Saliency-Guided Complexity Control for HEVC Decoding

Ren Yang, Student Member, IEEE, Mai Xu, Senior Member, IEEE, Zulin Wang, Member, IEEE, and Xiaoming Tao, Member, IEEE

Abstract—The latest High Efficiency Video Coding (HEVC) standard significantly improves coding efficiency over its previous video coding standards. The expense of such improvement is enormous computational complexity, from both encoding and decoding sides. Since computational capability and power capacity are diverse across portable devices, it is necessary to reduce decoding complexity to a target with tolerable quality loss, so-called complexity control. This paper proposes a Saliency-Guided Complexity Control (SGCC) approach for HEVC decoding, which reduces the decoding complexity to the target with minimal perceptual quality loss. First, we establish the SGCC formulation to minimize perceptual quality loss at the constraint on reduced decoding complexity, which is achieved via disabling Deblocking Filter (DF) and simplifying Motion Compensation (MC) of some non-salient Coding Tree Units (CTUs). One important component in this formulation is the modelled relationship between decoding complexity reduction and DF disabling/MC simplification, which determines the control accuracy of our approach. Another component is the modelled relationship between quality loss and DF disabling/MC simplification, responsible for optimizing perceptual quality. By solving the SGCC formulation for a given target complexity, we can obtain the DF and MC settings of each CTU, and then decoding complexity can be reduced to the target. Finally, the experimental results validate the effectiveness of our SGCC approach, from the aspects of control performance, complexity-distortion performance, fluctuation of quality loss and subjective quality.

Index Terms—HEVC, decoding complexity reduction, decoding complexity control.

I. INTRODUCTION

HIGH Efficiency Video Coding (HEVC) standard [1] was officially approved in April 2013, significantly improving the efficiency of video coding. It is able to save around 60% bit rates with similar subjective quality [2], compared with its former H.264/AVC standard. However, the cost of bit rate saving in HEVC is the huge computational complexity [3], from the aspects of both encoding and decoding. It is thus necessary to reduce encoding and decoding complexity of HEVC. The past couple of years have witnessed extensive works [4]–[10] on encoding complexity reduction for HEVC. Unfortunately, there are relatively few approaches on reducing HEVC decoding complexity. Actually, decoding is far more common than encoding for existing coding standards including HEVC. For example, according to [11], the amount of videos encoded and uploaded to YouTube is only around 65 thousands every day, while there are about 100 millions videos are decoded and viewed everyday. The number of decoded videos is more than 1,000 times of encoded videos. Therefore, the study on complexity reduction is more urgent for decoding.

Moreover, different devices may be diverse in computational capability. For example, the computational capability of laptops (e.g., MacBook) is probably over twice higher than that of tablets (e.g., iPad) [12]. Therefore, HEVC decoding need to be adaptive to diverse computational capability. That is, it is necessary to study on reducing HEVC decoding complexity to a target, via developing complexity control approach. Unfortunately, to our best knowledge, there exists few works on complexity control for HEVC decoding. In this paper, we propose an efficient approach to achieve this goal.

B. Related works

In early time, there existed a handful of studies [13], [14] on decoding complexity reduction, for the previous H.264/AVC standard. Most recently, several approaches [15]–[24] have been proposed to reduce decoding complexity/time, for the latest HEVC standard. Among them, there are two main research directions: hardware-based and algorithmic approaches.

Some works, such as [15]–[20], have been devoted to accelerating the HEVC decoding speed using hardware techniques. For example, Yan et al. [15] and Chi et al. [16] proposed to take advantage of Single Instruction Multiple Data (SIMD) instructions for increasing HEVC decoding speed. Souza et al. [17] achieved the HEVC decoding acceleration, which benefits from the parallel computing of Graphics Processing Unit (GPU). Similarly, [25] presented a new parallelization approach for accelerating HEVC decoding speed with higher
frame rate. The above approaches can save HEVC decoding time in some specific hardware, but they cannot reduce the complexity and power consumed by HEVC decoding. For reducing power consumption, [13] and [26] were proposed to dynamically adjust the frequency of CPU, taking advantage of Dynamic Voltage and Frequency Scaling (DVFS) technology. As such, the decoding power consumption can be reduced for H.264/AVC [13] and HEVC [26], by means of the dynamic adjustment of CPU frequency. In the Field-Programmable Gate Array (FPGA) platform, [20] achieved the power reduction in HEVC decoding, by designing a high-performance intra prediction hardware based on Verilog Hardware Description Language (Verilog HDL). However, all these approaches can be merely implemented on the specific hardware (e.g., GPU with SIMD, DVFS, FPGA, etc.) at the decoder side, and they are hardly adaptive to generic hardware.

For overcoming the drawback of hardware-based approaches, some algorithmic approaches have been developed to decrease video decoding complexity, via simplifying some encoding/decoding components. These approaches include [14], [21]–[24]. For H.264/AVC, Liu et al. [14] proposed to detect Region-of-Interest (ROI), and to allocate less computational resources to non-ROIs. Specifically, the total decoding complexity can be reduced with simplified coding components, according to an ROI based Rate-Distortion-Complexity (R-D-C) cost function. Later, Naccari et al. [21] proposed an approach for reducing decoding complexity of both H.264/AVC and HEVC. In [21], the offsets in Deblocking Filter (DF) are estimated with optimization on Generalized Block-edge Impairment Metric (GBIM), instead of the conventional brute force optimization. This way, the computational complexity of decoding can be saved. For HEVC, the decoding complexity is reduced in [22], by modifying the structure of prediction during encoding. However, [22] is not practical for already encoded videos, since it requires the modification at the encoder side. Most recently, [23] and [24] have been proposed to modify the components at the decoder side, to make decoding complexity reduction more practical in HEVC. To be more specific, they proposed to remove some in-loop filters, and to shorten the FIR filter sizes in Motion Compensation (MC), such that HEVC decoding complexity can be reduced. In comparison with hardware-based approaches, the algorithmic approaches on decoding complexity reduction can be implemented in any power-limited devices, but at the expense of visual quality loss.

Unfortunately, all above approaches, from both hardware-based and algorithmic aspects, cannot reduce the decoding complexity to a given target, leading to insufficient or wasteful use of power resources in some portable devices. There are only a few works on controlling decoding complexity for video coding. For example, Langroodi et al. [27] developed a decoding complexity control approach for H.264/AVC. In [27], the decoder sends its computational resource demand to the encoder side. Then, MC is optimized at the encoder side, such that decoding complexity can be controlled at the decoder side. However, [27] can be only applied to the previous H.264/AVC standard, and it is not suitable for off-line decoding because of the communication between encoder and decoder sides. To our best knowledge, there exists no approach on controlling decoding complexity for the latest HEVC standard or for off-line scenarios. More importantly, for HEVC all existing complexity reduction approaches do not take perceptual visual quality into consideration, which can be well modelled by video saliency [28]–[30].

C. Our work and contributions

In this paper, we propose a Saliency-Guided Complexity Control (SGCC) approach, which controls decoding complexity of HEVC, with minimization on perceptual quality loss modelled by video saliency. In our approach, we first use the method of [31] to predict video saliency map in HEVC compression domain. Then, perceptual quality is modelled, in which Mean Square Error (MSE) is weighted with the corresponding saliency values. Second, the SGCC formulation is proposed to minimize the loss of perceptual quality, when reducing HEVC decoding complexity to the target. Since DF and MC take up large proportions in the decoding time of HEVC, the decoding complexity is reduced in our SGCC formulation by disabling DF and simplifying MC for some non-salient CTUs. Third, the relationship between decoding complexity reduction and DF disabling/MC simplification is modelled for the SGCC formulation. Similarly, the influence of DF disabling/MC simplification on visual quality is also modelled. Finally, we develop a solution to the proposed SGCC formation, such that HEVC decoding complexity can be controlled to a target, while the perceptual quality is optimized.

