Complexity-rate-distortion Evaluation of Video Encoding for Cloud Media Computing

Ming Yang, Jianfei Cai, Yonggang Wen and Chuan Heng Foh
School of Computer Engineering, Nanyang Technological University, Singapore 639798
Email: \{yang0258, asjfcai, ygwen, aschfoh\}@ntu.edu.sg

Abstract—Cloud computing provides a promising solution to cope with the increased complexity in new video compression standards and the increased data volume in video source. It not only saves the cost of too frequent equipment upgrading but also gives individual users the flexibilities to choose the amount of computing according to their needs. To facilitate cloud computing for real-time video encoding, in this paper we evaluate the amount of computing resource needed for H.264 and H.264 SVC encoding. We focus on evaluating the complexity-rate-distortion (C-R-D) relationship with a fixed encoding process but under different external configuration parameters. We believe such an empirical study is meaningful for eventually realizing real-time video encoding in an optimal way in a cloud environment.

I. INTRODUCTION

With decades of development, video coding technology has become very mature. Various video compression standards have been developed, including the ITU-T H.26x series and the MPEG series. All these compression standards are driven either by achieving better rate-distortion (R-D) performance or by meeting the requirements of diverse media applications. The latest video coding standard, H.264/AVC [1], reports the best R-D performance among the standard codecs, and its scalable video coding (SVC) extension, H.264 SVC [2], provides great flexibility to adapt to network dynamics and diverse scenarios while still maintaining decent R-D performance.

However, a codec with excellent R-D performance does not guarantee its success in practical applications, where complexity of the codec must be considered. For many practical applications such as IPTV, video conference, video surveillance and video recording, real-time video encoding is required. It is well-known that real-time video encoding is not easy to achieve due to the high complexity of video encoding, especially for H.264 and H.264 SVC, where the good R-D performance is obtained at the cost of greatly increased complexity. Moreover, with the reduced price and the increased capability of digital cameras, we can see the trend of capturing videos at higher resolution, higher frame rate and higher quality. This makes the real-time video encoding even more challenging. To achieve real-time encoding, adopting or upgrading to more powerful processing machines or devices is a common solution to cope with the increased complexity in video compression and the increased data volume in video source. This is definitely not an economical-efficient and environment-friendly solution.

The recently emerged concept of cloud computing provides a promising solution to the above mentioned dilemma. Cloud computing facilitates the borrowing of computing power from others distributed processors. It is an ideal solution for dynamic media processing with increased computation requirement [3]. It not only saves the cost of too frequent equipment upgrading but also gives individual users the flexibilities to choose the amount of computing according to their needs.

To utilize cloud computing for real-time video encoding, a fundamental question we need to answer is how much computing resource is needed for real-time video encoding, which requires a complexity-rate-distortion (C-R-D) analysis of video encoding process. Unlike R-D analysis, which has been well studied in literature, C-R-D analysis is still in its infant stage. There are only a few studies [4], [5] on C-R-D analysis. Most of them focus on how to design an optimal video encoder so as to maximize the R-D performance given a certain complexity or power constraint.

Unlike the existing C-R-D analysis, our considered cloud-assisted video encoding scenario is not interested in modifying the video encoding process to meet a certain complexity or energy constraint. Instead, we are interested in the C-R-D relationship with a fixed encoding process but under different external configuration parameters. This paper presents our preliminary attempts to evaluate the C-R-D performance of the H.264/AVC encoding and the H.264 SVC encoding. For H.264 SVC encoding, we evaluate the C-R-D performance of individual scalability as well as multiple scalability. We believe such an empirical study is meaningful for eventually realizing real-time video encoding in an optimal way in a cloud environment.

II. RELATED WORK AND DEFINITIONS

In this section, we introduce some background information, discuss some related work, and give the definitions of complexity, rate and distortion used in our study.

A. Overview of H.264 SVC

The scalable video coding (SVC) extension of H.264/AVC has been standardized to facilitate the easy adaptation of video streams for the variety of requirements from storage devices, terminals, communication networks and user preference. H.264 SVC [2], typically consisting of one base layer and multiple enhancement layers, provides temporal, quality and spatial scalability at the cost of lower rate-distortion (R-D) performance. Fig. 1 shows the hierarchical prediction structure used of H.264 SVC in one GoP, which facilitates temporal scalability and also serves as an independent coding unit.
B. Cloud-assisted Video Encoding

The computational resource in cloud platform is commonly organized as groups of virtual machines (VMs), which each VM is assumed to be able to perform computing independently. When performing video encoding in cloud, the encoding process needs to be represented in a cloud-friendly way, i.e. being decomposed into individual parallel processing units. A common way is to decompose a raw video into non-overlapping coding-independent groups of pictures (GOPs) and dispatch GOPs to individual VMs. For shorter encoding delay and the match between individual coding units and VMs, a finer parallelization granularity might be needed, which can be obtained through macro-block (MB) level parallelization.

