	1280 × 720	600 @60fps

Table 4. Complexity control performance comparison between our and comparing approaches

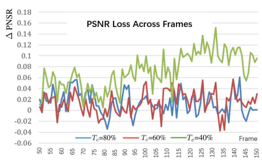
T_c=60%	Our approach			Comparing [3]		
	R_c(%)	BD-PSNR	BD-rate	R_c(%)	BD-PSNR	BD-rate(%)
Johnny	58.80	0.00 dB	0.00	65.18	0.00 dB	0.10
KristenAndSara	61.41	-0.01 dB	0.01	64.54	-0.03dB	0.11
Fourpeople	60.35	-0.01 dB	0.46	67.78	-0.03 dB	0.66
Vidyo1	57.39	-0.02 dB	0.26	67.24	-0.02 dB	0.24
Vidyo3	58.12	-0.06 dB	1.02	66.72	-0.07 dB	0.89
Vidyo4	62.21	-0.02 dB	0.30	63.32	-0.04 dB	0.87
Average	**59.71**	**-0.02 dB**	**0.34**	65.80	-0.03 dB	0.48
	Our approach			Comparing [3]		
T_c=40%	R_c(%)	BD-PSNR	BD-rate	R_c(%)	BD-PSNR	BD-rate(%)
Johnny	42.33	-0.06 dB	2.32	35.22	-0.21 dB	5.72
KristenAndSara	40.10	-0.07 dB	3.25	33.57	-0.53dB	9.21
Fourpeople	41.23	-0.24 dB	6.50	42.83	-0.35 dB	10.78
Vidyo1	37.36	-0.06 dB	1.93	27.09	-0.23 dB	9.89
Vidyo3	38.37	-0.11 dB	3.67	31.39	-0.28 dB	10.32
Vidyo4	40.16	-0.11 dB	3.88	33.89	-0.21 dB	5.87
Average	**39.92**	**-0.11 dB**	**3.59**	34.00	-0.30 dB	8.63

Table 5. Δ P-PSNR results of our approach

Δ P-PSNR (dB)	Johnny	KAndS	Fourpeople	Vidyo1	Vidyo3	Vidyo4
T_c=80	0.00	0.00	0.01	0.01	0.02	0.01
T_c=60	0.00	0.02	0.05	0.06	0.06	0.04
T_c=40	0.02	0.06	0.13	0.14	0.13	0.10



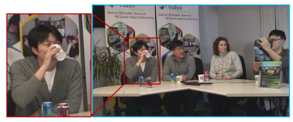
(a)



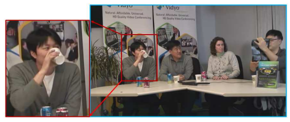
(b)

Fig. 6. (a) shows the running complexity across frames with 80%, 60%, and 40% targets of *Fourpeople*. (b) shows the corresponding Δ PSNR across frames. Here, Δ PSNR refers to the PSNR loss caused by complexity reduction.

(a) Original picture



(b) Picture with 80% target



(c) Picture with 60% target

(d) Picture with 40% target

Fig. 7. The 85-th frames of *Fourpeople* with different complexity reductions.

3. EXPERIMENTAL RESULTS

Experiments were done on HM 16.0 and the test videos are video conferencing sequences selected from HEVC standard test sequences, shown in Table 3. The test condition was chosen according to [14], and lowdelay_P_main configuration is used, because video conferencing requires low latency.

Table 4 shows the results of control accuracy, BD-rate and BD-PSNR for 60% and 40% target complexities of our and comparing approach [3]. R_c is the actual running complexity. We can see that our approach outperforms [3] in making R_c much closer to the target complexity T_c with small bias. Meanwhile, our approach keeps the objective quality well. For *Johnny* @60% there is even no PSNR loss. Fig.6-(a) plot the running complexity change with frames for different targets and we can see that the controlling process is basically steady with a little fluctuation. Fig.6-(b) shows the PSNR loss across frames and it is obvious that for 80% and 60% targets, there is little PSNR loss. Interestingly, many frames have negative PSNR loss indicating that we improve the PSNR while reducing the complexity.

