Advanced Science and Technology Letters
Vol.75 (SIP 2014), pp.26-29
http://dx.doi.org/10.14257/astl.2014.75.07

Cloud-based Distributed HEVC Video Encoding

Byoung-Dai Lee

Department of Computer Science, Kyonggi University
Suwon, South Korea
blee@kgu.ac.kr

Abstract. As users demand increasingly for high-quality, high-resolution digital video content such as Ultra High-Definition (UHD), the complexity of video coding technologies supporting such requirements also increases. Recently, for performance and cost reasons, distributed video coding via cloud computing has gained increasing attention. In this paper, we explore different performance aspects of distributed HEVC encoding employing two basic video partitioning methods.

Keywords: Cloud Computing, Distributed Video Coding, HEVC, MapReduce.

1 Introduction

High Efficiency Video Coding (HEVC) [1] is the latest video coding technology, developed to meet recent market trends toward large-sized display devices for Ultra High-Definition (UHD) video services. The main advantage of HEVC is its coding efficiency. For instance, it requires only half the bit rate of H.264/MPEG-4 AVC at equivalent video quality ([2][3]). However, its coding efficiency is accompanied by significantly increased computation complexity. Although dedicated hardware solutions such as special-purpose coprocessors and digital signal processors can provide the best performance, they are not only expensive but also lack of flexibility and extendability. As a result, from the performance and economic perspectives, cloud-based distributed HEVC encoding has gained increasing attention in recent years.

Distributed video encoding requires the input video to be partitioned into multiple chunks, which are then encoded independently via computational nodes. As such, video partitioning is an essential step in the encoding process; therefore this study explored the effects of video partitioning methods on different performance in terms of encoding time and bitrate. The study findings inform performance considerations for video partitioning.

The remainder of this paper is organized as follows: Section 2 explains video partitioning methods. Section 3 presents empirical analysis of the chosen video partitioning methods. Finally, Section 4 provides a summary and suggestions for future work.

ISSN: 2287-1233 ASTL

2 Video Partitioning Methods

Uniform-partitioning is a basic partitioning method used in many existing distributed encoding systems. In this method, the input video is partitioned into N video segments of the same size (M/N), where M and N represent the total number of frames to encode and the total number of nodes to participate in encoding, respectively. Therefore, individual segments can contain frames belonging to different GOPs. For instance, a segment comprises frames of "GOP X" and some frames of "GOP Y". In that case, the frames from "GOP Y" cannot reference frames at the tail area of "GOP Y," leading to inefficient inter-coding. In addition, as each node encodes the received segment independently, the first frame of the encoded video must be an I-frame. Therefore, the number of I-frames may increase in accordance with the segments formed. This is crucial in reducing the bitrate of the encoded video because the bitrates of I-frames are considerably higher than those of P- or B-frames.

In the GOP-based partitioning method, the unit of distribution is the GOP; therefore, the size of the segment is equal to that of the GOP and there are M/G segments in total, where G denotes the GOP size. Similar to the uniform partitioning method, GOP-based partitioning also suffers from limited use of reference pictures between GOPs as a result of partitioning. Unlike uniform partitioning, however, GOP-based partitioning does not allow the GOP itself to be partitioned, so there is no additional increase in the number of I-frames. Note that the problem of limited use of reference pictures occurs only in Open-GOP structure because the Close-GOP is an independent unit that can be coded without having any frames of other GOPs. In this paper, we assume that all video streams have Open-GOP structure, as this provides higher coding efficiency.

