Toward Content-Aware Video Partitioning Methods for Distributed HEVC Video Encoding

Myunghoon Jeon, Byoung-Dai Lee
Department of Computer Science, Kyonggi University, Korea

Article Info

Article history:

Received Jan 16, 2015
Revised Apr 20, 2015
Accepted May 4, 2015

Keyword:

Cloud computing
Content-awareness
Distributed encoding
HEVC
Video partitioning

ABSTRACT

Recently, cloud computing has emerged as a potential platform for distributed video encoding due to its advantages in terms of costs as well as performance. For distributed video encoding, the input video must be partitioned into several segments, each of which is processed over distributed resources. This paper describes the effect of different video partitioning schemes on overall encoding performance in the distributed encoding of High-Efficiency Video Coding (HEVC). In addition, we explored performances of video partitioning schemes on the basis of the types of the content to be encoded.

Corresponding Author:

Byoung-Dai Lee
Departement of Computer Science,
Kyonggi University,
Iui-dong, Yeongtong-gu, Suwon-si, Gyeonggi-do, Korea.
Email: blee@kyonggi.ac.kr

1. INTRODUCTION

High-Efficiency Video Coding (HEVC) [1] is the most recent video coding standard, and it has been designed to address increased video resolution, such as ultra-high definition (UHD) content. However, the superiority of HEVC encoding performance over that of existing standard coding technologies comes with high computational complexity that produces high encoding bit rates and long encoding times. An essential task for HEVC encoders, therefore, is to resolve the issue of high computing power during the encoding process. Many studies have been devoted to solving these problems. Recent initiatives include the improvement of performance by applyingcloud computing to video coding technology. In particular, considerable attention has been paid to distributed encoding, in which encoding jobs that require high computational complexity are distributed during video encoding by using the unlimited computing resources available in a cloud computing environment. This approach more effectively reduces encoding time than does the use of existing stand-alone encoders.[1]

This paper provides a performance evaluation of distributed HEVC encoding. It focuses on the effects of the video partitioning method on overall distributed encoding performance. The video partitioning methods adopted in this paper can be categorized into two types. The first is a general method, in which video is uniformly partitioned on the basis of the number of nodes that perform encoding tasks. The second is video partitioning on the basis of GOP that considers a video encoding option. Along with empirical analysis of two well-known video partitioning schemes, we extended the GOP-based video partitioning scheme in such a way that dynamic parts of the input video are partitioned into more segments than static parts. In doing so, we can reduce the encoding time with moderate increases in the overall bitrate. In summary, we

[1] This paper is an extended version of the ICONI 2014.[6]

performed distributed encoding by applying two partitioning methods and compared the performance results. The comparison enabled us to identify a video partitioning method that efficiently processes distributed encoding, and we can then recommend considerations for video partitioning.

2. RELATED WORK

As video coding technology advances, the issue of high computational complexity in encoding and decoding processes has arisen. This has prompted numerous researchers to resolve consequent problems in different ways.

One of the typical approaches proposes improving overall performance by enhancing existing video technologies. Research on distributed processing that uses cloud computing technology has also been proposed. In [2], Hadoop [3] proposed split and merge architecture to process H.264 encoding in a distributed manner. In [4], a parallelizing video transcoding system was put forward with the application of the Map Reduce framework [3]. A heuristic algorithm for the efficient scheduling of partitioned video segments was proposed, so that transcoding can be executed while considering task overhead and the computing capacity of a system. The distributed video processing in these studies focused mainly on encoding time, but because processing speed and efficient bit rate control methods have been highlighted as components of HEVC coding technology for processing UHD quality videos, studying video encoding from various viewpoints is necessary.

3. SYSTEM ARCHITECTURE

To measure distributed encoding performance, we developed a distributed HEVC encoding system based on the Hadoop framework. Figure 1 shows the developed system architecture and schematic workflow. The system workflow largely consists of partitioning, encoding, and combining steps. In the partitioning step, a video partitioner partitions an input video for distributed encoding. In the encoding step, partitioned video segments are transferred to mappers where encoding is performed. An encoded video from each map task is then merged through the reducer to create a final HEVC encoded video file. Given that the developed system is based on Hadoop execution, each map task receives a split video on standby from the video partitioner when the current encoding task is complete.

