IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 18, 2016, accepted September 19, 2016, date of publication September 22, 2016, date of current version November 18, 2016.

Digital Object Identifier 10.1109/ACCESS.2016.2612691

Hierarchical Complexity Control of HEVC for Live Video Encoding

XIN DENG[1], (Student Member, IEEE), MAI XU[2], (Member, IEEE), AND CHEN LI[2]
[1]Department of Electrical and Electronic Engineering, Imperial College London, London, SW7 2AZ, U.K.
[2]School of Electronic and Information Engineering, Beihang University, Beijing 100191, China

Corresponding author: M. Xu (maixu@buaa.edu.cn)

This work was supported in part by NSFC under Grant 61537037 and in part by the CSC Imperial Scholarship.

ABSTRACT As the latest video coding standard, High Efficiency Video Coding (HEVC) tremendously improves the encoding efficiency compared with the preceding H.264/AVC standard, but at the cost of higher encoding complexity. This huge encoding complexity makes the implementation of HEVC intractable on live videos. For coping with this problem, two major challenges need to be solved: 1) How to accurately reduce the encoding complexity to achieve the target complexity? and 2) How to maintain the video quality after encoding complexity reduction? To solve these two challenges, we propose, in this paper, a hierarchical complexity control approach of HEVC. For the first goal, the complexity control is implemented in two levels to assure the control accuracy. In the largest coding unit (LCU) level, we adjust the maximum depths of LCUs in a frame to reduce the encoding complexity to the target. Since each frame has numerous LCUs, and each LCU can choose its maximum depth from one of the four maximum depths, the large degree of freedom contributes to the high control accuracy. However, there may be still some errors. These errors can be compensated in the frame level by a proposed frame level complexity control algorithm. For the second goal, one objective weight map and one subjective weight map are proposed to use in the process of complexity control to keep the objective and subjective video quality simultaneously. Finally, The experimental results show that our approach outperforms other state-of-the-art approaches, in terms of both control accuracy and video quality.

INDEX TERMS High Efficiency Video Coding (HEVC), complexity control.

I. INTRODUCTION

Recently, with the increased demand of immersive video watching experience, watching live videos has been an important part of people's lives. People are more interested in live videos than recorded videos, because the former can bring them real-time experience and feeling. People watch live football matches, use video chat to communicate with their parents and friends in the distance, and the corporations use video conferencing to make a meeting with staff from different places. For more immersive experience, the resolutions of live videos are expected to be high, usually as 1080p or 4K. According to Cisco Visual Networking Index (VNI), by 2019, more than 30% of connected flat-panel TV sets will be 4K and the traffic of 1080p and 4K videos will make up 90.9% of the global video traffic.

The high resolution and huge data of live videos can lead to some problems. As we know, thousands of live videos need to be broadcasted on TV or Internet everyday. However, with the high resolutions, the broadcasting of live videos is facing two main challenges. Firstly, the transmission bandwidth is too limited to transmit the live videos with such a high bit rate. Secondly, the live videos require low-delay encoding, which means that each frame must be encoded in a very short time. Fortunately, the release of the High Efficiency Video Coding (HEVC), also called H.265, copes with the first challenge well. According to [1], HEVC has successfully saved 59% bit-rate over the previous H.264/ MPEG-4 AVC with similar subjective quality. However, this high coding efficiency relies on many time-consuming coding techniques, which makes the encoding complexity of HEVC quite enormous. In other words, the encoding of each frame usually consumes a lot of time by HEVC, making the solving of the second challenge quite intractable.

To solve this problem, some works have been done to control encoding complexity of HEVC. In 2013, Correa *et al.* [2] proposed to control the encoding complexity of HEVC by

calculating the number of constrained frames (i.e., frames with complexity reduction) according to the target complexity. However, the control range in [2] is quite limited, only from 100% to 60%. In 2015, we proposed in [3] a novel complexity control approach for HEVC to overcome the disadvantages of [2], which can achieve complexity control with a quite large range, i.e., from 100% to 20%. However, this approach has two flaws. Firstly, the complexity control is only done on a single level and all the control parameters are fixed during the encoding process, and thus its control accuracy is not satisfactory. Secondly, the splitting of many largest coding units (LCUs) is skipped according to their subjective weights, and the improper early-skip leads to some unnecessary bit-rate increase and objective quality loss. To solve these problems, we propose in this paper a new approach based on [3], called Hierarchical Complexity Control (HCC) for HEVC. In this approach, encoding complexity is controlled in two levels, i.e., frame level and LCU level, and thus the control accuracy can be significantly improved. Another technique we use to increase the control accuracy is the classified relationship models, which is introduced in detail in Section III-A. As for the objective quality loss, we employ the new proposed objective weight maps to protect the objective video quality, which is explained in Section IV-A. Figure 1 shows an application example of our approach. When a basketball match is playing and need to be encoded, we can adjust the complexity allocation according to the importance of the different regions, in order to decrease the encoding complexity to the target and keep the video quality.

In this paper, we propose a hierarchical complexity control approach for live videos encoding on the HEVC platform. Compared with our previous work [3], this new approach overcomes many drawbacks of the old work by introducing some new techniques. Specifically, the main technical contributions of our HCC approach in this paper are summarized as follows,

- We investigate and establish a classified relationship between LCU maximum depth and encoding complexity to increase the control accuracy.
- We achieve complexity control in two levels: LCU level to achieve complexity control of each frame, and further frame level to achieve complexity control of the whole video.
- We employ both objective and subjective weight maps to simultaneously keep the objective and subjective video quality.

The whole paper is organized as follows. In Section II, we briefly review the related works on the encoding complexity reduction and control of HEVC. Then, in Section III-A, two classified relationships about encoding complexity, video quality distortion, and LCU maximum depth are trained and modelled. In Section IV, details about the proposed HCC approach are discussed. The experimental results are shown in Section V to verify the effectiveness of our HCC approach from different aspects. Finally, Section VI concludes this paper.

II. RELATED WORKS

The works with respect to the encoding complexity of HEVC have been extensively studied, since the issue of HEVC in January, 2013. Among these works, there are two main research directions, i.e., complexity reduction and complexity control.

For HEVC complexity reduction, extensive studies pay attention to the new coding tree unit (CTU) partitioning scheme, which leads to huge encoding complexity. Among these studies, [2] and [4]–[9] devoted to find ways to reduce the encoding complexity on exhaustively searching for optimal CU sizes in the block partitioning process. Specifically, Leng *et al.* [10] proposed an early CU depth prediction approach at frame level. The basic idea of this approach is to skip some CU depths which are rarely used in the previous frames, thus simplifying the RDO search process to save encoding complexity. Literature [2], [4] developed similar approaches at CU level, with the central idea to narrow the current CU depth search range, by virtue of the depth information of adjacent CUs. Different from this kind of approaches, Shen *et al.* [7] developed a fast CU size selection

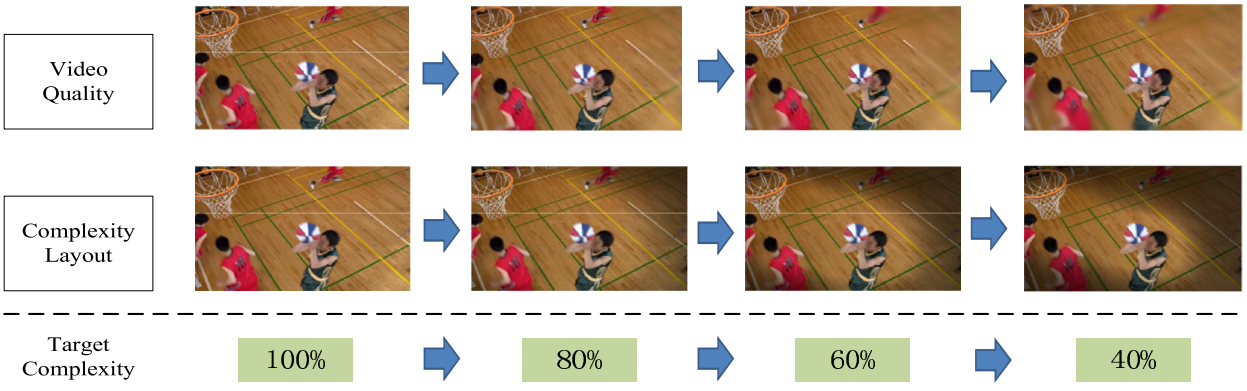


FIGURE 1. One application of our approach. The complexity layout changes according to the target complexity, of which the dark areas are allocated with less complexity resources (i.e., smaller maximum depths).

approach for complexity reduction in HEVC, based on a Bayesian decision rule. It employs some computational-friendly features to make a precise and fast selection on CU sizes, by minimizing the Bayesian risk of RD cost. Xiong *et al.* [11] proposed a fast pyramid motion divergence based CU selection method and used k nearest neighboring like method to determine the CU splitings. Besides, some early prediction unit (PU) and transform unit (TU) size decision methods were proposed in [12]–[15] to speed up the PU and TU size selection process. Specifically, Yoo and Suh [14] checked the code block flag (CBF) and RD cost of the current PU to terminate the prediction process of the next PU for complexity reduction. Except for CU, PU, and TU size decisions, there are still other components in HEVC affecting the encoding complexity, such as in-loop filtering, and multi-directional intra predictions. From these aspects, [16]–[18] provided several methods to reduce the encoding complexity of HEVC. For example, Zhang and Ma [16] proposed a fast intra mode decision for HEVC encoder. It introduces a rough mode search scheme to selectively check some potential modes instead of all candidates, thus simplifying the intra prediction process for complexity reduction. Apart from these approaches, most recently, machine learning and data mining techniques have also been used for reducing the encoding complexity of HEVC [19], [20].