3 Experimental Results

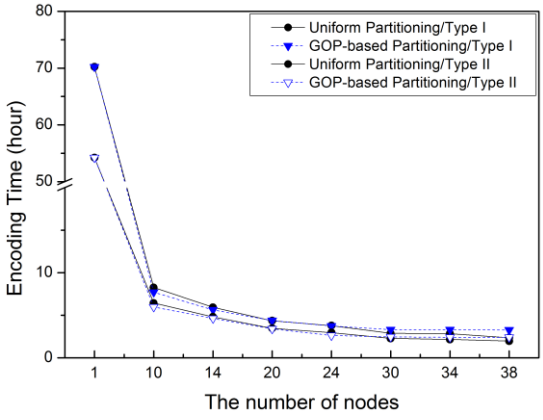

Fig. 1. Changes in encoding time based on video partitioning methods

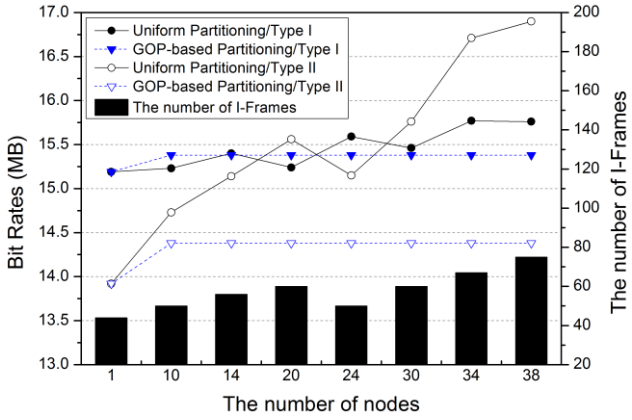

Fig. 2. Changes in the bitrate based on video partitioning methods

In order to explore the impact of video partitioning, we developed a MapReduce-based distributed HEVC encoding system. The workflow of the developed system consists of three stages. In the partitioning stage, the uncompressed video file is partitioned into several non-overlapping segments. In the encoding stage, individual *Map* tasks perform HEVC encoding (we used the HEVC test model (HM 14.0) [4]). Finally, in the combining stage, the *Reduce* task receives HEVC encoded segments from *Map* tasks and merges them into a final HEVC encoded video file.

For the experiments, we used two different types of 4K videos. Type I represents scenes with many motion changes between frames, whereas Type II represents static scenes with relatively few movements and, thus, similar adjacent frames. For HM configuration, GOP was set to 32 and CRA mode was used for the Open-GOP structure.

Fig. 1 shows that the uniform partitioning method requires less encoding time than the GOP-based method. This is because, in the uniform video partitioning, all of the segments are processed in one cycle, whereas in GOP-based partitioning, as the number of segments exceeds the maximum allowable nodes (in this case 38), some segments can only be processed only after the *Map* tasks finishes encoding the current segments. However, as the number of nodes increases, the differences in the encoding times of the two methods tend to be reduced as a result of network I/O overhead.

Fig. 2 shows changes in the bitrate. For the uniform video partitioning, as the number of computational nodes performing encoding increases, the bitrate also increases due to the increased number of I-frames and limited use of reference pictures. By contrast, the GOP-based partitioning shows the same bitrate regardless of the nodes (except in the single node case) because the number of segments and thus the coding structure remain the same. Note that although bitrate generally increases with the number of nodes, the total bitrate decreased in some cases. For example, in Fig. 2, using 14 or 20 nodes to encode Type I video generated 56 and 60 I-frames respectively, with corresponding bitrates of 15.4 MB and 15.24 MB. This result suggests that, depending on how the input video is partitioned, it might be possible to avoid performance loss in terms of both the encoding time and the bitrate. A similar result was also observed for the Type II video.

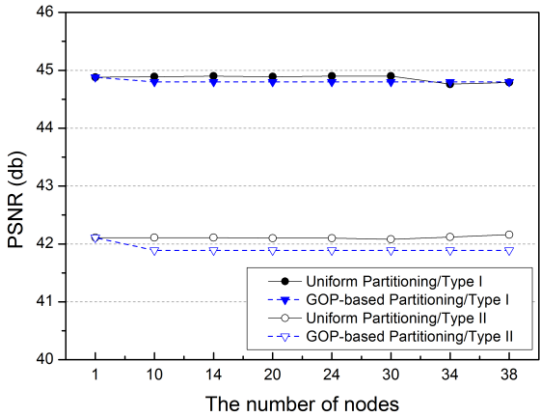

Fig. 3. Changes in PSNR based on video partitioning methods.

As shown in Fig. 3, although there are differences in peak signal-to-noise ratio (PSNR) between content types, the differences between video partitioning methods are negligible.

4 Conclusions

In this paper, we explored the effects of basic video partitioning methods on different performance criteria such as encoding time and bitrate. The experimental results show that distributed processing generally reduces total encoding time significantly at the coat of increased bitrate, due to increased I-framed and/or limited use of reference pictures. However, for uniform partitioning, a case with fewer segments will not always outperform one with more segments in terms of bitrate. Our research plan is to develop video partitioning algorithms that consider various quality metrics for video encoding.

References

1. ITU-T and ISO/IEC JTC 1, High Efficiency Video Coding (HEVC), ITU-T Recommendation H.265 and ISO/IEC 23008-2, Apr. (2013)
2. J. Ohm, G. Sulliva, H. Schwarz, T. Tan, and T. Wiegand, Comparison of the Coding Efficiency of Video Coding Standards – Including High Efficiency Coding(HEVC), IEEE Transactions. Circuits and Systems for Video Technology, Vol. 22, No. 12, pp. 1669-1684, Dec. (2012)
3. M. Pourazad, C. Doutre, M. Azimi, and P. Nasiopoulos, HEVC: The New Gold Standard for Video Compression," IEEE Consumer Electronics Magazine, Vol. 1, No. 3, pp. 36-46, Jan. (2012)
4. HEVC Test Model (HM 14.0), http://hevc.hhi.fraunhofer.de/.