Recently, a cloud-based video proxy framework was proposed in [6], which utilizes cloud computing to transcode non-scalable videos into H.264 SVC in real time so as to facilitate streaming adaptation to network dynamics. The framework relies on multi-level transcoding parallelization to speed up the transcoding process with plenty of cloud resource.

C. Related Work on R-D and C-R-D Analysis

R-D analysis has been a major research topic for video compression. The basic target is to establish a mathematical model to accurately describe the rate-distortion relationship. Based on the model, an encoder can estimate the rate and the distortion before the encoding process, and adjust the encoding parameters based on the bit rate or the distortion constraints. However, to set up a general and accurate R-D model, it requires a lost of additional computing.

With the emergence of H.264 SVC, there are also some studies on the R-D modeling of H.264 SVC. For example, in [7], the authors proposed a distribution-based R-D model for video content coded by coarse-grain quality scalable (CGS) coding. The model uses the by-product information in motion estimation and quantization to estimate the R-D curve in the early stage of encoding. In [8], stream rate is modeled as a function of frame rate and quantization parameter.

As aforementioned, there are only a few studies on C-R-D analysis. In [4], He et al. investigated and established an analytic power-rate-distortion (P-R-D) model for MPEG-4 encoding. With the P-R-D model, given a certain power constraint, video encoding can be optimized through adjusting some complexity control parameters such as the number of absolute difference computations (SAD), the number of DCT computations and frame rate. It is pointed out in [4] that motion estimation (ME) is the most computation-intensive module in all the standard video encoding systems. In [5], the C-R-D analysis is conducted on H.264/AVC, where the authors addressed two main problems: (1) how to allocate the computational resource to different frames; (2) how to efficiently utilize the allocated computation resource through adjusting encoding parameters. The module analysis shows that the most computation-intensive modules in H.264/AVC encoder include the ME with fractional motion vector (MV) and the R-D optimized coding mode decision, where ME with quarter-pixel precision usually occupies 60% to 80% of CPU time.

D. Definitions

Before we conduct C-R-D evaluation of video encoding, we need to define how to quantitatively measure the C-R-D values. Among the three terms, the definition of rate $R$ is very clear, which is typically measured in terms of the number bits spent per second (bps). For complexity $C$, there are several different ways to quantitatively measure it. For example, it can be measured in terms of the number of basic operations such as addition and multiplication, or processing time, or the power consumption. In this research, we follow the work in [5] to measure the complexity in terms of the number of the consumed processor cycles, of which the power consumption is a monotonic ascending function.

As for distortion $D$, which is often measured in terms of mean square error (MSE) or peak signal-to-noise ratio (PSNR). However, MSE or PSNR based quality metrics are usually used to assess the visual quality in the cases that the spatial resolution and frame rate are fixed. In addition, the MSE or PSNR metrics might not match the human perception. Considering that we are also evaluating the C-R-D performance of H.264 SVC, we adopt the newly developed quality metric in [9] for quality measurement. In particular, the authors in [9] investigated the impact of temporal scalability and quality scalability on perceptual quality. The developed quality metric uses the product of a spatial quality factor (SQF) and a temporal correction factor (TCF), where the SQF assesses the fidelity quality of video based on the average PSNR and the TCF assesses the quality degradation with the decreasing frame rate. The quality metric is formulated as [9]

$$Q(PSNR, f) = Q_{\text{max}}(1 - \frac{1}{1 + e^{(PSNR-s)}}) \frac{1 - e^{-d/f_{\text{max}}}}{1 - e^{-d}}$$

where $Q_{\text{max}}$ is the quality rating given for the highest quality video, ranging from 0 to 100, parameter $p$ is fixed to 0.34 according to empirical studies, $s$ is a video content-dependent parameter, $f_{\text{max}}$ is the maximum frame rate, and parameter $d$ characterizes how fast the quality drops as the frame rate drops.