Compared with HEVC complexity reduction, the works for HEVC complexity control is not too much. As for the existing researches, there are three main thoughts employed for HEVC complexity control: complexity allocation, parameter control, and early terminating. The complexity allocation aims to achieve a target complexity by reasonably allocating the complexity resources. The parameter control is to find out and configure several encoding parameters to make complexity controllable. The early terminating leverages various early termination algorithms to control complexity. The existing approaches on HEVC complexity control may use one or more of the three thoughts. Specifically, Zhao *et al.* [21] employed a user-defined complexity factor to control the number of coding modes, such that the encoding complexity of HEVC can be adjusted. However, it cannot accurately control the encoding complexity. Correa *et al.* [2] achieved HEVC complexity control by means of early terminating the coding tree splitting process. In this approach, frames are divided into two categories: unconstrained frames (Fu) and constrained frames (Fc). The coding unit (CU) depth in Fc is early determined with the same as CU depth in its previous Fu frame. Through adjusting the number of Fc, in which the CU splitting process is early terminated, the target encoding complexity can be achieved. Jimenez Moreno *et al.* [22] have proposed a complexity control approach for HEVC which is based on a set of early termination conditions. However, the control accuracy is fluctuated among the test sequences, with the highest control error more than 6%, and in the same sequence, the control accuracy is fluctuated sharply across different frames. Most recently, we proposed an approach [3] to control the encoding complexity of HEVC, which can achieve complexity control of HEVC from 100% to 20%. However, the control accuracy of this work is not very high. Besides, a lot of bit-rates are wasted in [3] due to improper early-skip of LCU splitting. Thus, further improvements can still be achieved in the field of HEVC encoding complexity control.

In this paper, we propose a new HCC approach for HEVC to control the encoding complexity with large range and high accuracy. Different from [3], we explore the relationship between encoding complexity and LCU maximum depth more accurately and classify it into two categories according to the frame order, so that the depth-complexity relationship modelled in this work are more accurate. Moreover, due to the new introduced frame level complexity control technique, the control parameters can be adaptively adjusted during the encoding process, which significantly contributes to the control accuracy. More importantly, both objective and subjective weight maps are taken into consideration when decreasing the encoding complexity of LCUs. This way, both the objective and subjective video quality are kindly preserved in this approach.

III. RELATIONSHIPS MODELLING

A. CLASSIFIED DEPTH-COMPLEXITY RELATIONSHIP

In HEVC, one important parameter for LCU splitting is the allowed LCU maximum splitting depth. We only call it LCU maximum depth in this paper for convenience. This parameter is important because it determines the smallest coding unit (CU) each LCU can be split into, and it has been verified as an effective parameter to adjust the encoding complexity of HEVC in [3]. Since we focus on the complexity control for live videos, the Low-delay P main configuration is used with the default hierarchical coding structure for model training. The hierarchical coding structure is a default coding structure in HEVC and Figure 2 shows this structure with the size of group of pictures (GOP) default as 4. We can see that in a GOP, frames can be divided into different levels according to their quantization parameter (QP) values. As demonstrated

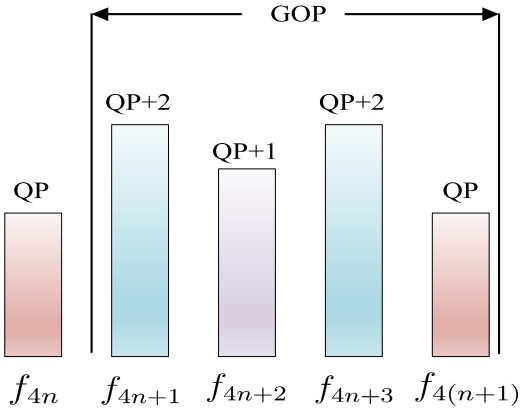


FIGURE 2. Hierarchical coding structure for the Low-delay P configuration [23]. f_{4n}, f_{4n+1}, f_{4n+2}, f_{4n+3}, and $f_{4(n+1)}$ are the frame order.

later, this hierarchical coding structure has effects on the relationships between LCU maximum depth and encoding complexity.

For analyzing the relationship between maximum depth M_d and encoding complexity E_c, we trained three video sequences at four different maximum depths (i.e., 3, 2, 1, 0) on HM 16.0 software, with Low-delay P main setting. The training video sequences were selected from the standard HEVC test sequence database, as shown in Table 1. Note that these training sequences are different from the test video sequences in Section V. We used a 64-bit Windows PC with Inter(R) Core(TM) i7-4770 processor @3.40 GHz to carry out the training process. The training sequences were encoded with four different QPs, i.e., 22, 27, 32, and 37. Given each QP, the LCU maximum depths were set as 3, 2, 1, and 0, and the encoding time of each LCU was recorded. Here, the encoding time with maximum depth being 3 was regarded as the reference full time, which was utilized to normalize the encoding time of other maximum depths, i.e., 2, 1, 0. Assume that $E_c(M_d)$, $M_d \in \{3, 2, 1, 0\}$, denotes the LCU encoding time at maximum depth M_d. Then, $E_c(M_d)$ was normalized using the reference time $E_c(M_d = 3)$:

$$\widetilde{E}_c(M_d) = \frac{E_c(M_d)}{E_c(M_d = 3)}, \quad M_d \in \{3, 2, 1, 0\}. \tag{1}$$

Then, the normalized encoding time of all LCUs is averaged for each frame. Figure 3 shows the encoding complexity of different frames with different maximum depths. We find that the encoding complexity is different among four frames in GOP. Interestingly, this difference is consistent with the hierarchical coding structure in Figure 2. The 4-th frame has the smallest QP and correspondingly, its encoding complexity is the highest in the GOP. It is probably because more bits are assigned to the 4-th frame in the GOP, which may lead to the increase of encoding complexity. Thus, for more accurate depth-complexity model, the 4-th frame should be separated from the other three frames. Since the difference among the first three frames is not that much, they can be analysed together.

Figure 4 shows the specific analyzing procedure for the 4-th frame and the first 3 frames. For each sequence, the encoding complexity $\widetilde{E}_c(M_d)$ is first averaged among all frames of the training sequence, and then averaged among four QPs. Assume that $E_c'(M_d)$ and $E_c''(M_d)$ are the relationships between maximum depth M_d and encoding complexity for the 4-th frames and other three frames in GOPs, respectively. The specific fitting curves about $E_c'(M_d)$ and $E_c''(M_d)$ for all three sequences are shown in Figure 5. From this figure, we can find that there is little difference between the three training sequences for both $E_c'(M_d)$ and $E_c''(M_d)$. Thus, we averaged the encoding complexity for all three sequences to model the final relationships, shown as the red curves in Figure 5. We can see that the R-square values of the red curves are 0.9997 and 1 for $E_c'(M_d)$ and $E_c''(M_d)$, verifying the accuracy of the relationships. Finally, the encoding complexity at different M_d (i.e., 3, 2, 1, 0) was obtained, as presented in Table 2.

B. CLASSIFIED DEPTH-DISTORTION RELATIONSHIP

We have established the relationship between LCU maximum depth and encoding complexity. From now on, we concentrate on exploring the influence of LCU maximum depth on the video distortion. It is evident that the reduction of maximum depth usually results in more video quality distortion. Here, the training sequences we used are the same as those in the former subsection, as presented in Table 1. Note that in this paper, the video quality distortion is measured by mean square error (MSE). The MSEs at different maximum depths (i.e., 3, 2, 1, 0) were recorded for each frame, defined as MSE(M_d). Next, the normalized distortion can be computed as follows,

$$\widetilde{D}(M_d) = \frac{\mathrm{MSE}(M_d) - \mathrm{MSE}(M_d = 3)}{\mathrm{MSE}(M_d = 3)}, \quad M_d \in \{3, 2, 1, 0\}. \tag{2}$$

Similar to the training procedure in the former subsection, $\widetilde{D}(M_d)$ was first averaged among all all frames and four QPs in each sequence, and further averaged among

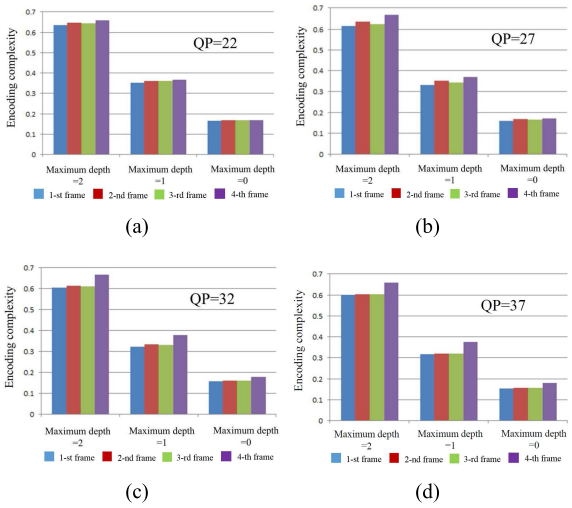


FIGURE 3. The encoding complexity difference across the four frames in GOPs with different QPs. The results are from *KristenAndSara* sequence.

TABLE 1. Training sequences and configuration.