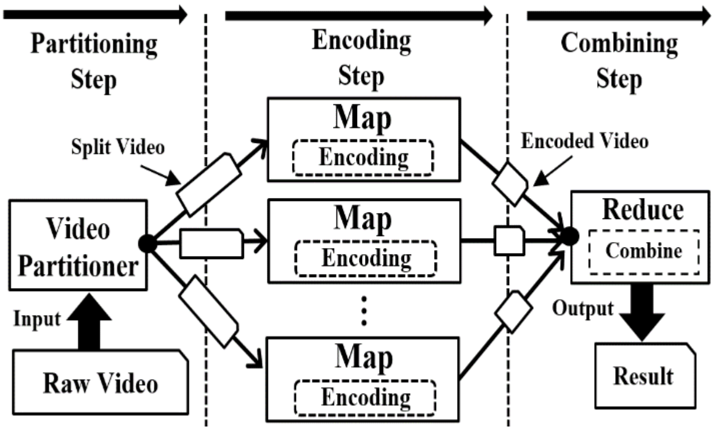


Figure 1. System architecture of distributed encoding for HEVC.

4. VIDEO PARTITIONING SCHEME

4.1. No Partitioning Scheme

Item B in Figure 2 denotes the expected result for the encoding task without partitioning A (e.g., raw video). As shown in B, the I-frame and GOP are generated following the encoding option specified at the lower right side of the figure. The inter-coding processes through the picture reference performed in this process are indicated by an arrow. That is, B is a basic criterion that can compare distributed encoding performance.

4.2. No Partitioning Scheme

Node count–based video partitioning is a basic partitioning scheme used in many existing distributed encoding studies. Item C in Figure 2 shows an example of encoding results derived by uniformly partitioning a video into n nodes. As a video is partitioned, a partitioning area (PA) is created. This PA is an input unit of distributed encoding. In C, therefore, (x/n) PAs are created, which is calculated by dividing node counts (e.g., n) by the number of frames in A (e.g., x). Here, the encoding result of C differs from that of B in two ways. First, the result of C is limited in terms of the use of the reference pictures. For example, GOP 2 in B can reference any pictures in GOP 2, so that an optimum estimated value for inter-coding can be derived. In C, however, GOP 2 is partitioned into GOP 2'; thus, the reference pictured cannot be used in the same manner that is used for B. In C, therefore, limitations on a usable reference picture can arise, leading to inefficient inter-coding. A layout change in the I-frame can occur in accordance with the PA formed, constituting another difference. Because the first frame of the formed PA should be the I-frame when intra-coding is being performed, the overall bit rate increases as the number of I-frames with high bit rates rises. In summary, partitioners are compelled to use reference pictures in a limited manner because of video partitioning, and an increase in I-frames can cause a rise in the bit rates of encoded video.

4.3. GOP-based Partitioning Scheme

With regard to GOP-based video partitioning scheme, D in Figure 2 shows the expected result of GOP-based video partitioning and encoding. Due to the partitioning, this scheme also involves the limited use of reference pictures between GOPs. For example, GOP 1 in D cannot reference the I-frame in GOP 2 (which could be used previously) because of partitioning. However, this scheme is different from the node-count based scheme in that the GOP itself cannot be partitioned, as shown in GOP 2', GOP 3', and GOP 4' in C. Compared with the result of B, therefore, the result of D shows no additional increase in I-frames, and thus prevents an increase inbitrates.Nevertheless, GOP-based partitioning may partition segments that are higher or lower than the number of nodes that perform encoding, depending on content type. This situation diminishes the efficiency of the encoding speed, which is lower than that achieved with node count-based partitioning.