Fig.1 shows an example of our SGCC approach. As seen in Fig.1, the video quality degrades along with the reduction of decoding complexity. However, when decoding complexity reduces, our SGCC approach preserves the visual quality of ROI (e.g., face), while the quality of non-ROI degrades. As such, the perceptual quality can be optimized by our SGCC approach, when reducing decoding complexity of HEVC.

To our best knowledge, our SGCC approach is the first work to reduce decoding complexity to a target (i.e., complexity control) for HEVC, and it is also the first one to minimize perceptual quality loss in decoding complexity reduction for HEVC. This paper is an extended version of our conference paper [32], with extensive advanced works summarized as follows. (1) We propose to simplify MC in our new SGCC optimization formulation, with the well modelled relationship among MC simplification, quality degradation and complexity reduction. As a result, the Maximal Achievable Reduction (MAR) of HEVC decoding can increase from ∼15% to ∼40%. (2) For the new SGCC optimization, an efficient solution is mathematically derived. (3) The performance of our SGCC approach is thoroughly evaluated with more test sequences, comparing approaches and evaluation metrics, than [32]. The code of our SGCC approach is available online: https://github.com/SGCCmaterials/SGCCcode.git. As HEVC normally has hierarchical coding structure, temporal scalability may be applied to save some decoding complexity, which drops some upper layer frames without decoding. Our SGCC approach can be combined with temporal scalability to achieve higher reduction of decoding complexity.
II. FORMULATION FOR SALIENCY-GUIDED COMPLEXITY CONTROL APPROACH

A. Preliminary

In [3], it has been verified that DF takes up 13%-27% of HEVC decoding complexity (13%-27% for x86 and 13%-20% for ARM). Hence, HEVC decoding complexity can be reduced by disabling DF of some CTUs. We define \( f_n \in \{0, 1\} \) to indicate whether the DF of the \( n \)-th CTU is enabled (\( f_n = 0 \)) or disabled (\( f_n = 1 \)). Given saliency value \( w_n \) of each CTU, we define \( \Delta C_n(f_n, w_n) \) as the decoding complexity reduction of a frame caused by disabling the DF of the \( n \)-th CTU. Note that, in this paper, the decoding complexity of HEVC is measured by the computational time on a Windows PC with Intel(R) Core(TM) i7-4790K CPU.

Also, [3] has investigated that MC consumes 35%-61% of HEVC decoding complexity (37%-61% for x86 and 35%-53% for ARM). Thus, simplifying MC is an effective way to reduce HEVC decoding complexity. In MC, each sample of a CTU is calculated according to the corresponding samples in the reference frames. To save decoding complexity, the MC step can be skipped for some prediction samples. Instead, these samples are reconstructed by Nearest Neighbor (NN) interpolation from neighboring prediction samples, which are generated by the original MC step. In our method, for the \( n \)-th CTU, \( g_n \in \{0, 1, 2, 3\} \) defines that \( g_n/4 \) of each sample is estimated by NN interpolation rather than applying MC. The remaining \((1 - g_n/4)\) of prediction samples are decoded with the original MC step, as the reference for NN interpolation. As a result, \( g_n \) implies the degree of simplifying MC. For example, \( g_n = 3 \) indicates the highest degree of the simplification, as \(3/4\) of total samples in the \( n \)-th CTU skip MC. \( \Delta C_M(g_n, w_n) \) is defined as the decoding complexity reduction of a frame, due to simplifying MC of the \( n \)-th CTU.

As the cost of decoding complexity reduction, the visual quality of decoded videos degrades (as Fig. 3 shows). Fortunately, it has been investigated [29] [30] that visual attention of the HVS does not focus on the whole picture, but only a small region around fixation (called foveal vision). Hence, the degraded quality may slightly influence the visual quality, by taking visual attention into account in our approach. In this paper, we use the compression domain saliency detection method proposed in [31] to directly obtain the saliency value of each CTU from HEVC bitstreams. Then, we follow [34] to weight the MSE of each CTU using its saliency value. We define \( w_n \) as the saliency value of the \( n \)-th CTU in a frame, and \( w_n \in [0, 1] \) always holds. Assuming that there are in total \( N \) CTUs in a video frame, the Saliency Weighted MSE (SW-MSE) of this frame is denoted by

\[
\Delta S_n(f_n, g_n, w_n) = \frac{\sum_{n=1}^{N} w_n \cdot \text{MSE}(f_n, g_n)}{\sum_{n=1}^{N} w_n}.
\]

In (1), \( \text{MSE}(f_n, g_n) \) is the MSE between CTUs decoded by original HEVC and by HEVC with our approach (when the parameters are \( f_n \) and \( g_n \)). Note that \( \Delta S_n(f_n = 0, g_n = 0, w_n) = 0 \), due to the fact that CTUs decoded by original HEVC are the same as those by our approach with \( f_n = 0 \) and \( g_n = 0 \). In the following, we focus on minimizing the SW-MSE when reducing decoding complexity. This way, the Quality of Experience (QoE) can be ensured.

B. Formulation for SGCC approach

Our SGCC approach aims at controlling the reduction of decoding complexity to the target, meanwhile minimizing perceptual quality loss (in terms of SW-MSE). Here, \( \Delta S_n(f_n, g_n, w_n) \) and \( \Delta C_n(f_n, g_n, w_n) \) are the SW-MSE and complexity reduction of the \( n \)-th CTU in a frame. \( \Delta C_T \) is the target of complexity reduction. The optimization formulation of SGCC can be expressed by

\[
\min_{\{f_n, g_n\}} \sum_{n=1}^{N} \Delta S_n(f_n, g_n, w_n) \quad \text{s.t.} \quad \sum_{n=1}^{N} \Delta C_n(f_n, g_n, w_n) \geq \Delta C_T,
\]

where \( N \) is the total number of CTUs in a frame.

Next, we discuss how to decompose \( \Delta C_n(f_n, g_n, w_n) \) and \( \Delta S_n(f_n, g_n, w_n) \) in our SGCC approach, which is the first step to solve the SGCC formulation of (2). For the decomposition, we have the following Observations.

**Observation 1:** \( \Delta C_D(f_n, w_n) \) and \( \Delta C_M(g_n, w_n) \) are almost independent with each other. Mathematically, it holds for

\[
\sum_{n=1}^{N} \Delta C_n(f_n, g_n, w_n) \approx \sum_{n=1}^{N} \left( \Delta C_D(f_n, w_n) + \Delta C_M(g_n, w_n) \right).
\]

**Analysis 1:** For [3], the error rate of complexity reduction can be measured:

\[
\Delta C_e = \frac{\sum_{n=1}^{N} \Delta C_n(f_n, g_n, w_n) - \sum_{n=1}^{N} \left( \Delta C_D(f_n, w_n) + \Delta C_M(g_n, w_n) \right)}{\sum_{n=1}^{N} \Delta C_n(f_n, g_n, w_n)}.
\]

If \( \Delta C_e \to 0 \), then [3] can be obtained. Here, Table IV reports \( \Delta C_e \) of decoding several videos at QP = 22, 27, 32 and 37. Note that the settings for decoding are the same as the experiments of Section V. As can be seen from Table IV, almost all average \( \Delta C_e \) is less than 1.5%. Thus, we can find \( \Delta C_e \to 0 \), and this verifies Observation 1.