Sequences	Class	Resolution	Training frames	Configuration	QP	LCU size
Cactus	B	1920×1080	500@50fps	Low-delay P	22, 27, 32, 37	64×64
BQMall	C	832×480	600@60fps	Low-delay P	22, 27, 32, 37	64×64
KristenAndSara	E	1280×720	600@60fps	Low-delay P	22, 27, 32, 37	64×64

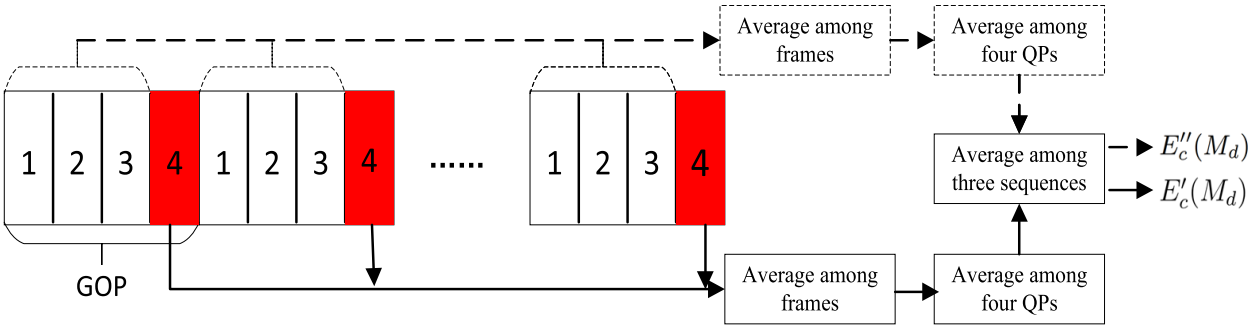
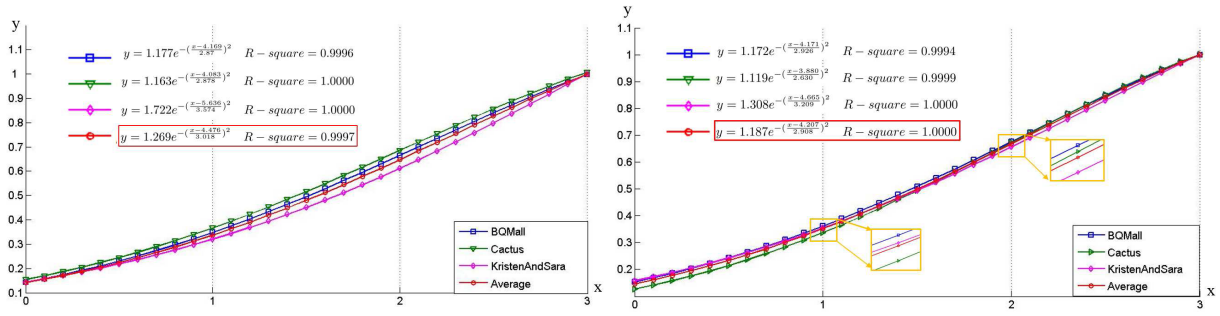


FIGURE 4. Complexity analyzing procedure: solid line is the procedure for the 4-th frame; dashed line is the procedure for other 3 frames in each GOP.



(a) (b)

FIGURE 5. Fitting curves between maximum depth M_d and encoding complexity $E_c'(M_d)$ for the 4-th frames, and $E_c''(M_d)$ for the other three frames. Note that the horizontal axis x stands for maximum depth M_d, and the vertical axis y stands for the encoding complexity. All the curves are obtained by curve fitting using Gaussian models, with the R-square values showing in the figures as well. The red curve with its corresponding function is the final relationship adopted in this paper. (a) $E_c'(M_d)$. (b) $E_c''(M_d)$.

TABLE 2. Relationships between M_d and encoding complexity E_c.

$E_c(M_d)$	M_d=3	M_d=2	M_d=1	M_d=0
$E_c'(M_d)$	1.000	0.666	0.355	0.144
$E_c''(M_d)$	1.000	0.643	0.345	0.134

TABLE 3. Relationships between M_d and distortion D.

$D(M_d)$	M_d=3	M_d=2	M_d=1	M_d=0
$D'(M_d)$	0.000	0.016	0.051	0.132
$D''(M_d)$	0.000	0.018	0.076	0.183

all three training sequences. The results are shown in Table 3. $D'(M_d)$ and $D''(M_d)$ are the distortion for the 4-th frame and other three frames respectively. From this table, we can observe that the video distortion increases drastically alongside the reduction of maximum depth.

IV. COMPLEXITY CONTROL

In this paper, complexity control is done in two levels. In the LCU level, the maximum depth of LCUs in each frame is adjusted to make the frame encoding complexity reach the frame target. In the frame level, the target complexity for each frame is adjusted to make the whole video encoding complexity reach the video target. But before introducing the complexity control algorithms, we first elaborate the two weight maps which are of great importance in solving the complexity control problem later.

A. OBJECTIVE WEIGHT MAP

The objective weight map is used to preserve objective quality and coding efficiency in the complexity control process. 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 allocation well. Here, the optimal depth refers to the depth calculated through the RDO process. As we can see in Figure 6, the LCUs allocated with more bits tend to have larger optimal depths, and LCUs with smaller bits have great chance being not split, i.e., the optimal depth is 0.

In order to accurately analyse the dependency between the optimal depth and bit allocation, two conditional probability $\mathcal{P}(D|B)$ and $\mathcal{P}(B|D)$ are adopted, where D denotes the event that the optimal depth is 0, 1, 2 or 3, and B is the bit allocation 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 4 shows the average results of $\mathcal{P}(D|B)$ for three different videos. We can see that the lower the bit order is, the more probability the optimal depth can have a smaller value.

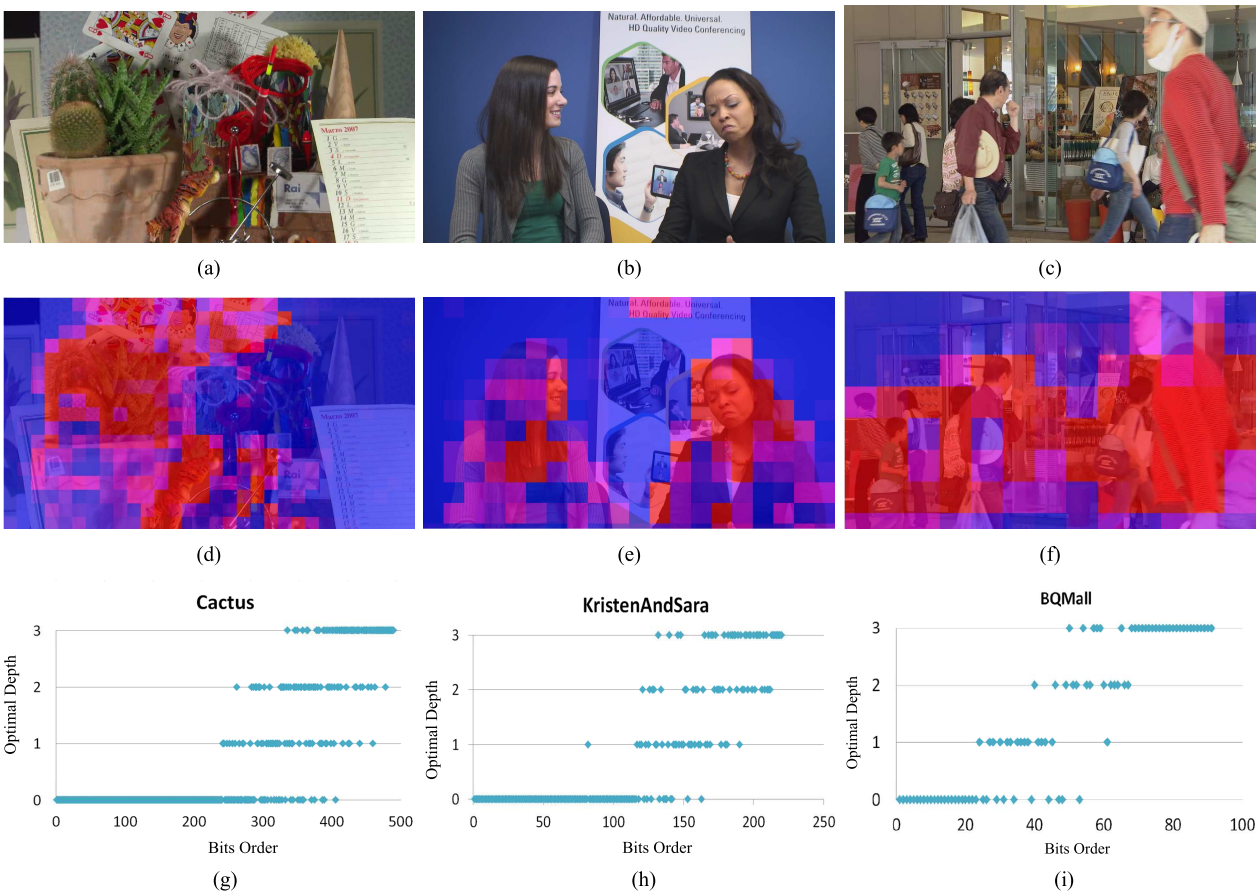


FIGURE 6. The pictures in the first row are original frames. The pictures in the second row are bit allocation maps, i.e., objective weight maps. (g), (h) and (i) show relationships between optimal depth and bit allocation, with the horizontal axis being the ascending order of bits allocated to each LCU.

TABLE 4. Average results of the probability $\mathcal{P}(D|B)$.

	$\mathcal{P}(D\|B)$	D=0	D=1	D=2	D=3
Cactus	$B < 20$	**99.80**	0.20	0.00	0.00
	$20 \leq B < 40$	**97.54**	2.45	0.02	0.00
	$40 \leq B < 60$	**57.33**	37.07	5.15	0.45
	$60 \leq B < 80$	12.60	**57.96**	25.06	4.37
	$80 \leq B < 100$	3.06	40.52	**42.49**	13.93
	$\mathcal{P}(D\|B)$	D=0	D=1	D=2	D=3
KristenAndSara	$B < 20$	**99.81**	0.19	0.00	0.00
	$20 \leq B < 40$	**99.66**	0.34	0.00	0.00
	$40 \leq B < 60$	**97.65**	2.34	0.01	0.00
	$60 \leq B < 80$	**73.56**	24.25	2.12	0.08
	$80 \leq B < 100$	20.18	**57.22**	19.57	3.03
	$\mathcal{P}(D\|B)$	D=0	D=1	D=2	D=3
BQMall	$B < 20$	**99.89**	0.11	0.00	0.00
	$20 \leq B < 40$	**97.52**	2.45	0.03	0.00
	$40 \leq B < 60$	**60.13**	35.25	4.25	0.38
	$60 \leq B < 80$	12.22	**62.48**	21.53	3.76
	$80 \leq B < 100$	0.21	28.84	**47.34**	23.61

Specifically, when B is less than 20%, the probability of the event that LCUs do not split (i.e., $D = 0$) is 99.80 for *Cactus*, 99.81 for *KristenAndSara*, and 99.89 for *BQMall*. In other words, by setting the maximum depths of these LCUs to be 0, the encoding complexity can be saved with little quality and coding efficiency loss. In addition to $\mathcal{P}(D|B)$, we also