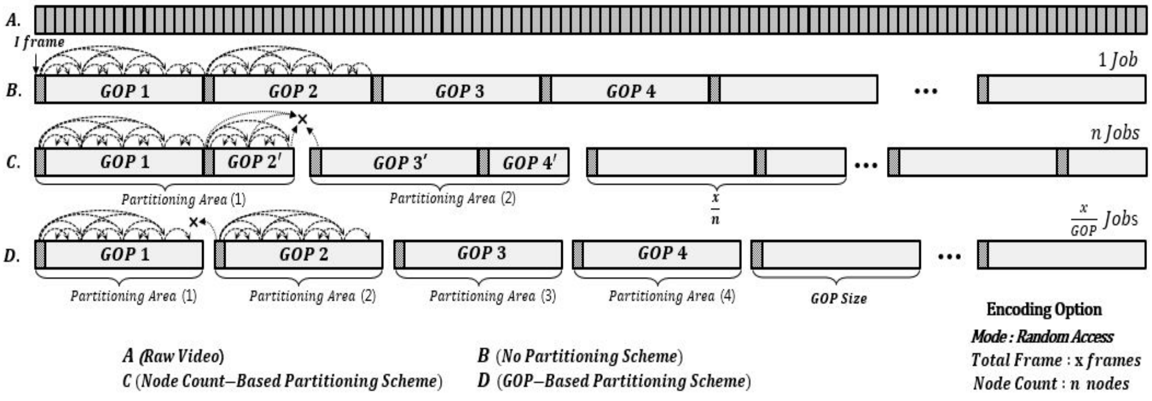


Figure 2. Expected results of distributed encoding with video partitioning.

5. EVALUATION

We developed a distributed encoding system by using the Hadoop framework and performed encoding via the HEVC test model (HM 14.0) [5]. The maximum number of execution nodes used in the experiment is 38. As the encoding option of HM is 14.0, GOP was set to 32. Prediction between frames used the "Random-Access" mode, and "Decoding RefreshType" used the CRA mode. In addition, an input raw video used a 4K video comprising 1,400 frames. To evaluate the effect of the video partitioning scheme on encoding performance, we conducted experiments in which changes in bit rate and PSNR were measured. Figure 3 shows the changes in encoding time according to the number of nodes. When the number of nodes is 1, this number refers to the encoding result with video partitioning. When the number of nodes is two or more, this number refers to the distributed encoding result. The no partitioning scheme exhibits the longest encoding time (see Figure 3). As the number of nodes increases, the node-based partitioning scheme uses less encoding time than does the GOP-based partitioning scheme. This finding is attributed to the fact that the number of PAs formed via node-based partitioning is the same as the number of nodes used. Thus, all encoding tasks are performed in parallel.

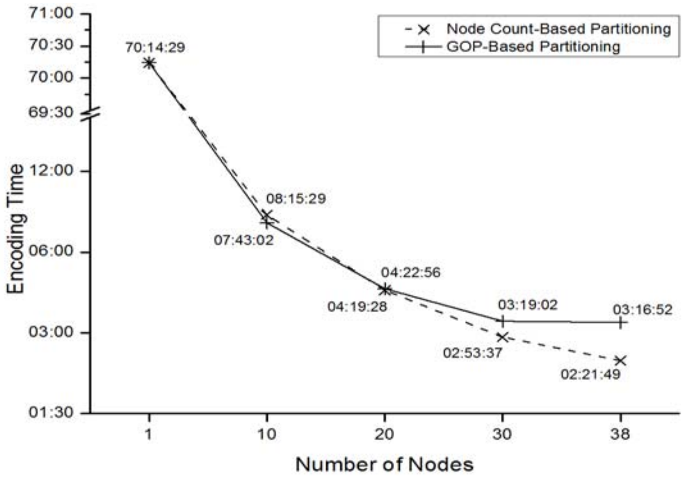


Figure 3. Encoding Time

The GOP-based partitioning scheme forms 44 PAs, which exceeds the number of maximum allowable nodes. As a result, this partitioning cannot be performed within one parallel execution and needs standby PAs. This results in an encoding time that is longer than that observed in node-based partitioning scheme. Note that when the number of nodes falls to between 10 and 20, the two schemes exhibit similar encoding time. This result is attributed to the simultaneous processing capability of standby PAs. Figure 4 shows the changes in bit rates and encoding times. The no partitioning scheme exhibits the lowest bit rate. This finding is due to the fact that there is no increase in the number of I-frames, and there is an absence of limitations on the use of reference pictures. The GOP-based partitioning scheme creates higher bit rates than does the no partitioning scheme, but no change in bit rates occurs even with an increased number of nodes.