---

1. Since saliency detection of [31] consumes high computational time, we replace the Support Vector Machine (SVM) in [31] by linear combination to simplify the method. As such, the computational time overhead of saliency detection is negligible, as presented in Section [28]. Note that the simplified method achieves similar performance as [31], with only 2% loss of Area Under receiver operating characteristic Curve (AUC) [33].
Observation 2: Assume that $\Delta S_D(f_n, w_n)$ and $\Delta S_M(g_n, w_n)$ are the SW-MSEs of disabling DF and simplifying MC, respectively. They are almost independent with each other. Mathematically, it holds for
\[
\sum_{n=1}^{N} \Delta S_n(f_n, g_n, w_n) = \sum_{n=1}^{N} (\Delta S_D(f_n, w_n) + \Delta S_M(g_n, w_n)).
\]

\[\text{Analysis 2: For [3], the error rate of SW-MSE can be measured:}
\[
\Delta S_e = \frac{\sum_{n=1}^{N} \Delta S_n(f_n, g_n, w_n)}{\sum_{n=1}^{N} \Delta S_n(f_n, g_n, w_n)}
\]

If $\Delta S_e \rightarrow 0$, [3] can be acquired. Table II tabulates $\Delta S_e$ of decoding several videos at different QPs. Note that the settings are the same as the experiments of Section V. We can see from Table II that most of average $\Delta S_e$ is less than 2.5%. Thus, we can conclude that $\Delta S_e \rightarrow 0$, and this verifies Observation 2.

Upon above two Observations, formulation [2] can be turned to
\[
\min_{(f_n, g_n)} \sum_{n=1}^{N} \left( \Delta S_D(f_n, w_n) + \Delta S_M(g_n, w_n) \right)
\]
\[\text{s.t. } \sum_{n=1}^{N} \left( \Delta C_D(f_n, w_n) + \Delta C_M(g_n, w_n) \right) \geq \Delta C_T.\]

Next, we move to learn the functions of $\Delta C_D(f_n, w_n)$, $\Delta C_M(g_n, w_n)$, $\Delta S_D(f_n, w_n)$ and $\Delta S_M(g_n, w_n)$, for solving our SGCC formulation.

III. RELATIONSHIP MODELLING FOR SGCC APPROACH
A. Relationship modelling for $\Delta S_D(f_n, w_n)$, $\Delta S_M(g_n, w_n)$

According to [1] and Observation 2, $\Delta S_D(f_n, w_n)$ and $\Delta S_M(g_n, w_n)$ can be represented by
\[
\Delta S_D(f_n, w_n) = \frac{w_n}{\sum_{n=1}^{N} w_n} \text{MSE}_D(f_n),
\]
\[
\Delta S_M(g_n, w_n) = \frac{w_n}{\sum_{n=1}^{N} w_n} \text{MSE}_M(g_n).
\]

In [8], $\text{MSE}_D(f_n)$ is defined as the MSE between CTUs, decoded by our approach with $f_n \in \{0, 1\}$ and by original HEVC (i.e., $f_n = 0$). Similarly, $\text{MSE}_M(g_n)$ is the MSE between the CTUs decoded by our approach with $g_n \in \{0, 1, 2, 3\}$ and by original HEVC (i.e., $g_n = 0$).

It is intractable to model $\text{MSE}_D(f_n)$ and $\text{MSE}_M(g_n)$ of [8], since they vary hugely across video content. However, we can use $w_n \text{MSE}_D(f_n=1)$ and $w_n \text{MSE}_M(g_n)$ instead of $w_n \text{MSE}_D(f_n)$ and $w_n \text{MSE}_M(g_n)$, respectively, since their correlation is rather high. Specifically, we evaluate the Spearman Rank Correlation Coefficient (SRCC) between $w_n \text{MSE}_D(f_n)$ and $w_n \text{MSE}_M(g_n)$ among all CTUs for each frame. The SRCC averaged over all frames of four training sequences is 0.92. Similarly, the averaged SRCC between $w_n \text{MSE}_D(g_n)$ and $w_n \text{MSE}_M(g_n)$ is 0.70. Consequently, on the basis of [8], the normalization can be written by
\[
\Delta \tilde{S}_D(f_n, w_n) = \frac{\Delta S_D(f_n, w_n)}{\Delta S_D(f_n=1, w_n=1)} = \frac{\text{MSE}_D(f_n)}{\text{MSE}_D(f_n=1)},
\]
and
\[
\Delta \tilde{S}_M(g_n, w_n) = \frac{\Delta S_M(g_n, w_n)}{\Delta S_M(g_n=3, w_n=1)} = \frac{\text{MSE}_M(g_n)}{\text{MSE}_M(g_n=3)}.
\]

First, we deal with the estimation on $\text{MSE}_D(f_n)$ and $\text{MSE}_M(g_n)$ for $\Delta \tilde{S}_D(f_n, w_n)$ and $\Delta \tilde{S}_M(g_n, w_n)$.

Second, we discuss on learning $\text{MSE}_M(g_n)$ from some training sequences. Four sequences, selected from JCT-VC database [15], are used for training, including two 1920 × 1080 sequences Cactus and BasketballDrive from Class B, as well as two 832 × 480 sequences BQMall and BasketballDrive from Class C. The sequences are compressed by HM 16.0 at four different QPs, i.e., $QP = 22, 27, 32$ and 37. All settings are the same as those in Section V.

Four training sequences (at $QP = 22, 27, 32$ and 37) are decoded with MC skipped for each four samples, corresponding to $g_n = 0, 1, 2, 3$. As such, 0, 1/4, 1/2 and 3/4 of total samples are skipped for MC in each training CTU. Accordingly, for a training sequence, the MSE caused by skipping MC can be estimated by
\[
\text{MSE}^*_M(g_n) = \frac{1}{L} \sum_{l=1}^{L} \|I_l(g_n = g_n) - I_l(g_n = 0)\|_2^2,
\]
where $I_l$ denotes the sample set of the $l$-th training CTU, and $L$ is the total CTU number in the training sequence. $P_l$ is the number of samples in the $l$-th training CTU, and $g_l$ denotes the proportion of its samples with MC skipped. Given [13], $\{\text{MSE}^*_M(g_n)\}_{g_n=0}^{3}$ can be obtained for each training sequence at one QP. Afterwards, $\text{MSE}^*_M(g_n)$ is normalized by $\text{MSE}^*_M(g_n=3)$. Based on the samples of $\text{MSE}^*_M(g_n=3)$ for...
all training sequences at four QPs, we utilize the least-
square fitting of the third-order polynomial regression to learn
MSE_M(g_n=3).

The fitting curve is shown in Fig. 3 each of which indicates a pair of \( (g_n, \text{MSE}_M^3(g_n=3)) \) for a training sequence at
one QP. Obviously, \( \text{MSE}_M^3(g_n=3) = 1 \) for \( g_n = 3 \) and
\( \text{MSE}_M^3(g_n=0) = 0 \) for \( g_n = 0 \), due to \( \text{MSE}_M^3(g_n = 0) = 0 \).
The R-square value of the fitting in Fig. 3 is 0.9980, verifying the
effectiveness of the fitting model. Finally, the learnt
polynomial function is as follows,

\[
\text{MSE}_M(g_n) = h_1 \cdot g_n^3 + h_2 \cdot g_n^2 + h_3 \cdot g_n,
\]

(14)
where the values of \( h_1, h_2 \) and \( h_3 \) are presented in Table III.

Consequently, (10) can be turned to

\[
\Delta S_M(g_n, w_n) = w_n \cdot (h_1 \cdot g_n^3 + h_2 \cdot g_n^2 + h_3 \cdot g_n),
\]

(15)

B. Relationship Modelling for \( \Delta C_D(f_n, w_n) \)

Now, we move to the modelling of \( \Delta C_D(f_n, w_n) \). Obvi-
osously, we have \( \Delta C_D(f_n = 0, w_n) = 0 \), as the decoding
complexity is not reduced when DF is enabled \( (f_n = 0) \) for
the \( n \)-th CTU. Next, we provide a way to learn \( \Delta C_D(f_n = 1, w_n) \).