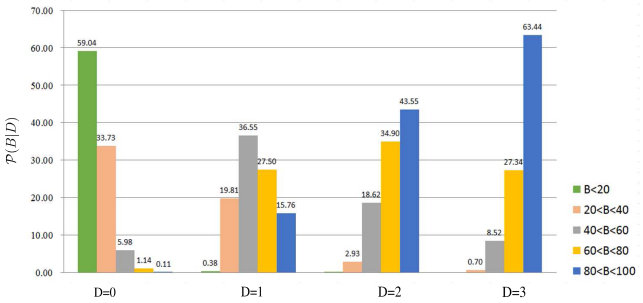
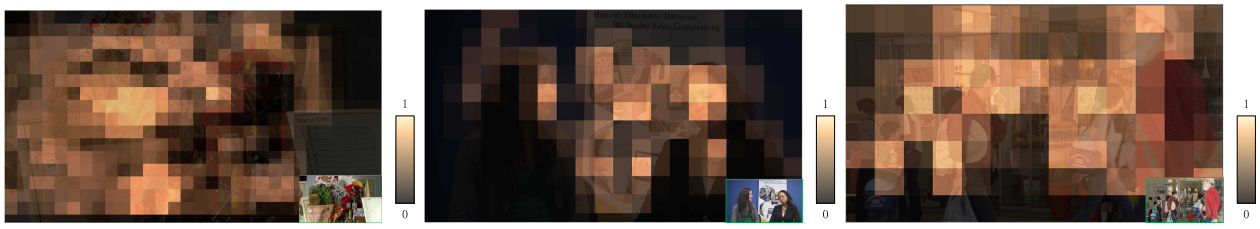


FIGURE 7. $\mathcal{P}(B|D)$ of *BQMall*, where B is divided into five categories: $B < 20$, $20 < B < 40$, $40 < B < 60$, $60 < B < 80$, and $80 < B < 100$.

calculate $\mathcal{P}(B|D)$ for explain the dependency between the optimal depth and bit allocation more completely. Figure 7 shows the results. $\mathcal{P}(B < 20|D = 0)$ indicates the probability of the event that the bit order is less than 20% when the LCU optimal depth is 0. We can see from Figure 7 that when $D = 0$, $\mathcal{P}(B < 20|D = 0)$ is the highest, and when $D = 3$, $\mathcal{P}(80 < B < 100|D = 3)$ is the highest. We can get the conclusion that compared with smaller D, larger D has greater chance to be allocated with more bits.



(a) (b) (c)

FIGURE 8. Examples of subjective weight maps. For the original pictures, refer to the first row of Figure 6. (a) *Cactus*: the 60-th frame. (b) *KristenAndSara*: the 104-th frame. (c) *BQMall*: the 48-th frame.

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. Finally, let B_j be the bits allocated to the j-th LCU, we can get the normalised objective weight of the j-th LCU O_j as:

$$O_j = \frac{B_j}{B_{max}}, \tag{3}$$

where B_{max} is the largest bits among LCUs in a frame. O_j will be used in Section IV-C for coping with the M_d allocation problem.

B. SUBJECTIVE WEIGHT MAP

Different from objective weight map, subjective weight map is employed for optimizing the subjective quality. Here, the PQFT algorithm [24] is used to yield the saliency value of each LCU as the element in the map. The subjective weight maps can highlight regions attracting people's attention most when they are watching videos. Fig. 8-(a) and (b) show the subjective weight maps. The normalised subjective weight of j-th LCU can be calculated as:

$$S_j = \frac{V_j}{V_{max}}, \tag{4}$$

where V_j is the saliency value of the j-th LCU, and V_{max} is the largest saliency value among all LCUs in a frame. For protecting the subjective video quality, we have the following observation:

1) OBSERVATION

The LCUs with greater subjective weights should be assigned with larger maximum depths.

2) ANALYSIS

When the subjective weights of some LCUs are greater than those of others, it means that these LCUs can attract more visual attention. In other words, the video distortion of these LCUs, like the blocking artifacts, has greater effect on the whole subjective quality than others. Thus, for keeping the subjective quality, the video quality of LCUs with greater subjective weights should be protected as much as possible. We can see from Table 3 that along with the decreasing of LCU maximum depth, the video distortion is drastically increased. Accordingly, towards better subjective quality, larger maximum depths should be imposed on LCUs which have greater subjective weights. This completes the analysis of observation 1).

However, the subjective weight has lower relevance with optimal depth. For example, many LCUs have large subjective weights but their optimal depths are pretty small. Relying on subjective weights to determine the maximum depths of LCUs may incur objective quality and coding efficiency loss. That is also the main reason why [3] has large objective quality loss. By comparison, the objective weights can protect the objective quality, but may impair the subjective quality. Thus, in order to keep a balance between the objective and perceived quality, we take both the objective weight O_j and subjective weight S_j into consideration when deciding the maximum depths of LCUs in the process of complexity control.

C. COMPLEXITY CONTROL AT LCU LEVEL

The aim of LCU level complexity control is to reduce the encoding complexity of the frame to be a given target, which is based on the classified depth-complexity relationship modeled in Section III-A. Since the relationships are classfied for the 4-th frame and other three frames in GOPs, the controlling algorithm can be expressed by the following two optimizations:

$$\text{Optimization I.} \quad \underbrace{\min_{\{M_d\}} \left| \frac{1}{J} \sum_{j=1}^{J} E_c'(M_d)_j - \widetilde{T}_k \right|}_{\text{For the 4-th frame}}, \tag{5}$$

$$\text{Optimization II.} \quad \underbrace{\min_{\{M_d\}} \left| \frac{1}{J} \sum_{j=1}^{J} E_c''(M_d)_j - \widetilde{T}_k \right|}_{\text{For the first 3 frames}}, \tag{6}$$

where $\widetilde{T}_k$ ($\leq$ 100%) is the target complexity for the k-th frame, the value of which is decided by the frame level complexity control to be discussed in the next subsection. J is the total number of LCUs in each frame. $E_c'(M_d)_j$ and $E_c''(M_d)_j$ are the encoding complexity of the j-th LCU with maximum depth being M_d in the 4-th frame and the first 3 frames, respectively. Next, we concentrate on the solution to the optimization of (5) and (6).

For solving this optimization, we introduce a reasonable assumption first. That is, the encoding complexity of each

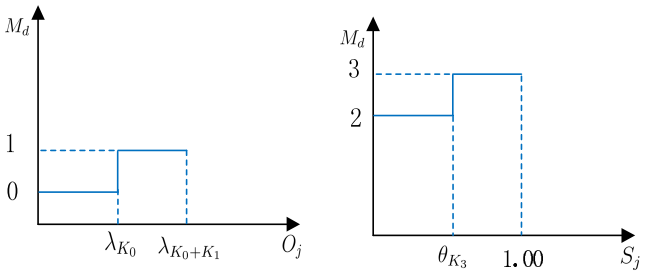


FIGURE 9. Examples of how to find the M_d for each LCU.

LCU is unified for LCUs with the same maximum depth M_d. This assumption is reasonable, because only the sum of LCU's encoding complexity is considered in the encoding complexity goal. Given this assumption, (5) can be turned to

$$\min_{\{K_n\}_{n=0}^{3}} \left| \frac{1}{J} \sum_{n=0}^{3} E_c'(M_d = n) K_n - \widetilde{T}_k \right| \quad \text{s.t.} \quad \sum_{n=0}^{3} K_n = J \quad (7)$$

where K_n is the number of LCUs with maximum depth M_d being n in the frame ($n \in \{0, 1, 2, 3\}$). Note that the sum of K_n is equivalent to the total number J of LCUs in the frame. Actually, (7) is a *NP*-hard optimization problem and can be solved using Branch-and-Bound algorithm. For details about this algorithm, we refer to [25]. After solving (7), we can get the values of K_n in the 4-th frame in GOP. The same solving process can be used to (6) to get K_n for the first 3 frames in GOP. Next, the most important thing is how to locate the LCUs which deserve the corresponding M_d.

Recall the objective and subjective weights in Section IV-A and IV-B, each LCU can have two important properties, i.e., the objective weight O_j and subjective weight S_j. Next, we design an algorithm to find the M_d of each LCU based on its normalised objective and subjective weight. Firstly, we need to sort the objective weights of all LCUs in the frame in an ascending order, and sort the subjective weights of all LCUs in the frame in a descending order. After that, we can get two thresholds, λ_{K_0} and $\lambda_{K_0+K_1}$, from the sorted objective weights, and one threshold θ_{K_3} from the sorted subjective weights. λ_{K_0} and $\lambda_{K_0+K_1}$ are the K_0-th and $(K_0 + K_1)$-th smallest objective weights, and θ_{K_3} is the K_3-th largest subjective weight. Then, through comparing O_j with λ_{K_0} and $\lambda_{K_0+K_1}$, and comparing S_j with θ_{K_3}, we can obtain the M_d of the j-th LCU. The specific comparing algorithm is shown in Figure 9. When O_j is smaller than λ_{K_0}, M_d is 0. When O_j is larger than λ_{K_0} and smaller than $\lambda_{K_0+K_1}$, M_d is 1. For LCUs whose O_j is larger than $\lambda_{K_0+K_1}$, M_d is determined according to S_j. Specifically, when S_j is larger than θ_{K_3}, M_d is 3. Otherwise, M_d is 2. Here, we use objective weights to determine the smaller M_d (i.e., 0 and 1) and subjective weights to decide the larger M_d (i.e., 2 and 3), because we believe this can help to keep both the objective and subjective video quality well, and the experimental results also verifies this technique's effectiveness.

Figure 10 shows the M_d distribution in a frame using our and [3] approaches. From this figure, we can see that our approach can allocate M_d to each LCU more reasonably. Specifically, our approach offers four different M_d for each LCU to choose from, while [3] only has two. This way, even with very low encoding complexity, e.g., 40% in Figure 10, our approach can still preserve the quality of important regions (e.g., the face region in Figure 10) well through allocating the largest M_d to them.