We calculate perceptual driven quality P-PSNR following [15]. Table 5 shows the perceptual driven quality loss Δ P-PSNR caused by complexity reduction in our approach. In this figure, the loss is negligible until 40%. Fig.7 show the same picture with different complexity reductions, and we cannot feel obvious quality distortion among them, especially among 80%, 60% and original picture.

4. CONCLUSION

In this paper, we propose an HEVC complexity control approach for video conferencing encoding. We integrate the video quality protection problem within the control process. Specifically, we propose two weight maps to keep the objective and perceptual-driven video quality and these maps are fully incorporated in the controlling algorithm. Thus, our approach can simultaneously ensure the control accuracy and preserve video quality, including objective and perceptual-driven. The experimental results verifies the effectiveness of our approach comparing to other state-of-the-art approach.

5. REFERENCES

[1] Guilherme Correa, Pedro Assuncao, Luis A da Silva Cruz, and Luciano Agostini, "Encoding time control system for hevc based on rate-distortion-complexity analysis," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2015, pp. 1114–1117.

[2] Xin Deng, Mai Xu, Lai Jiang, Xiaoyan Sun, and Zulin Wang, "Subjective-driven complexity control approach for HEVC," *IEEE TCSVT*, vol. 26, no. 1, pp. 91–106, 2016.

[3] Amaya Jiménez-Moreno, Eduardo Martínez-Enríquez, and Fernando Díaz-de María, "Complexity control based on a fast coding unit decision method in the hevc video coding standard," *IEEE Transactions on Multimedia*, vol. 18, no. 4, pp. 563–575, 2016.

[4] Jiunn-Tsair Fang, Zong-Yi Chen, Chang-Rui Lai, and Pao-Chi Chang, "Computational complexity allocation and control for inter-coding of high efficiency video coding with fast coding unit split decision," *Journal of Visual Communication and Image Representation*, vol. 40, pp. 34–41, 2016.

[5] Tiesong Zhao, Zhou Wang, and Sam Kwong, "Flexible mode selection and complexity allocation in high efficiency video coding," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 6, pp. 1135–1144, 2013.

[6] Guilherme Corrêa, Pedro Assuncao, Luciano Agostini, and Luis A da Silva Cruz, "Complexity control of high efficiency video encoders for power-constrained devices," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 4, pp. 1866–1874, 2011.

[7] Guilherme Correa, Pedro Assuncao, Luciano Agostini, and Luis A Da Silva Cruz, "Coding tree depth estimation for complexity reduction of hevc," in *Data Compression Conference (DCC), 2013*. IEEE, 2013, pp. 43–52.

[8] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas 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, 2012.

[9] Jinlei Zhang, Bin Li, and Houqiang Li, "An efficient fast mode decision method for inter prediction in hevc," 2015.

[10] Ivan Zupancic, Saverio G Blasi, and Ebroul Izquierdo, "Multiple early termination for fast hevc coding of uhd content," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 1419–1423.

[11] Hyo-Song Kim and Rae-Hong Park, "Fast cu partitioning algorithm for hevc using an online-learning-based bayesian decision rule," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 130–138, 2016.

[12] Jian Xiong, Hongliang Li, Fanman Meng, Qingbo Wu, and King Ngi Ngan, "Fast hevc inter cu decision based on latent sad estimation," *IEEE Transactions on Multimedia*, vol. 17, no. 12, pp. 2147–2159, 2015.

[13] Chenlei Guo and Liming Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE TIP*, vol. 19, no. 1, pp. 185–198, 2010.

[14] Frank Bossen, "Common test conditions and software reference configurations," *JCTVC-F900*, 2011.

[15] Zhou Wang and Qiang Li, "Information content weighting for perceptual image quality assessment," *TIP*, vol. 20, no. 5, pp. 1185–1198, 2011.