For learning \( \Delta C_D(f_n = 1, w_n) \), four training sequences
at four QPs are decoded with DF enabled and disabled,
respectively. Then, for the \( l \)-th training CTU, the training sample \( \Delta C_D(f_l = 1, w_l) \) can be calculated as the percentage of
complexity reduction of a frame, after disabling the DF of
the \( l \)-th training CTU. Here, \( w_l \) is the saliency values of the
\( l \)-th training CTU.

We apply the least-square fitting of the linear regression to estimate \( \Delta C_D(f_n = 1, w_n) \) using the training data
\( \Delta C_D(f_l = 1, w_l) \). The fitting curves are plotted in Fig. 4.
Since \( \Delta C_D(f_n = 1, w_n) \) is the decoding complexity reduction
of a frame caused by disabling DF of the \( n \)-th CTU in this frame
(i.e., \( f_n = 1 \)), its value is also influenced by the total
number of CTUs in a frame. For example, in high resolution
videos, DF of one CTU occupies less decoding complexity
proportion of the whole frame, than that in lower resolution
videos. Such influence can be removed by multiplying \( N \), and
the function of \( N \Delta C_D(f_n = 1, w_n) \) for different resolutions
can be at the same scale and then trained together. For training
\( N \Delta C_D(f_n = 1, w_n), N \Delta C_D^*(f_l = 1, w_l) \) of 3,000 randomly
chosen CTUs are used as training samples, and each dot in Fig. 4
stands for one sample of \( (w_l, N \Delta C_D^*(f_l = 1, w_l)) \). Here,
the configurations of the encoder and decoder for training are
the same as those in experiments of Section V-A. Consequently,
the function of \( \Delta C_D(f_n, w_n) \) is

\[
\Delta C_D(f_n, w_n) = \frac{1}{N} \cdot (a \cdot w_n + b) \cdot f_n,
\]

(16)
where the values of \( a \) and \( b \) at different QPs are presented in
Table III. Finally, \( \Delta C_D(f_n, w_n) \) can be modelled.

C. Relationship Modelling for \( \Delta C_M(g_n, w_n) \)

Similarly, in order to model \( \Delta C_M(g_n, w_n) \), four training
sequences at four QPs are decoded with MC skipped for 0, 1, 2
and 3 samples among each four samples (i.e., \( g_n = 0, 1, 2, 3 \)).
The decoding complexity of each CTU is recorded for all
training sequences. Then, we define \( \Delta C_M^*(g_l, w_l) \) as the
percentage of complexity reduction of a frame, caused by
skipping MC of the \( l \)-th training CTU.

In Fig. 5, we plot the pairs of \( (w_l, N \Delta C_M^*(g_l = 1, w_l)) \) for
\( QP = 32 \), when decoding four training sequences at \( QP = 32 \).
Note that the dots in this figure indicate the pairs of \( (w_l, \Delta C_M^*(g_l = 1, w_l)) \) for 3,000 randomly selected CTUs,
with the same training configuration as Section V-B. Similar
results can be found for other values of \( g_n \) or other QPs.
Generally speaking, this figure indicates that \( \Delta C_M^*(g_l, w_l) \) is
independent of \( w_l \). Therefore, \( \Delta C_M(g_n, w_n) \) can be replaced by
\( \Delta C_M(g_n) \).

Next, we model \( \Delta C_M(g_n) \) by learning from training data
of \( \{\Delta C_M^*(g_l = g_n)\} \{g_n = 0, 1, 2, 3\} \). Sometimes, the CTU
number in each training video may be dramatically different,
such that the modeling of \( \Delta C_M^*(g_n, w_n) \) may bias toward
some of training video sequences. To avoid such bias, we can
estimate the averaged complexity reduction of each training
video sequence by

\[
\Delta C_M^*(g_n) = \frac{1}{L} \sum_{l=1}^{L} \Delta C_M^*(g_l = g_n),
\]

(17)
for each possible value of \( g_n \). Recall that \( L \) is the total
number of CTUs on the training sequences. Then, we have
\( \Delta C_M^*(g_n) \) for each training sequence at a specific QP. For
each case of a possible QP value (22, 27, 32 and 37), the
least-square fitting of the linear regression is applied on all
training data \( N \Delta C_M^*(g_n) \) of four training sequences.
Similar with modelling ∆C_D(f_n, w_n), we use N∆C_M(g_n) rather than ∆C_M(g_n) here to make the regression general for different resolutions. The fitting curves are plotted in Fig. 6. Accordingly, the function of ∆C_M(g_n) is obtained in the following,

$$\Delta C_M(g_n) = \frac{1}{N} \cdot c \cdot g_n,$$

where the values of c at different QPs are presented in Table III. Finally, ∆C_M(g_n) can be modelled. It is worth pointing out that the training sequences are encoded by hierarchical GOP structure, and the frame-level QP has the offset of 0 ~ +4 (encoder_randomaccess_main.cfg). For example, when setting QP = 22, its frame-level QP ranges from 22 to 26. Therefore, in our SGCC approach, the trained parameters a, b and c for QP = 22 are to be applied for frames with QP ranging from 22 to 26. Similar setting holds for QP = 27, 32 and 37.

IV. SOLUTION TO SGCC OPTIMIZATION FORMULATION

In this section, we concentrate on solving our SGCC formulation of (7), to achieve complexity control of HEVC decoding. Since Fig. 2 has shown that the loss of MSE caused by disabling DF is significantly less than that by simplifying MC, there exists ∆S_D(f_n, w_n) < ∆S_M(g_n, w_n). Thus, we can rewrite (7) of our SGCC formulation as (19).

As discussed in Section III-A, we replace ∆S_D(f_n, w_n) and ∆S_M(g_n, w_n) of (19) by their normalized functions ∆S_D(f_n, w_n) and ∆S_M(g_n, w_n). Then, given the relationship of (12), (15) and (16), formulation (19) can be finally turned to (20), where ∆C_T = ∆C_T - ∑_{n=1}^{N} \frac{1}{N} \cdot (a \cdot w_n + b).

$$\min \ \sum_{n=1}^{N} \Delta S_D(f_n, w_n) \ \text{s.t.} \ \sum_{n=1}^{N} \Delta C_D(f_n, w_n) \geq \Delta C_T,$$

$$\min \ \sum_{n=1}^{N} \Delta S_M(g_n, w_n) \ \text{s.t.} \ \sum_{n=1}^{N} \Delta C_M(g_n) \geq \Delta C_T - \sum_{n=1}^{N} \Delta C_D(f_n, w_n),$$

$$\min \ \sum_{n=1}^{N} w_n \cdot f_n \ \text{s.t.} \ \sum_{n=1}^{N} \frac{1}{N} \cdot (a \cdot w_n + b) \cdot f_n \geq \Delta C_T,$$

$$\min \ \sum_{n=1}^{N} w_n \cdot (h_1 \cdot g_n^3 + h_2 \cdot g_n^2 + h_3 \cdot g_n) \ \text{s.t.} \ \sum_{n=1}^{N} \frac{1}{N} \cdot c \cdot g_n \geq \Delta C_T.$$
Fig. 6. Fitting curves of $g_n$ versus $N \Delta C_M(g_n)$. Each dot indicates a pair of $(g_n, N \Delta C_M(g_n))$ for $g_n \in \{0, 1, 2, 3\}$.

Fig. 7. Framework of our SGCC approach.

B. Solution to formulation (20-b)

Next, we discuss the solution to formulation (20-b). First, (20-b) can be simplified by Lemma 4.