D. COMPLEXITY CONTROL AT FRAME LEVEL

After complexity control at LCU level, we can control the encoding complexity of each frame by allocating M_d to different LCUs in the frame. However, since the classified depth-complexity relationship is trained only by three sequences, it is hard for this relationship to be accurate enough for all the test sequences. In that case, the control accuracy can be negatively affected. For coping with this problem, we develop a complexity control algorithm at frame level, in order to compensate the control error and further improve the control accuracy.

The basic idea is to adjust the target complexity $\widetilde{T}_k$ for the $k-$th frame, to make it adaptive to the consumed encoding complexity of its previous encoded frames. To this end, we need to estimate a reference encoding time t_s based on an initial target complexity T_c. Specifically, the first N frames in the video should be encoded without complexity control. The whole encoding time of these N frames is recorded as t_g. According to t_g, t_s can be estimated by

$$t_s = T_c \cdot \frac{t_g}{N}. \quad (8)$$

In this paper, we set N to be the number of frames in the first 12 GOPs. As we can see from Figure 11, the average

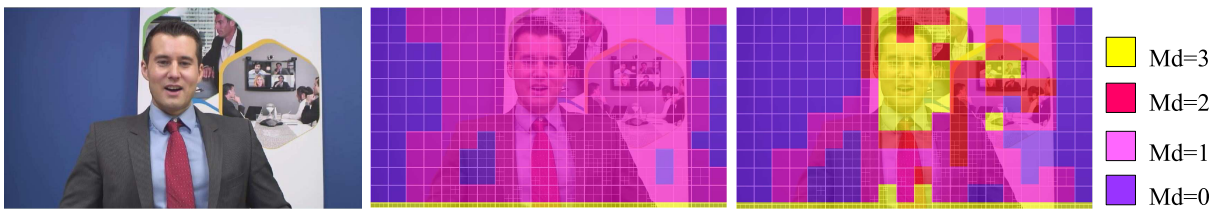


FIGURE 10. The maximum depth distribution among LCUs in our and [3] approaches. The pictures are all the 27-th frames of *Johnny* with 40% target complexity. (a) Original. (b) Approach [3]. (c) Our approach.

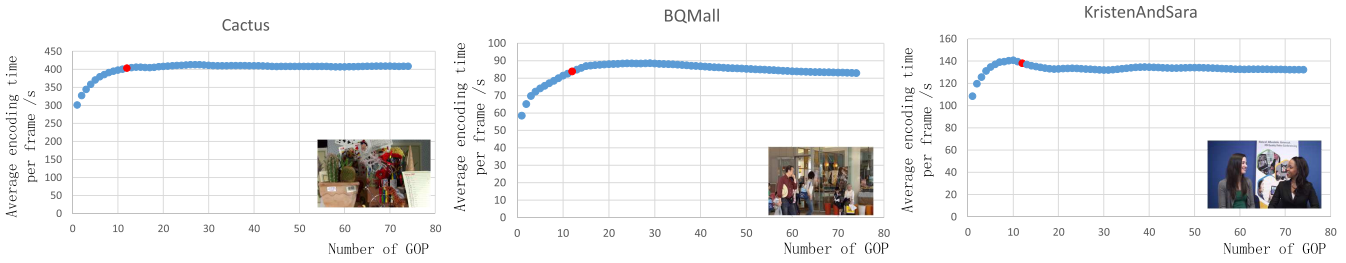


FIGURE 11. The change of averaged encoding time per frame with the increase of encoded GOPs. The horizontal axis is the number of encoded GOPs and the vertical axis is the averaged encoding time per frame. The red point indicates the averaged encoding time per frame of the first 12 GOPs.

encoding time of the first 12 GOPs is similar to the average time of encoding all frames.

Then, after the k-th ($k > N$) frame encoded, we record the encoding time e_k from the N-th frame to the k-th frame, and their averaged encoding time t_k can be calculated by

$$t_k = \frac{e_k}{k - N + 1}. \tag{9}$$

Next, according to t_s and t_k, we update target complexity of the $(k+1)$-th frame to $\widetilde{T}_{k+1}$ before encoding the $(k+1)$-th frame, using the following equation:

$$\widetilde{T}_{k+1} = \begin{cases} T_c - a & \text{if } t_k \geq \alpha t_s \\ T_c & \text{if } \beta t_s < t_k < \alpha t_s \\ T_c + b & \text{if } t_k \leq \beta t_s, \end{cases} \tag{10}$$

where α and β are the parameters to adjust the fluctuation range of complexity control. Note that α is larger than 1, and β is smaller than 1. Generally, the closer α and β are to 1, the narrower fluctuation range is allowed, but at the cost of higher fluctuation frequency. Besides, a and b are parameters to make a tradeoff between control accuracy and control fluctuation range. Usually, large values of a and b can improve the control accuracy, but at the expense of aggravated control fluctuation range. Suppose that the basic target complexity T_c is 60%, and both a and b are set to be 5%. According to (10), if $t_k \geq \alpha t_s$ or $t_k \leq \beta t_s$, the new target complexity $\widetilde{T}_{k+1}$ can make a quick and drastic adjustment, which may lead to a large fluctuation range between 55% (i.e., $T_c - a$) and 65% (i.e., $T_c + b$). However, thanks to this quick adjustment, the overall control accuracy can be quite high. In this paper, we empirically set a and b to be 2%, and α and β to be 1.05 and 0.95.

The frame level complexity budget can compensate the error of LCU level complexity control. When t_k is smaller than βt_s, indicating that there is more computational resource left for the subsequent un-encoded frames, the target complexity $\widetilde{T}_{k+1}$ can be increased slightly by (10). Otherwise, when t_k is larger than αt_s, the target complexity $\widetilde{T}_{k+1}$ is decreased a little in (10). For each frame, before encoding, its target complexity should be updated, to adaptively compensate the error of complexity control on its previous frames. According to the updated $\widetilde{T}_{k+1}$, the complexity control at LCU level is done for the new frame. The overall algorithm of our approach can be seen in Table 5.

TABLE 5. The basic algorithm of our SOCAS approach.

- **Input:** The initial target complexity T_c.
- **Output:**The maximum depth for each LCU in each frame.
- Initialize F to the total number of frames in a video.
- Initialize J to the total number of LCUs in a frame.
- Initialize N to the number of frames without complexity control.
- **For** $k = 1, k \leq N, k$++
  Record the total encoding time of the first N frames as t_g.
  End
- Calculate the reference encoding time t_s using (8).
- **For** $k = N+1, k < F, k$++
  1 **If** k is the 4-th frame in GOP.
  Calculate $\{K_n\}_{n=1}^{3}$ through solving (5)
  Else
  Calculate $\{K_n\}_{n=1}^{3}$ through solving (6).
  3 **For** $j = 0, j < J, j$++
  Calculate normalised objective weight O_j and subjective weight S_j for the j-th LCU.
  End
  Sort O_j and S_j, and get λ_{K_0}, $\lambda_{K_0+K_1}$, and θ_{K_3}.
  4 **For** $j = 0, j < J, j$++
  If $O_j < \lambda_{K_0}$, M_d=0.
  Else If $O_j < \lambda_{K_0+K_1}$, M_d=1.
  Else If $S_j > \theta_{K_3}$, M_d=3.
  Else M_d=2.
  End
  5 Calculate the averaged encoding time t_k using (9).
  6 Update target complexity $\widetilde{T}_{k+1}$ using (10).
  End
- Return the M_d for each LCU in a frame.

V. EXPERIMENTAL RESULTS

In this section, we evaluate the performance of our HCC approach from several aspects. We compare our approach with other two state-of-the-art complexity control approaches for HEVC, i.e., [3] and [22]. The experiments were carried out on HEVC test software HM 16.0, with typical configurations presented in Table 6. The test sequences were chosen from HEVC standard test sequence database. All the sequences were tested with four different QPs, i.e., 22, 27, 32, and 37. Note that the test sequences are different from the training sequences that have been used in Section III-A. Since we focus on the encoding complexity control of live videos, the Low-delay P configuration was used in the experiment.

TABLE 6. Typical configurations of HM 16.0.

LCU size	64 × 64
Maximum LCU depth	3
Coding configuration	low delay P main
SAO	1
FEN	1
Intra period	-1
GOP structure	IPPP

A. CONTROL ACCURACY AND FLUCTUATION

Table 7 shows the performance of the proposed approach in terms of control accuracy, ΔPSNR and ΔBR. Here, ΔPSNR and ΔBR are calculated according to [26]. In this table, R_c stands for the actual running complexity, and we can see that our approach can make R_c quite close to the target complexity. Specifically, the averaged R_c is 80.54% for 80% target complexity, 61.05% for 60% target complexity, 40.68% for 40% target complexity, and 21.49% for 20% target complexity. We also compare the results with [3] and [22] in Tables 8 and 9, and we can easily find that our approach has a smaller control error than the others.

As can be seen in Figure 7, the ΔPSNR and ΔBR performance of our approach are better in videos like *Johnny* and *Vidyo1* which have small scene changes. For *Johnny* at 60%, it is even surprising to find that the ΔPSNR is 0 and ΔBR is negative. It means that we decrease the encoding complexity to 60% with no PSNR loss and using less bits. While for the videos like *BasketballDrive* and *PartyScene* which have large scene changes, their performance is not as good as videos like *Johnny*. This is because the large scene changes may have influence on the objective weight maps which rely on the bit allocation of previous frame. Despite of this, we can still find in Tables 8 and 9 that the ΔPSNR and ΔBR performance of ours are better than the state-of-the-art approaches [3], [22].

Another advantage of the proposed approach is that it has small generalization error. Actually, the test sequences in Table 7 can be divided into two categories: camera-captured videos (i.e., Class B, C, and E) and screen content videos (i.e., Class F). The training process only uses three camera-captured training sequences. Surprisingly, as can be seen in Table 7, the training model not only works well in camera-captured videos, but also performs well in screen content videos (e.g., *SlideShow* and *SlideEdit*).

In addition to the control accuracy, the control fluctuation is another element to evaluate the performance of our approach. We compare in Figure 12 the control fluctuation performance of our and other two approaches with 60% target complexity. From Figure 12, we can obviously see that our approach can achieve complexity control more steadily than [22]. For [3], there is no obvious difference of control fluctuation between us. But the PSNR loss fluctuation performance of [3] is worse than ours. Here the PSNR loss is the PSNR difference between the 100% complexity and the target decreased complexity.