III. C-R-D Evaluation

A. Evaluation Setup

Three CIF video sequences, football, foreman and akiyo, with 30 fps are used as the test sequences. The H.264 SVC reference software JSVM-9.19 [10] and Oprofile [11] are used
to encode videos and evaluate the encoding consumption. Since H.264 SVC is back compatible with H.264/AVC, we use the same software to evaluate the C-R-D results of H.264/AVC by simply setting the number of quality layers to be one. The GoP size is set to 16 and the prediction structure is illustrated in Fig. 1.

### B. C-R-D Results

Fig. 2 shows the C-R-D results of different videos encoded by H.264 with different quantization parameters (QPs). It can be seen that the cycles, rate and quality drop with the increase of QP. Varying QP has relatively large effect on cycles for videos with fast motion such as football but has little effect on cycles for videos with slow motion such as akiyo. The relationship of CPU cycles and QP can be approximately modeled as a linear function. Fig. 3 shows the C-R-D distributions among different GoPs. We can see that the computing consumption is not constant within a video sequence of one common scene, where the variation is relatively larger for fast motion sequences. Different video content have significantly different computing consumptions.

We now evaluate the C-R-D performance of H.264 SVC with quality (SNR) and/or temporal scalability. Fig. 4 shows the C-R-D results of different videos encoded by H.264 SVC at different number of CGS layers, where the base layer QP is 40 and \( \Delta QP \) between adjacent CGS SNR layers is fixed to 3. It can be seen that the largest consumption increment to base layer is about 30% when the maximum 8 SNR enhancement layers are configured. However, in practice, too many CGS quality layers result in substantially degradation in R-D performance [12]. Thus, it is suggested to employ one base layer with up to two CGS enhancement layers for practical applications. From Fig. 4, we can see that the consumption increment of total three quality layers to only one base layer is minor, which is only about 4% to 7%. To facilitate fine granularity adaptation, each CGS layer’s DCT coefficients can be split into a few MGS (medium-grain quality scalability) sub-layers.

Fig. 5 shows the C-R-D Results of different videos encoded by H.264 SVC at different frame rates. Based on the hierarchical B-frame structure shown in Fig. 1, we select four temporal layers, corresponding to frame rates of 30,15,7.5,3.75 Hz, respectively. It can be seen that the computing consumption is approximately linear with frame rate. Different videos have different slopes for the curves of consumption versus frame rate.

Finally, we study the mixture behaviors of SNR and temporal scalability. Fig. 6 shows the C-R-D results of different videos encoded by H.264 SVC at different frame rates and quality layers. In each figure, each curve represents one quality layer setting, and BL, EL1 and EL2 refer to base layer, enhancement layer 1 and enhancement layer 2, respectively. In each curve, the lower-end point is the result with the lowest frame rate while the upper-end point is the one with highest frame rate. It can be seen that the computing consumption gap between adjacent frame rates is significantly larger than that for quality scalability. In other words, the temporal scalability dominates the CPU cycles.

### IV. Conclusion and Discussions

In this paper, we have evaluated the C-R-D performance of H.264 and H.264 SVC with quality and temporal scalability. We summarize our observations from the experimental results as follows. First, the computing consumption in terms of CPU cycles is approximately linear with QP, number of quality enhancement layers, and frame rate. Second, for H.264 SVC, the temporal scalability has much larger impacts on the computing consumption than the quality scalability. Third, within one video with a common scene, the spent CPU cycles are not constant for different GoPs and there exists considerable variation in fast motion videos.

This work is only a very preliminary study toward cloud media computing. There are a lot of issues that need to be further investigated. For example, we could study the problem of minimizing CPU cycles given certain rate and distortion constraints. Alternatively, we could also investigate how to maximize the quality given the constraints of computing resource and rate.

### References


Fig. 2. C-R-D results of different videos encoded by H.264 with different QPs.

Fig. 3. Per GoP C-R-D results of different videos encoded by H.264 with QP=22.

Fig. 4. C-R-D Results of different videos encoded by H.264 SVC at different number of quality layers.

Fig. 5. C-R-D Results of different videos encoded by H.264 SVC at different frame rates but with fixed three quality layers BL, EL1, EL2, which have QP values of 40, 30, 20, respectively.
Fig. 6. C-R-D results of different videos encoded by H.264 SVC at different frame rates and quality layers, where QP values are set to 40, 30 and 20 for base layer (BL), enhancement layer 1 (EL1) and enhancement layer 2 (EL2), respectively.