Lemma 4: The nonlinear integer programming (20-b) is equivalent to the linear integer programming problem as follows,

$$
\begin{align*}
\min_{N_1, N_2, N_3} & \quad N_1 \sum_{n=1}^{N_2} \left( \frac{a_1}{3} + \frac{a_2}{3} w_n + \frac{a_3}{3} \right) + \frac{N_1 + N_2 + N_3}{2} \sum_{n=N_3+1}^{N_4} \left( h_1 + h_2 + h_3 \right) \cdot w_n \\
\text{s.t.} & \quad \frac{1}{N} \cdot c \cdot (N_1 + 2N_2 + 3N_3) \geq \Delta C_T' 
\end{align*}
$$

In (26), $N_1$, $N_2$, and $N_3$ are the numbers of CTUs corresponding to $g_n = 1, 2$ and 3 in a frame, and they satisfy $N_1 + N_2 + N_3 \leq N$.

Proof 4: The proof for Lemma 4 is in Section II of the Supporting Document.

According to Lemma 4, the optimal solution to (20-b) can be obtained, once the formulation of (26) is worked out. In fact, (26) is a linear programming problem, which can be solved by the branch-and-bound algorithm [36]. However, the computational complexity of the solution is still enormous, especially for large CTU number $N$ in a frame with high resolution. It is because the branch-and-bound algorithm has to be carried out to solve (26) for each frame. Next, we further simplify (26) to reduce its computational complexity.

Proposition 5: $\bar{w}_n$ is of almost uniform distribution as follows,

$$
\forall N_1 \in \{n\}_{n=1}^{N}, \quad \sum_{n=1}^{N_1} \bar{w}_n \approx k \cdot N_1^2,
$$

where $k$ is a positive constant.

Proof 5: The proof for Proposition 5 is in Section III of the Supporting Document.

Based on Proposition 5, (26) can be rewritten by

$$
\begin{align*}
\min_{g_n, N_3, N_2, N_1} & \quad N_3^2 + (8h_1 + 4h_2 + 2h_3) \cdot ((N_1 + N_2 + N_3)^2 - N_1^2) \\
& \quad + (h_1 + h_2 + h_3) \cdot ((N_1 + N_2 + N_3)^2 - (N_2 + N_3)^2) \\
\text{s.t.} & \quad \frac{1}{N} \cdot c \cdot (N_1 + 2N_2 + 3N_3) \geq \Delta C_T' 
\end{align*}
$$

Note that $k$ is a constant which is independent of the minimization problem in (28), and thus $k$ can be simply removed from the minimization formulation.

Next, we apply the branch-and-bound algorithm to solve (25), and it only needs to be solved once before decoding. We establish a table for the solutions to (28) at each specific $\Delta C_T'$. Then, given $\Delta C_T'$, we can simply obtain $N_3', N_2$ and $N_1$ by table look-up. This way, the overhead of computational complexity on solving (20-b) can be avoided.

An example of $\{f_n\}_{n=1}^{N}$ and $\{g_n\}_{n=1}^{N}$ solved by our SGCC approach is shown in Fig. 8(c) and (d). As can be seen, larger $f_n$ or $g_n$ corresponds to smaller $w_n$, which is the saliency value as illustrated in Fig. 8(b). The detected saliency map, shown in Fig. 8(b), tallies well with the groundtruth, i.e., the fixation map in Fig. 8(a). As a result, the decoding complexity of CTUs in non-ROI is reduced in high priority, and the quality of ROI (e.g., face) rarely degrades. This indicates that the perceptual quality loss can be minimized in applying our SGCC approach.

C. Error propagation analysis

The quality loss of each decoded frame, which is caused by the above complexity control, may propagate to other frames predicted by this frame. Hence, it is necessary to analyze the error propagation across decoded frames. We find through the following observations that the hierarchical coding structure of HEVC can significantly alleviate the error propagation in our SGCC approach. Here, for analysis, we use the hierarchical GOP structure of the default HM Random Access (RA) with encoder_randomaccess_main.cfg file, as shown in Fig. 9. Similar results can be found for other GOP structure.

Observation 6: The quality loss of I frames does not incur any error propagation, when reducing decoding complexity by our SGCC approach.
Fig. 9. GOP structure and its hierarchical layers.

Fig. 10. Averaged propagation error of each frame at QP = 32 and $\Delta C_T = 20\%$, in terms PSNR reduction, along with Picture Order Count (POC).

**Analysis 6:** The reconstruction of I-frames is independent of other frames, and thus the quality loss of other frames has no impact on each decoded I frames. We further tested the error propagation of two neighboring I frames and four GOP between them (from frame 32 to frame 64), averaged over four training sequences. Here, the error propagation of the i-th frame is calculated as follows. First, we only apply our SGCC approach on frame i, and do not make any complexity reduction on other frames. Then, the quality of the i-th frame is evaluated by Y-PSNR in dB. For the anchor, we apply our SGCC approach on all frames, and also measure the quality of the i-th frame by Y-PSNR. Finally, the difference of above two PSNRs is calculated as the error propagation at frame i. The results are shown in Fig. 10 and we find that the PSNR reduction of each I frame is 0 dB.

Additionally, the quality loss of I frames does not incur any error propagation within the frame for our SGCC approach, as only intra prediction mode is applied in I frames. In the I-frames decoding, the DF is implemented in every frame after the reconstruction (intra prediction, etc.) of the whole frame [1]. Thus, the quality degradation caused by disabling DF cannot propagate through intra prediction among CTUs. Furthermore, since MC is only related to inter frame prediction, the error cannot propagate within I frame. This completes the validation of Observation 6.

**Observation 7:** The quality loss of B or P frames incurs small error propagation due to the hierarchical coding structure in HEVC, when reducing decoding complexity by our SGCC approach.

**Analysis 7:** In B or P frames, because of inter prediction, the quality degradation of the reference frames is possible to propagate to the currently decoded frame. Thus, error propagation exists in B or P frames. However, the error propagation is restricted to be small by the hierarchical GOP structure. First, each I frame does not incur any error propagation as pointed out by Observation 6. In addition, I frames do not have MC, such that their quality loss is only from disabling DF, which is significantly lower than that of simplifying MC (see Fig. 2). As a result, after I frames, the error propagation of B or P frames terminates, and their quality loss is resumed to be small. Second, although the B or P frames, especially in higher layers or far from I frames, suffer from error propagation, the reference frames at different layers of hierarchical coding structure ensure (see Fig. 9) that each decoded frame is predicted by several frames. Specifically, all frames of the first GOP after I frames are all predicted by I frames, which has little quality loss. Then, for the second GOP all frames have the reference frame directly predicted by I frames, such that the shortest prediction path to I frames is one frame. The shortest prediction path to I frames is two frames for the third GOP, and so on. Note that the error propagation of the frames of the GOP before an I frame can be reduced to be small, as they are also predicted by the incoming I frame. Therefore, in the hierarchical coding structure of HEVC, there exists small error propagation for the quality loss of B/P frames.

In addition, Fig. 10 shows the error propagation of all B frames between two neighboring frames averaged over four training sequences, when $\Delta C_T = 20\%$ and QP = 32. As shown, the averaged error propagation of B frames is only 0.19 dB. Thereby, we can conclude that the error propagation of quality loss for B or P frames is rather small. Finally, the analysis of Observation 7 is completed.

Note that when applying our SGCC approach, I frames should be inserted to terminate error propagation, according to Observation 6. In this paper, the period between I frames is set as 32 for training and test sequences, according to the default configuration of encoder_randomaccess_main.cfg.