B. CONTROL CONVERGENCE

Apart from the aforementioned two properties, the convergence property of the complexity control algorithm is also an

TABLE 7. Complexity control performance evaluation of our approach with different target complexities.

Test Sequence (class)	Resolution	T_C=80%			T_C=60%			T_C=40%			T_C=20%		
		R_C(%)	ΔPSNR	ΔBR(%)	R_C(%)	ΔPSNR	ΔBR(%)	R_C(%)	ΔPSNR	ΔBR(%)	R_C(%)	ΔPSNR	ΔBR(%)
BasketballDrive (B)	1920×1080	78.58	-0.03 dB	0.33	60.24	-0.15 dB	2.02	40.37	-0.28 dB	6.03	20.27	-0.49 dB	9.23
BQTerrace (B)	1920×1080	79.75	-0.02 dB	0.33	61.90	-0.08 dB	3.20	41.62	-0.31 dB	6.59	21.38	-0.55 dB	10.98
ParkScene (B)	1920×1080	80.00	-0.03 dB	0.60	61.27	-0.14 dB	5.87	40.02	-0.34 dB	8.24	21.28	-0.61 dB	11.09
Johnny (E)	1280×720	80.84	0.00 dB	-0.05	59.10	0.00 dB	-0.02	41.74	-0.06 dB	2.21	22.19	-0.10 dB	6.48
Fourpeople (E)	1280×720	80.93	0.00 dB	0.13	60.23	-0.01 dB	0.44	41.10	-0.24 dB	6.50	21.07	-0.38 dB	10.32
Vidyo1(E)	1280×720	81.98	0.00 dB	0.03	62.19	-0.02 dB	0.62	41.20	-0.05 dB	1.89	20.11	-0.11 dB	7.38
Vidyo3(E)	1280×720	81.11	0.00 dB	0.04	63.36	-0.05 dB	1.55	38.37	-0.11 dB	3.67	21.22	-0.21 dB	10.38
Vidyo4(E)	1280×720	82.28	0.00 dB	0.15	61.01	-0.01 dB	0.21	40.02	-0.10 dB	3.77	21.05	-0.17 dB	7.21
PartyScene (C)	832×480	78.73	-0.10 dB	2.53	60.50	-0.25 dB	8.29	41.42	-1.03 dB	13.24	21.73	-1.78 dB	20.32
RaceHorses (C)	832×480	78.31	-0.08 dB	1.56	58.13	-0.45 dB	7.37	37.55	-1.34 dB	11.12	22.57	-1.90 dB	20.56
BasketballDrill(C)	832×480	82.75	-0.02 dB	0.46	62.71	-0.18 dB	4.85	42.72	-0.27 dB	6.36	22.25	-0.44 dB	8.10
SlideShow (F)	1280×720	81.53	-0.03 dB	0.42	61.06	-0.30 dB	3.32	41.65	-0.87 dB	6.54	22.05	-1.43 dB	8.19
SlideEdit (F)	1280×720	80.18	-0.03 dB	0.40	62.00	-0.06 dB	0.58	41.12	-0.12 dB	1.83	22.23	-0.83 dB	2.90
Average	-	**80.54**	-0.02 dB	0.53	**61.05**	-0.13 dB	2.95	**40.68**	-0.39 dB	6.00	**21.49**	-0.69 dB	10.24

TABLE 8. Complexity control performance comparison between our and other approaches with 60% target.

T_C=60%	Resolution	Our approach			[3]			[22]		
		R_C(%)	ΔPSNR	ΔBR(%)	R_C(%)	ΔPSNR	ΔBR(%)	R_C(%)	ΔPSNR	ΔBR(%)
BasketballDrive (B)	1920×1080	60.24	-0.15 dB	2.02	62.12	-0.20 dB	4.29	71.81	-0.17 dB	3.59
BQTerrace (B)	1920×1080	61.90	-0.08 dB	3.20	63.90	-0.19 dB	7.33	65.33	-0.25 dB	6.21
ParkScene (B)	1920×1080	61.27	-0.14 dB	5.87	64.78	-0.34 dB	8.99	64.67	-0.31 dB	7.22
Johnny (E)	1280×720	59.10	0.00 dB	-0.02	62.23	-0.03 dB	2.24	65.18	0.00 dB	-0.10
Fourpeople (E)	1280×720	60.23	-0.01 dB	0.44	63.09	-0.11 dB	3.64	67.78	-0.03 dB	0.68
Vidyo1(E)	1280×720	62.19	-0.02 dB	0.62	65.22	-0.08 dB	1.03	69.64	-0.02 dB	0.24
Vidyo3(E)	1280×720	63.36	-0.05 dB	1.55	65.31	-0.07 dB	2.87	69.72	-0.03 dB	0.66
Vidyo4(E)	1280×720	61.01	-0.01 dB	0.21	63.32	-0.04 dB	1.36	69.99	-0.04 dB	0.87
PartyScene (C)	832×480	60.50	-0.25 dB	8.29	63.22	-0.53dB	14.10	57.05	-1.01 dB	25.12
RaceHorses (C)	832×480	58.13	-0.45 dB	7.37	58.13	-0.65 dB	15.39	52.83	-0.53 dB	14.20
BasketballDrill(C)	832×480	62.71	-0.18 dB	4.85	63.06	-0.78 dB	8.54	72.99	-0.24 dB	5.19
SlideShow (F)	1280×720	61.06	-0.30 dB	3.32	58.15	-0.56 dB	6.79	64.11	-0.52 dB	7.43
SlideEdit (F)	1280×720	62.00	-0.06 dB	0.58	57.60	-0.15 dB	1.36	67.48	-0.11 dB	0.42
Average	-	**61.05**	**-0.13 dB**	**2.95**	**62.31**	**-0.29 dB**	**5.99**	**62.06**	**-0.25 dB**	**5.52**

TABLE 9. Complexity control performance comparison between our and other approaches with 40% target.

T_c=40%	Resolution	Our approach			[3]			[22]		
		R_c(%)	ΔPSNR	ΔBR(%)	R_c(%)	ΔPSNR	ΔBR(%)	R_c(%)	ΔPSNR	ΔBR(%)
BasketballDrive (B)	1920×1080	40.37	-0.28 dB	6.03	44.32	-0.47 dB	13.21	51.14	-0.85 dB	15.73
BQTerrace (B)	1920×1080	41.62	-0.31 dB	6.59	42.23	-0.65 dB	14.87	33.01	-0.81 dB	17.56
ParkScene (B)	1920×1080	40.02	-0.34 dB	8.24	41.39	-1.57 dB	16.73	35.18	-0.94 dB	15.20
Johnny (E)	1280×720	41.74	-0.06 dB	2.21	37.22	-0.12 dB	4.54	35.22	-0.21 dB	5.72
Fourpeople (E)	1280×720	41.10	-0.24 dB	6.50	35.27	-0.44dB	10.53	46.81	-0.35 dB	10.78
Vidyo1(E)	1280×720	41.20	-0.05 dB	1.89	42.68	-0.22 dB	10.54	27.09	-0.23 dB	9.89
Vidyo3(E)	1280×720	38.37	-0.11 dB	3.67	37.34	-0.23 dB	11.31	31.93	-0.28 dB	10.32
Vidyo4(E)	1280×720	40.02	-0.10 dB	3.77	41.79	-0.27 dB	12.23	33.89	-0.21 dB	5.87
PartyScene (C)	832×480	41.42	-1.03 dB	13.24	45.20	-1.57 dB	28.16	33.53	-1.65 dB	33.21
RaceHorses (C)	832×480	37.55	-1.34 dB	11.12	45.53	-1.77 dB	27.49	35.39	-2.03 dB	35.78
BasketballDrill(C)	832×480	42.72	-0.27 dB	6.36	42.20	-0.68 dB	13.27	35.47	-1.28 dB	15.77
SlideShow (F)	1280×720	41.65	-0.87 dB	6.54	46.06	-1.37 dB	16.38	46.69	-0.97 dB	14.82
SlideEdit (F)	1280×720	41.12	-0.12 dB	1.83	45.00	-1.24 dB	15.26	43.49	-0.78 dB	12.99
Average	-	**40.68**	**-0.39 dB**	**6.00**	**42.02**	**-0.82 dB**	**14.96**	**37.75**	**-0.87 dB**	**15.66**

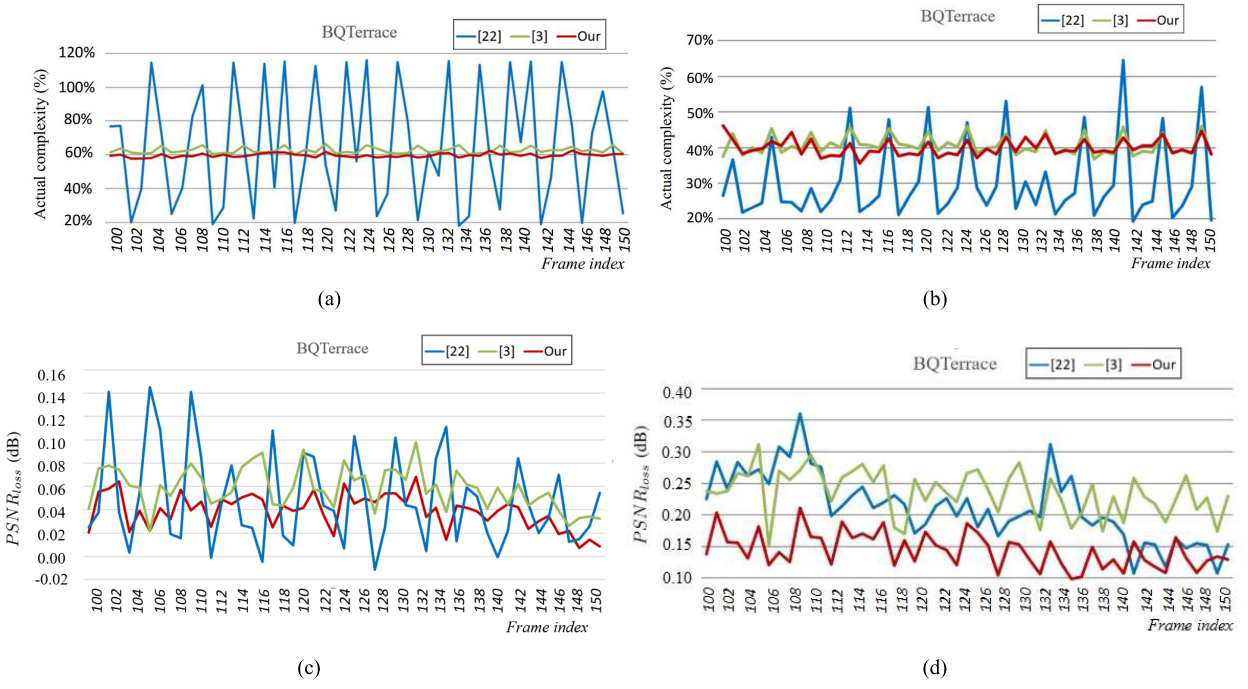


FIGURE 12. The control fluctuation and PSNR loss fluctuation comparisons between our, [3] and [22] approaches. (a) Control fluctuation with 60% target complexity. (b) Control fluctuation with 40% target complexity. (c) PSNR loss fluctuation with 60% target complexity. (d) PSNR loss fluctuation with 40% target complexity.

important indicator of the algorithm performance. Here, the convergence ability indicates how quickly and accurately the approach can make an adjustment to the change of the target complexity during the encoding process. In order to investigate the convergence ability of our approach, we set the target complexity of the first 48 frames to be 100%, and the target complexity from the 49-th to 150-th frames was set to 80%, and the target complexity from the 150-th to 300-th frames was set to 40%. We aim to find the response of the actual running complexity when the target complexity is abruptly dropped from 100% to 80% (small change), and from 80% to 40% (large change). Figure 13 shows the experimental results with the above experimental setup. We can see that the proposed approach can react instantly and accurately to the change of target complexity, whether the change is large

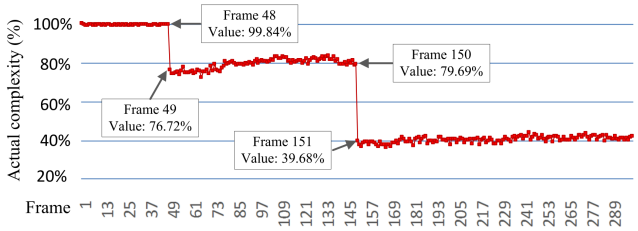


FIGURE 13. The convergence property of the proposed approach. The results are from sequence *PartyScene*.

or small. Specifically, when the target complexity is reduced from 100% in the 48-th frame to 80% in the 49-th frame, the actual running complexity immediately decreases

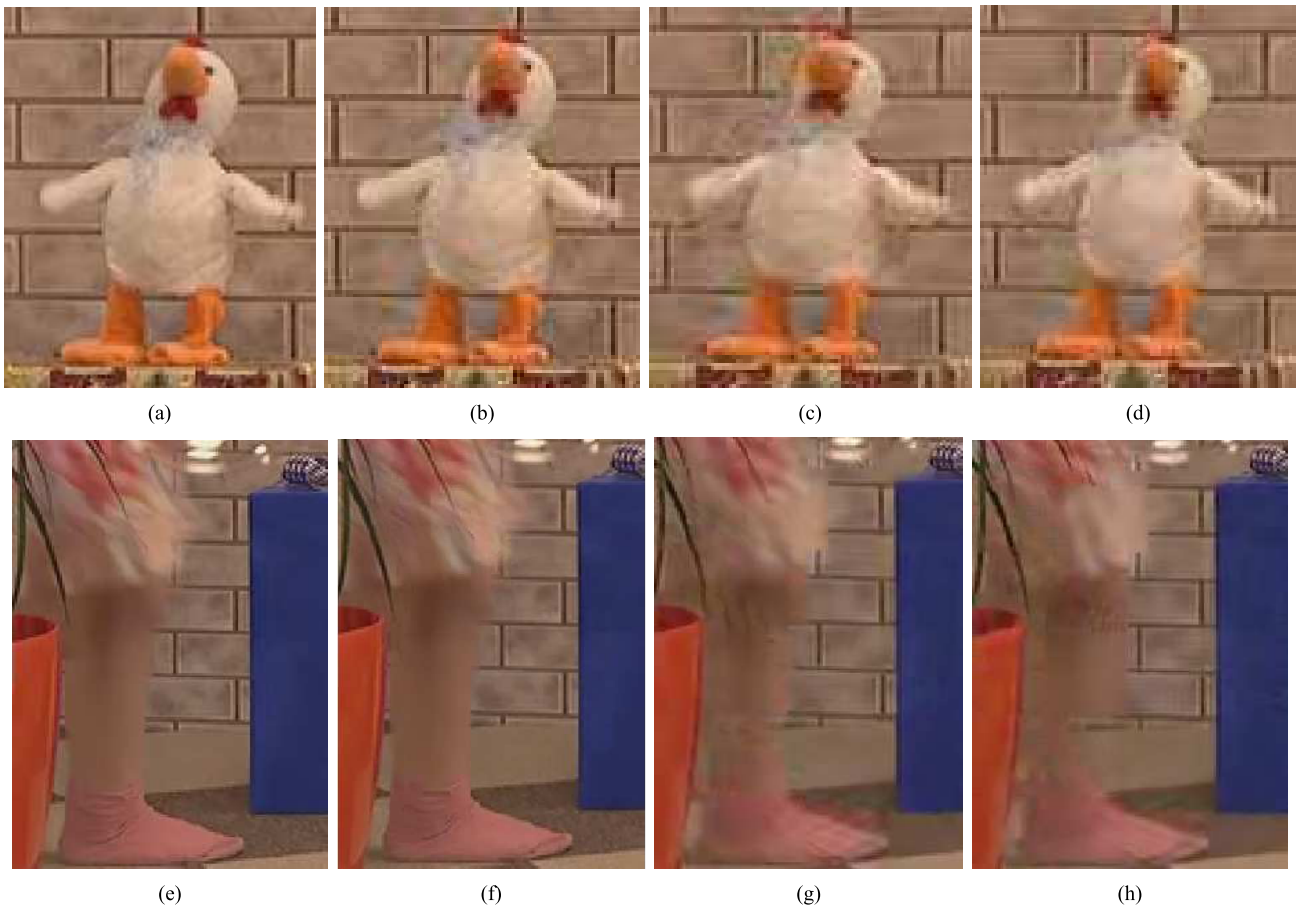


FIGURE 14. Examples of video quality comparisons between our, [3] and [22] approaches. The first row pictures are all selected from the 99-th frame of *PartyScene*, and the second row pictures are all from the 136-th frames of *PartyScene*, at 40% target complexity, with QP=32. (a) Original. (b) Our approach. (c) Approach [3]. (d) Approach [22]. (e) Original. (f) Our approach. (g) Approach [3]. (h) Approach [22].

TABLE 10. Subjective quality evaluation results.

Sequence	BasketballDrive	BQTerrace	ParkScene	Johnny	Fourpeople	Vidyo1	Vidyo3	Vidyo4	PartyScene	RaceHorses	BasketballDrill	SlideShow	SlideEdit
Our DMOS	58.13	50.26	48.80	27.10	51.74	48.79	48.69	43.02	34.52	51.35	43.64	51.02	44.77
[22] DMOS	58.72	49.80	54.49	42.09	55.34	52.65	55.30	50.85	67.24	55.71	53.60	50.76	51.57
DMOS Difference	-0.59	0.46	-5.69	-14.99	-3.60	-3.86	-6.61	-7.83	-32.72	-4.36	-9.96	0.26	-6.80

from 99.84% in the 48-th frame to 76.72% in the 49-th frame. When the target complexity is changed from 80% in the 150-th frame to 40% in the 151-th frame, the running complexity instantly decreases from 79.69% to 39.68%. Our approach has the ability to quickly adjust the encoding complexity to reach the new target only in one frame.

C. OBJECTIVE AND SUBJECTIVE QUALITY ASSESSMENT

We have shown in Tables 8 and 9 the objective quality superiority of our approach over the other two approaches in details. For directly showing the superiority of our approach in video quality, we present in Figure 14 the same compressed pictures using our and other approaches. In Figure 14, we can observe significant blocking artifacts in toy's face and girl's leg in other approaches, especially [22], whereas there is no such severe artifacts in our approach.

In addition to the objective quality, we also do the subjective quality evaluation using a single stimulus continuous quality scale (SSCQS) procedure, proposed by Rec. ITU-R BT.500 [27] and improved by [28]. There were 16 subjects involved in the experiments, including 5 females and 11 males. The videos were played on a 23'' DELL U2312HM LCD monitor with its resolution being 1920 × 1080. After the experiment, we recorded the difference mean opinion score (DMOS) of each video as the indicator of the subjective quality. The smaller DMOS value means the better subjective

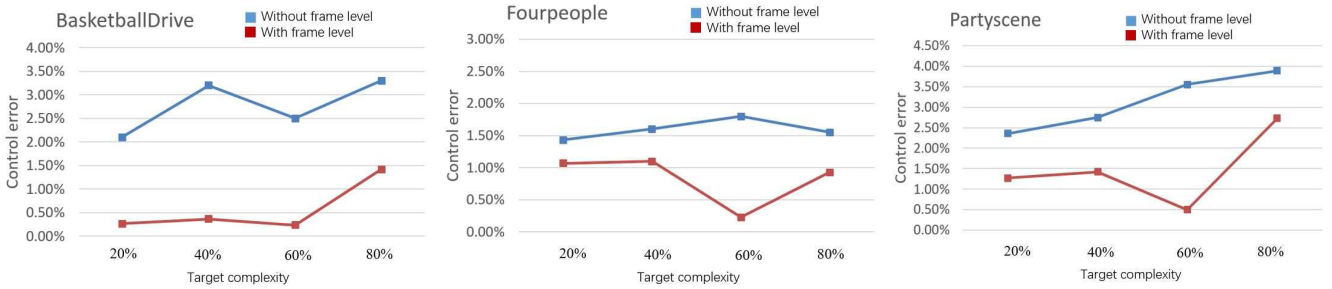


FIGURE 15. The comparison of control accuracy with and without frame level complexity control algorithm. The vertical axis is the control error-the absolute difference between actual encoding complexity and target encoding complexity.

quality. Table 10 shows the subjective quality comparison between our and comparing approaches with 60% complexity. We can see that except for two sequences, i.e., *BQTerrace* and *SlideShow*, of which our DMOS is slightly larger than the comparing DMOS, our DMOS is on average much smaller than the comparing [22] DMOS. Thus, our approach can offer better subjective video quality than [22] when reducing the encoding complexity to the similar level. While for another comparing approach [3] in this paper of which the results are not shown, there is no big difference between us, mainly because we adopt the same PQFT algorithm to calculate the subjective weights. However, our superiorly over [3] is significant in the aspects of objective quality and BD-rate.