**D. Complexity overhead analysis**

Finally, we analyze the complexity overhead in applying our SGCC approach. The complexity overhead of our SGCC approach includes calculating $\{w_n\}$, computation on (20-a) and (20-b). Their computational time is evaluated and reported in Table IV. Note that the function QueryPerformanceCounter() in Visual C++ was used to record the computational time. The experiment was performed on a Windows PC with Inter(R) Core(TM) i7-4790K CPU.

It can be seen from Table IV that the complexity overhead of our SGCC approach is rather small. In particular, calculating saliency values $\{w_n\}$ consumes averagely 0.058 ms per 1080p frame. When calculating (20-a), saliency values $\{w_n\}^N_{n=1}$ need to be sorted as $\{\tilde{w}_n\}^N_{n=1}$ by the quicksort algorithm, which averagely consumes 0.010 ms per 1080p frame. Besides, computing $I$ in (25) consumes averagely 0.001 ms per frame for 1080p videos for solving (20-a). For solving (22)-b, as mentioned in Section IV-B, we establish a look-up table for the solutions to (22)-b, and we can simply obtain the solution by the table look-up, when decoding HEVC bitstreams. Therefore, the computational time of (22)-b is only for reading the values $N_1$, $N_2$ and $N_3$ from the table,
TABLE V

| Classes | Sequences | QP = 22 | QP = 27 | QP = 32 | QP = 37 |
|---------|-----------|---------|---------|---------|---------|
|         |           | ∆Cp (%) | ∆Cp (%) | ∆Cp (%) | ∆Cp (%) |
| | | 10 | 20 | 30 | 40 | 10 | 20 | 30 | 40 | 10 | 20 | 30 | 40 | 10 | 20 | 30 | 40 |
| A Traffic | +0.81 | +0.46 | +0.97 | +3.01 | +7.30 | -0.19 | +0.78 | +4.14 | +0.44 | -0.29 | +2.20 | +4.29 |
| B ParkScene | -1.30 | -0.32 | +1.57 | +0.94 | +6.80 | +0.50 | -0.45 | +3.67 | +1.62 | -1.43 | +1.89 | +4.81 |
| C RaceHorses | -0.72 | +2.14 | +1.47 | +1.33 | +6.02 | +3.01 | +2.71 | +3.35 | +3.22 | +2.96 | +3.37 | +6.09 |
| D BQSquare | -1.78 | -3.79 | -1.21 | -2.15 | 0.91 | -2.41 | -1.80 | -2.63 | -0.57 | -1.49 | -1.79 | -2.46 |

V. EXPERIMENTAL RESULTS

In this section, experimental results are presented to validate the effectiveness of our SGCC approach, in comparison with the latest HEVC decoding complexity reduction approaches [23] and [24].

A. Settings

All 15 sequences of Classes A-D (except 10-bit sequences) from the JCT-VC database [35] were divided into non-overlapping training and test sets. Four sequences were selected as the training set to learn the relationship of Section III. Then, we tested our approach on the remaining sequences, including two 2560 × 1600 sequences Traffic and PeopleOnStreet from Class A, three 1920 × 1080 sequences Kimono, ParkScene and BQTerrace from Class B, two 832 × 480 sequences RaceHorses and PartyScene from Class C, and four 416 × 240 sequences RaceHorses, BQSquare, BlowingBubbles and BasketballPass from Class D. First, all tested sequences were encoded by the HM 16.0 encoder. Here, the configuration of RA was implemented with GOP size being 8. Four common QPs, i.e., 22, 27, 32 and 37, were chosen to encode the test sequences. All other parameters were set by default in the encoder, using the encoder_randomaccess_main.cfg file.

Besides, HM 16.0 with its default settings was also utilized as the decoder. In our experiments, our SGCC approach is implemented in the HM platform, the same as most of existing HEVC complexity reduction works [5]–[10]. Compared to encoding, HM is more practical in decoding, since our experiments found that it is able to achieve real-time decoding for 1080p videos at 24 fps and QP = 37 on a Windows PC with Inter(R) Core(TM) i7-4790K CPU.

The experiments were all performed on a Windows PC with Inter(R) Core(TM) i7-4790K CPU and 32G RAM. To evaluate visual quality, both Y-PSNR difference (ΔPSNR) and Eye-tracking Weighted Y-PSNR difference (ΔEW-PSNR) [37] are assessed. Here, Y-PSNR and EW-PSNR are calculated upon the raw and decoded sequences. Then, ΔPSNR and ΔEW-PSNR quantify the PSNR and EW-PSNR degradation, when decoding sequences by HEVC with our SGCC [23] and [24] approaches, instead of the original HEVC decoder. As such, the smaller ΔPSNR and ΔEW-PSNR indicate better performance in quality loss. In calculating ΔEW-PSNR, we utilize human fixation maps from eye-tracking experiment to weight MSE, for fair comparison. In addition, the results of the Difference Mean Opinion Score (DMOS) [38] are also measured to assess the subjective quality of decoding sequences.

B. Evaluation on control performance

First of all, we evaluate the control performance of our SGCC approach in HEVC decoding. The performance evaluation consists of two parts: Maximal Achievable Reduction (MAR) and control error. First, we compare the MAR results of our SGCC approach with those of [23] and [24]. Here, to obtain MAR of our SGCC approach, we set $f_n = 1$ and $g_n = 3$ for all the CTUs to achieve the maximal decoding complexity reduction. Then, we record the ratio of such reduction as MAR. For [23] and [24], we also make their complexity reduction reach maximal values using the ways reported in [23] and [24]. Note that the complexity overhead of our approach (analyzed in Section IV-D), which is far less than HEVC decoding complexity, is included for evaluation.

MAR: Fig. 11 demonstrates the mean and standard deviation of MARs for our SGCC, [23] and [24] approaches.

![Fig. 11. Mean and standard deviation of MARs for our SGCC, [23] and [24] approaches.](image-url)
| Class | Sequence | Aprr. | QP=32 | MRE=10% | QP=22 | MRE=20% | PSNR | QEWS-PSNR | \(\Delta C_T\) = 10% | \(\Delta C_T\) = 20% | Error (MRE) for each specific complexity reduction target (i.e., \(\Delta C_T\) = 10% and 20%) | Mean Absolute Error (MAE) and Mean Relative Error (MRE) across different complexity reduction targets (i.e., \(\Delta C_T\)) with \([23]\) and \([24]\) in control error, since \([23]\) and \([24]\) approach increases to 1.7644dB. It is because MC simplification of (20-b) brings in larger distortion, in comparison with DF disabling of (20-a). It can be further seen from Table VI that our SGCC approach significantly outperforms \([23]\) and \([24]\) in terms of \(\Delta\)PSNR, especially at high complexity reduction. Specifically, once decoding complexity reduction increases to 23%, \([24]\) incurs averagely 7.8504dB Y-PSNR loss at QP = 32, far more than 1.7644dB of our SGCC approach. Besides, \([23]\) is incapable of reducing decoding complexity of HEVC to 23%. Despite much better than \([23]\) and \([24]\), the objective quality loss of our method is not very small at high complexity reduction (e.g., \(\Delta\)PSNR = 1.7644 dB at 23% reduction and \(\Delta C_T\) = 32). However, the perceptual quality loss by our method can be alleviated (e.g., \(\Delta\)EW-PSNR = 1.1428 dB at 23% complexity reduction). Since \([23]\) and \([24]\) cannot control decoding complexity reduction, we were not able to set complexity reduction target \(\Delta C_T\) in \([23]\) and \([24]\). Instead, we first decoded the test sequences with \([23]\) and \([24]\), and we found that their complexity reduction is around some specific values, e.g., 5%, 10% and 20% at QP = 32, and 8%, 18% and 23% at QP = 32. Then, we set \(\Delta C_T\) of our SGCC approach to these values for fair comparison.

\[
\text{MRE} = \frac{\text{MAE}}{\Delta C_T} \times 100\%.
\]

(29)
Fig. 12. ∆PSNR and ∆EW-PSNR versus decoding complexity reduction at QP = 27 and 32.