D. CONTROL CONTROL PERFORMANCE WITHOUT FRAME LEVEL CONTROL

For demonstrating the effectiveness of our frame level complexity control algorithm, we remove the frame level algorithm from our complete HCC approach, and then test its performance. As shown in Figure 15, the control error is increased without frame level algorithm. This verifies that the frame level complexity control algorithm contributes to increasing the control accuracy. This is because the classified depth-complexity relationship (i.e., $E_c'(M_d)$ and $E_c''(M_d)$) is obtained by training only three sequences, which is not quite suitable for every sequence. As a result, while the relationship is accurate for one sequence, it is not quite accurate for another sequence. Without frame level algorithm, the control accuracy can only rely on the trained relationship, and thus inaccurate relationship may lead to control error.

In conclusion, the aim of frame level control algorithm is to compensate the control error caused by the inaccurate relationship through adjusting the target complexity for each frame. Thus, it plays an important role in increasing the control accuracy in our work.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a novel HCC approach to hierarchically control the HEVC complexity for live videos' encoding. The complexity control in our approach is achieved at two levels: the LCU level and the frame level. At the LCU level, the maximum depth of each LCU is calculated to achieve the target complexity designated for each frame. However, there may exist some deviations between the actual running complexity and the target complexity of each frame. Then, a frame level complexity control algorithm is developed to compensate such control deviations, to further improve the control accuracy. The experimental results show that our approach can achieve complexity control of HEVC with very high accuracy, average no more than 2%. In addition, one advantage of the proposed algorithm is that it can make a quick adjust to the abrupt change of the target complexity during the encoding process. For protecting the video quality after complexity reduction, the objective map and subjective map, i.e., the bit allocation map and saliency value map respectively, are introduced in this paper to keep the objective and subjective quality. The experimental results prove that our approach outperforms other approaches with better ΔBR and ΔPSNR performance.

An interesting future work would be to find a more accurate objective map to integrate with the bit allocation map in this paper to further improve the ΔBR and ΔPSNR performance. For the videos with high scene changing rate, the bit allocation map has delay to describe the objective weights of LCUs and the video quality may be influenced. Thus, it would be quite useful to design an objective map which can adapt to the scene change.

REFERENCES

[1] T. K. Tan *et al.*, "Video quality evaluation methodology and verification testing of HEVC compression performance," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 76–90, Jan. 2016.
[2] G. Correa, P. Assuncao, L. Agostini, and L. A. Da Silva Cruz, "Coding tree depth estimation for complexity reduction of HEVC," in *Proc. Data Compress. Conf. (DCC)*, Mar. 2013, pp. 43–52.
[3] X. Deng, M. Xu, L. Jiang, X. Sun, and Z. Wang, "Subjective-driven complexity control approach for HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 91–106, Jan. 2016.
[4] Y. Zhang, H. Wang, and Z. Li, "Fast coding unit depth decision algorithm for interframe coding in HEVC," in *Proc. Data Compress. Conf. (DCC)*, Mar. 2013, pp. 53–62.
[5] S. Ahn, M. Kim, and S. Park, "Fast decision of CU partitioning based on SAO parameter, motion and PU/TU split information for HEVC," in *Proc. Picture Coding Symp. (PCS)*, Dec. 2013, pp. 113–116.

[6] J. Xiong, H. Li, F. Meng, S. Zhu, Q. Wu, and B. Zeng, "MRF-based fast HEVC inter CU decision with the variance of absolute differences," *IEEE Trans. Multimedia*, vol. 16, no. 8, pp. 2141–2153, Dec. 2014.

[7] X. Shen, L. Yu, and J. Chen, "Fast coding unit size selection for HEVC based on Bayesian decision rule," in *Proc. Picture Coding Symp. (PCS)*, May 2012, pp. 453–456.

[8] J. Lee, S. Kim, K. Lim, and S. Lee, "A fast CU size decision algorithm for HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 411–421, Mar. 2015.

[9] L. Shen, Z. Liu, X. Zhang, W. Zhao, and Z. Zhang, "An effective CU size decision method for HEVC encoders," *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 465–470, Feb. 2013.

[10] J. Leng, L. Sun, T. Ikenaga, and S. Sakaida, "Content based hierarchical fast coding unit decision algorithm for HEVC," in *Proc. Int. Conf. Multimedia Signal Process. (CMSP)*, vol. 1. May 2011, pp. 56–59.

[11] J. Xiong, H. Li, Q. Wu, and F. Meng, "A fast HEVC inter CU selection method based on pyramid motion divergence," *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 559–564, Feb. 2014.

[12] J. Kim, J. Yang, K. Won, and B. Jeon, "Early determination of mode decision for HEVC," in *Proc. Picture Coding Symp. (PCS)*, May 2012, pp. 449–452.

[13] M. U. K. Khan, M. Shafique, and J. Henkel, "An adaptive complexity reduction scheme with fast prediction unit decision for HEVC intra encoding," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2013, pp. 1578–1582.

[14] H.-M. Yoo and J.-W. Suh, "Fast coding unit decision algorithm based on inter and intra prediction unit termination for HEVC," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2013, pp. 300–301.

[15] C.-C. Wang, Y.-C. Liao, J.-W. Wang, and C.-W. Tung, "An effective TU size decision method for fast HEVC encoders," in *Proc. Int. Symp. Comput., Consum. Control (IS3C)*, Jun. 2014, pp. 1195–1198.

[16] H. Zhang and Z. Ma, "Fast intra mode decision for high efficiency video coding (HEVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 4, pp. 660–668, Apr. 2014.

[17] J. Vanne, M. Viitanen, and T. D. Hamalainen, "Efficient mode decision schemes for HEVC inter prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 9, pp. 1579–1593, Sep. 2014.

[18] K. Miyazawa, T. Murakami, A. Minezawa, and H. Sakate, "Complexity reduction of in-loop filtering for compressed image restoration in HEVC," in *Proc. Picture Coding Symp. (PCS)*, May 2012, pp. 413–416.

[19] G. Correa, P. A. Assuncao, L. V. Agostini, and L. A. da Silva Cruz, "Fast HEVC encoding decisions using data mining," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 4, pp. 660–673, Apr. 2015.

[20] R. Garcia, D. Ruiz-Coll, H. Kalva, and G. Fernández-Escribano, "HEVC decision optimization for low bandwidth in video conferencing applications in mobile environments," in *Proc. ICMEW*, Jul. 2013, pp. 1–6.

[21] T. Zhao, Z. Wang, and S. Kwong, "Flexible mode selection and complexity allocation in high efficiency video coding," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1135–1144, Dec. 2013.

[22] A. Jiménez-Moreno, E. Martínez-Enríquez, and F. Díaz-de-María, "Complexity control based on a fast coding unit decision method in the HEVC video coding standard,"*IEEE Trans. Multimedia*, vol. 18, no. 4, pp. 563–575, Apr. 2016.

[23] 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.

[24] C. Guo and L. Zhang, "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 185–198, Jan. 2010.

[25] J. Clausen, "Branch and bound algorithms-principles and examples," Dept. Comput. Sci., Univ. Copenhagen, Copenhagen, Denmark, Tech. Rep., 1999, pp. 1–30. [Online]. Available: http://www.diku.dk/OLD/undervisning/2003e/datV-optimer/JensClausenNoter.pdf

[26] G. Bjøntegaard, *Calculation of Average PSNR Differences Between RD-Curves*, document VCEG-m33, ITU-Telecommunication Standardization, 2001, pp. 290–294.

[27] ITU, *Methodology for the Subjective Assessment of the Quality of Television Pictures*. Standard BT.500-11, International Telecommunication Union, ITU, Geneva, Switzerland, 2002, pp. 53–56. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.500-11-200206-SHPDF-E.pdf

[28] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack, "Study of subjective and objective quality assessment of video," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1427–1441, Jun. 2010.

XIN DENG (S'14) received the B.S. degree and M.S. degree in electrical engineering from Beihang University in 2013 and 2016, respectively. She is currently pursuing the Ph.D. degree with the Department of Electrical and Electronic Engineering, Imperial College London. In 2014, she received the IEEE Circuits and Systems Society Student Travel Award. In 2016, she won the CSC-Imperial College London Scholarship for three year's Ph.D. study. Her research interests include image processing, perceptual image and video compression.

MAI XU (M'10) received the B.S. degree from Beihang University, Beijing, China, in 2003, the M.S. degree from Tsinghua University, Beijing, China, in 2006, and the Ph.D. degree from Imperial College London, London, U.K., in 2010. From 2010 to 2012, he was a Research Fellow with the Electrical Engineering Department, Tsinghua University. Since 2013, he has been with Beihang University as an Associate Professor. During 2014 to 2015, he was a Visiting Researcher with MSRA. He has authored over 60 technical papers in international journals and conference proceedings. His research interests mainly include visual communication and image processing. He received the best paper awards of two IEEE conferences.

CHEN LI is currently pursuing the bachelor's degree with Beihang University. His research interests include image and video signal processing, and multi-view video coding standard in VR.