Fig. 13. The frame-level ∆PSNR and ∆EW-PSNR when ∆C_T = 23% and QP = 32.

PartyScene, QP = 27
PartyScene, QP = 32
BQTerrace, QP = 27
BQTerrace, QP = 32

D. Assessment on fluctuation of quality loss

Next, we assess the frame-level fluctuation of quality loss caused by our SGCC approach, since the error propagation of our approach may increase the fluctuation of quality loss. Fig. 13 plots the objective and perceptual quality loss along with decoded frames at ∆C_T = 23% and QP = 32, averaged over all 11 test sequences. First, it can be seen that I-frames have slight quality loss, which incur no error propagation. More importantly, the quality loss can be resumed to be near zero successively after I frames, validating the effectiveness of I frames in preventing error propagation of quality loss. This is in accordance with Observation 6. Second, the quality degradation of the frames at the first layer is less than that at upper layers, within a GOP. As such, the fluctuation of quality loss can be relieved. This indicates the small error propagation of our approach due to the hierarchical coding structure of HEVC, satisfying Observation 7. Finally, one may see that the range of ∆EW-PSNR (0.5-1.7 dB) is much smaller than that of ∆PSNR (1-2.5 dB), for non-I frames. Thus, it verifies that the perceptual quality loss of our SGCC approach has less fluctuation, compared with objective quality loss.

E. Assessment on subjective quality

We further assess the subjective quality of our SGCC approach compared with [23] and [24]. In our experiment, the DMOS test was conducted to rate subjective quality of the decoded sequences, by the means of Single Stimulus Continuous Quality Evaluation (SSCQE), which is processed...
Table VII: DMOS Values at QP = 32 of SGCC, [23] and [24].

| ∆C_T | Sequences | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | 11  | Average |
|-------|-----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------|
| 8%    | SGCC      | 33.35 | 34.75 | 42.41 | 37.50 | 45.03 | 53.54 | 41.30 | 52.06 | 33.20 | 37.07 | 33.05 | 41.11 |
|       | [23]      | 40.79 | 44.28 | 44.25 | 45.47 | 45.79 | 45.97 | 48.40 | 46.20 | 49.97 | 37.90 | 41.10 | 44.70  |
|       | [24]      | 47.26 | 36.65 | 44.08 | 45.27 | 47.69 | 47.97 | 46.88 | 46.07 | 35.07 | 42.25 | 41.55 | 43.74  |
| 23%   | SGCC      | 45.98 | 47.23 | 61.06 | 43.33 | 54.81 | 54.43 | 55.07 | 60.83 | 63.06 | 64.37 | 48.22 | 54.40  |
|       | [23]      | 47.97 | 47.69 | 43.74 | 37.07 | 54.81 | 63.09 | 66.11 | 46.07 | 41.30 | 44.28 | 43.75 | 41.76  |
|       | [24]      | 43.33 | 3 | 47.26 | 68.08 | 57.50 | 44.70 | 47.23 | 44.68 | 70.11 | 45.79 | 45.03 | 33.05  |
|       | T         | 8%   | 23%  | T    | T    | T    | T    | T    | T    | T    | T    | T    | T       |

1: Traffic 2: PeopleOnStreet 3: ParkScene 4: BQTerrace 5: Kimono 6: RaceHorses (832 × 480)
7: PartyScene 8: RaceHorses (416 × 240) 9: BQSquare 10: BlowingBubbles 11: BasketballPass

This paper has proposed a decoding complexity control approach (namely SGCC) for HEVC, aiming to reduce HEVC.

by Rec. ITU-R BT.500 [38]. During the test sequences were displayed in random order. After viewing each decoded sequence, the subjects were asked to rate the sequence. As a result, DMOS value of each decoded sequence can be calculated to measure the difference of subjective quality between sequences decoded by original HEVC and by HEVC with our SGCC approach or other conventional approaches [23] and [24].

Table VII shows the DMOS values of three approaches for all test sequences, with complexity reduction being approximately 8% and 23%. Note that the smaller values of DMOS mean the better subjective quality, since DMOS quantifies the subjective quality difference between the uncompressed and compressed sequences. Obviously, when complexity reduction is around 8%, our SGCC approach has smaller DMOS values than [23] and [24] for 8 among 11 test sequences. Besides, the averaged DMOS value of our SGCC approach is smallest among all three approaches at ∆C_T = 8%. Once decoding complexity is further decreased to 23%, our SGCC approach is greatly superior to [24] for all 11 test sequences, in terms of DMOS. Recall that decoding complexity reduction of [23] cannot arrive at 23%, and we thus only compare with [24] for ∆C_T = 23% in Table VII.

Furthermore, Fig. 14 shows some frames of two selected sequences, decoded by HEVC with our SGCC, [23] and [24] approaches, at ∆C_T = 8% and ∆C_T = 23%. The MSEs of ROI in the four selected frames are given. The MSEs of our SGCC approach are significantly smaller than those of [23] and [24].

![Image of subjective quality with ROI highlighted](image_url)

(a) Subjective quality at ∆C_T = 8%, QP = 32

(b) Subjective quality at ∆C_T = 23%, QP = 32

F. Performance on HEVC bitstreams with rate control

In real applications, the HEVC encoder usually enables rate control, so that QP may vary within a sequence. We further implemented our SGCC approach on HEVC bitstreams with rate control enabling. Here, parameters a, b and c are chosen according to the range of frame-level QP, as discussed in Section III-C. The results are shown in Table VIII. It can be seen that the complexity control accuracy of decoding sequences with rate control is comparable to that without rate control (Table VII). Note that we follow the most recent rate control work of [39] to set the target bit rates the same as the actual bit rates at fixed QP (22, 27, 32 and 37), and those target bit rates are denoted by Bit rates 1-4 in Table VIII.

VI. Conclusion

This paper has proposed a decoding complexity control approach (namely SGCC) for HEVC, aiming to reduce HEVC.
decoding complexity to a target with minimal loss on perceptual quality. We found two ways to reduce the decoding complexity of some CTUs: (1) disabling DF and (2) simplifying MC. However, disabling DF or simplifying MC may cause some visual quality loss in decoded videos. Thus, the SGCC formulation was proposed to reduce HEVC decoding complexity to the target, meanwhile minimizing perceptual quality loss. In this paper, perceptual quality loss was evaluated on the basis of video saliency. For our formulation, the least square fitting on training data was applied to model the relationship between complexity reduction/qulity loss and DF disabling/MC simplification. Finally, a potential solution to the proposed formulation was developed, such that SGCC can be accomplished for HEVC decoding. As verified in experimental results, our SGCC approach is efficient in complexity control for HEVC decoding, evaluated in control performance, complexity-distortion performance, fluctuation of quality loss, and subjective quality.

Our work in current form is implemented on HEVC RA bitstreams with hierarchical and open GOP structure. It is an interesting future work to apply our work on other settings, like close GOP structure or LD scenario. Moreover, in current stage, our SGCC approach only concentrates on LCU level complexity control for HEVC decoding. It is another promising future work to control HEVC decoding complexity at frame level, which may make our SGCC approach more flexible for controlling decoding complexity.

REFERENCES

[1] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, “Overview of the High Efficiency Video Coding (HEVC) standard,” IEEE TCSVT, pp. 1649–1668, 2012.

[2] T. K. Tan, R. Wachnik, M. Krak, N. Ramzan, V. Baroncini, J. R. Ohm, and G. J. Sullivan, “Video quality evaluation methodology and verification testing of HEVC compression performance,” IEEE TCSVT, pp. 76–90, Jan 2016.

[3] F. Bossen, B. Bross, K. Suhring, and D. Flynn, “HEVC complexity and implementation analysis,” IEEE TCSVT, pp. 1685–1696, 2012.

[4] G. Correa, P. A. Assuncão, L. Volcan Agostini, and L. A. da Silva Cruz, “Fast HEVC encoding decisions using data mining,” IEEE TCSVT, pp. 663–673, 2015.

[5] H. L. Tan, C. C. Ko, and S. Rahardja, “Fast coding quad-tree decisions using prediction residuals statistics for high efficiency video coding (HEVC),” IEEE Transactions on Broadcasting, pp. 128–133, 2016.

[6] J. Vanne, M. Viitanen, and T. D. Hamalainen, “Efficient mode decision schemes for hevc inter prediction,” IEEE TCSVT, pp. 1579–1593, 2014.

[7] Z. Pan, J. Lei, Y. Zhang, X. Sun, and S. Kwong, “Fast motion estimation based on content property for low-complexity h.265/HEVC encoder,” IEEE Transactions on Broadcasting, pp. 675–684, 2016.

[8] X. Deng, M. Xu, S. Li, and Z. Wang, “Complexity control of hevc based on region-of-interest attention model,” in VCIP, 2014.

[9] Y. Zhang, S. Kwong, X. Wang, H. Yuan, Z. Pan, and L. Xu, “Machine learning-based coding unit depth decisions for flexible complexity allocation in high efficiency video coding,” IEEE TIP, pp. 2225–2238, 2015.

[10] X. Deng, M. Xu, L. Jiang, X. Sun, and Z. Wang, “Subjective-driven complexity control approach for HEVC,” IEEE TCSVT, 2015.

TABLE VIII
COMPLEXITY CONTROL ERROR (%) OF OUR SGCC APPROACH FOR HEVC BITSTREAMS WITH RATE CONTROL.

| ΔC/P | Bit rate 1 | Bit rate 2 | Bit rate 3 | Bit rate 4 |
|------|------------|------------|------------|------------|
|      | 10         | 20         | 10         | 20         |
| MAE  | 1.86       | 3.19       | 1.23       | 1.91       |
| MRE  | 18.0       | 15.9       | 12.3       | 9.53       |

[11] D. Dubey, A. Amritphale, A. Sawhney, D. Dubey, and N. Srivastav, “Analysis of youtube as a source of information for west nile virus infection,” Clinical medicine & research, pp. 129–132, 2014.

[12] Everymac.com, “What are the main differences between the ipad and the macbook air?” 2014, http://www.everymac.com/.

[13] Z. Ma, H. Hu, and Y. Wang, “On complexity modeling of h.264/avc video decoding and its application for energy efficient decoding,” IEEE TMM, pp. 1240–1255, 2011.

[14] Y. Liu, Z. G. Li, and Y. C. Soh, “Region-of-interest based resource allocation for conversational video communication of H.264/AVC,” IEEE TCSVT, pp. 134–139, 2008.

[15] L. Yan, Y. Duan, J. Sun, and Z. Guo, “Implementation of HEVC decoder on x86 processors with SIMD optimization,” in VCIP, 2012.

[16] C. C. Chi, M. Alvarez-Mesa, B. Bross, B. Juurlink, and T. Schierl, “SIMD acceleration for HEVC decoding,” IEEE TCSVT, 2014.

[17] D. F. de Souza, A. Ilic, N. Roma, and L. Sousa, “Towards GPU HEVC intra decoding: Seizing fine-grain parallelism,” in ICME, 2015.

[18] C. C. Chi, M. Alvarez-Mesa, B. Juurlink, G. Clare, F. Henry, S. Pateux, and T. Schierl, “Parallel scalability and efficiency of HEVC parallelization approaches,” IEEE TCSVT, pp. 1827–1838, 2012.

[19] M. Alvarez-Mesa, C. C. Chi, B. Juurlink, V. George, and T. Schierl, “Parallel video decoding in the emerging HEVC standard,” in ICASSP, 2012.

[20] E. Kalali, Y. Adibelli, and I. Hamzaoglu, “A high performance and low energy intra prediction hardware for hevc video decoding,” in DASIP, 2012.

[21] M. Naccari, C. Brites, J. Ascenso, and F. Pereira, “Low complexity deblocking filter perceptual optimization for the HEVC codec,” in ICIP, 2011.

[22] C. Feldmann, F. Jager, and M. Wien, “Decoder complexity reduction for the scalable extension of HEVC,” in ICIP, 2014.

[23] E. Nogues, S. Holmbacka, M. Pelcat, D. Menard, and J. Lilius, “Power-aware HEVC decoding with tunable image quality,” in SIPS, 2014.

[24] E. Nogues, E. Raffin, M. Pelcat, and D. Menard, “A modified HEVC decoder for low power decoding,” in ACM CF, 2015.

[25] C. C. Chi, M. Alvarez-Mesa, B. Juurlink, V. George, and T. Schierl, “Improving the parallelization efficiency of hevc decoding,” in ICIP, 2012.

[26] E. Nogues, R. Berrada, M. Pelcat, D. Menard, and E. Raffin, “A DVFS based HEVC decoder for energy-efficient software implementation on embedded processors,” in ICME, 2015.

[27] M. J. Langroodi, J. Peters, and S. Shirmohammadi, “Decoder-complexity-aware encoding of motion compensation for multiple heterogeneous receivers,” ACM TOMP, pp. 46, 2015.

[28] C. t. Blakemore and F. Campbell, “On the existence of neurons in the human visual system selectively sensitive to the orientation and size of retinal images,” The Journal of physiology, p. 237, 1969.

[29] W. S. Geisler and J. S. Perry, “Real-time foveated multisiresolution system for low-bandwidth video communication,” in Photonics West’98 Electronic Imaging. International Society for Optics and Photonics, 1998, pp. 294–305.

[30] Z. Wang and A. C. Bovik, “Embedded foveation image coding,” IEEE TIP, pp. 1397–1410, 2001.

[31] M. Xu, L. Jiang, X. Sun, Z. Ye, and Z. Wang, “Learning to detect video saliency with HEVC features,” IEEE TIP, vol. 26, no. 1, pp. 369–385, 2017.

[32] R. Yang, M. Xu, L. Jiang, and Z. Wang, “Subjective-quality-optimized complexity control for HEVC decoding,” in ICME, 2016.

[33] A. Borji and I. Laurent, “State-of-the-art in visual attention modeling,” IEEE TIPAMI, vol. 35, no. 1, pp. 184–207, Jan. 2013.

[34] Z. Wang and Q. Li, “Information content weighting for perceptual image quality assessment,” IEEE TIP, pp. 1185–1198, 2011.

[35] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, “Comparison of the coding efficiency of video coding standards including high efficiency video coding (HEVC),” IEEE TCSVT, pp. 1669–1684, 2012.

[36] D. Li and X. Sun, Nonlinear integer programming. Springer Science & Business Media, 2006.

[37] Z. Li, S. Qin, and L. Itti, “Visual attention guided bit allocation in video compression,” Image and Vision Computing, vol. 29, no. 1, pp. 1–14, 2011.

[38] I. Recommendation, “500-11.methodology for the subjective assessment of the quality of television pictures, Recommendation ITU-R BT. 500-11,” ITU Telecom. Standardization Sector of ITU, vol. 7, 2002.

[39] B. Li, H. Li, L. Li, and J. Zhang, “A haptic domain rate control algorithm for high efficiency video coding,” IEEE TIP, pp. 3841–3854, 2014.