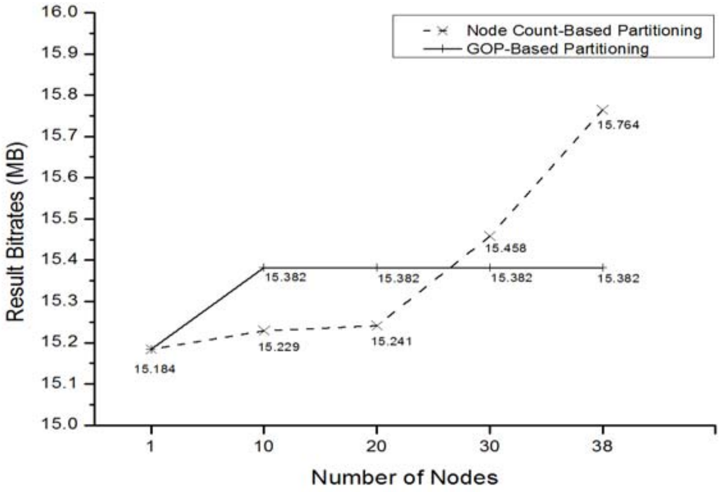


Figure 4. Bitrate

This result is attributed to the fact that the number of PAs remains constant, regardless of the number of nodes. In contrast, in the node-based partitioning scheme, the number of PAs changes with the number of nodes, so that the number of I-frames and references pictures changes as well. This feature results in changes in bit rate. In particular, the node-based partitioning scheme exhibits lower bit rates than does the

GOP-based partitioning scheme by up to 27 nodes. Beyond this number, node-based partitioning exhibits higher bit rates. This finding is attributed to the increase in the number of I-frames and reference limits as the number of nodes increases. Although the PSNR exhibits some changes, they are minimal (see Figure 5). These differences in bit rate and encoding time based on the video partitioning scheme indicate that the video partitioning scheme is a critical factor that affects overall encoding performance. It is particularly important to consider that the overall encoding performance changes according to whether each partitioning scheme is applied. Hence, it is necessary to use a video partitioning scheme that can guarantee efficiency in order to improve the overall encoding performance. However, it is hard to determine which of the two previously mentioned partitioning schemes, (i.e., one based on the number of nodes and another based on GOP) guarantees more efficient performance. This kind of problem occurs due to the difference between how the two partitioning schemes consider the video partitioning process. Even though the two partitioning schemes account for the encoding video and system environment in the process of partitioning, they do not account for the video itself and thus cannot guarantee efficient encoding performance for diverse video.

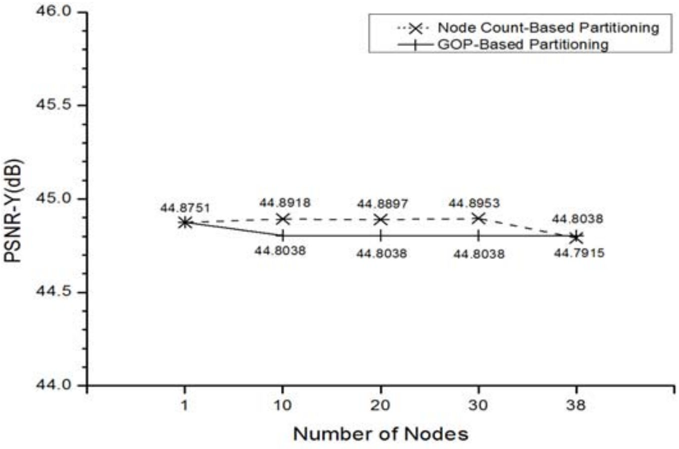


Figure 5. PSNR

We measured the efficiency change of the two partitioning schemes according to content type in order to clarify specific reasons for this problem. Moreover, we recognize that a partitioning scheme that does not account for the video content is problematic, and we provide an analysis of the results of the measurement (see below for specific descriptions).

6. CONTENT TYPE

This chapter describes the change of partitioning scheme performance based on the characteristics of the content of the video. First, we defined video content as static and dynamic. Next, we performed encoding by applying the partitioning scheme based on node number and the partitioning scheme based on GOP size on each of the defined content types. Finally, we compared the efficiency of the two partitioning schemes according to the content type by comparing the measurement results.

In cases where the video content was simple and had little variation, we defined its content type as static, i.e., static-type characteristics mean the video is composed of relatively simple scenes when visually observed. Moreover, static-type content has the characteristic of a relatively low bit rate and encoding time. As the subject of the content does not vary that much, the similarity between the frames is high, which increases inter-coding process efficiency. Figure 6 presents a simple example of this characteristic. Frames 1 to 60 in the picture tend to form a relatively low bit rate and encoding time distribution compared to other intervals. Moreover, when visually determining the complexity of the video by actually playing the relevant interval (where a monkey moves slowly at the center of the screen), it can be confirmed that it is a relatively simple type video. Hence, by aggregating these characteristics, we determined the content that possessed visual characteristics similar to frame 1 to 60 of the video was the static type.

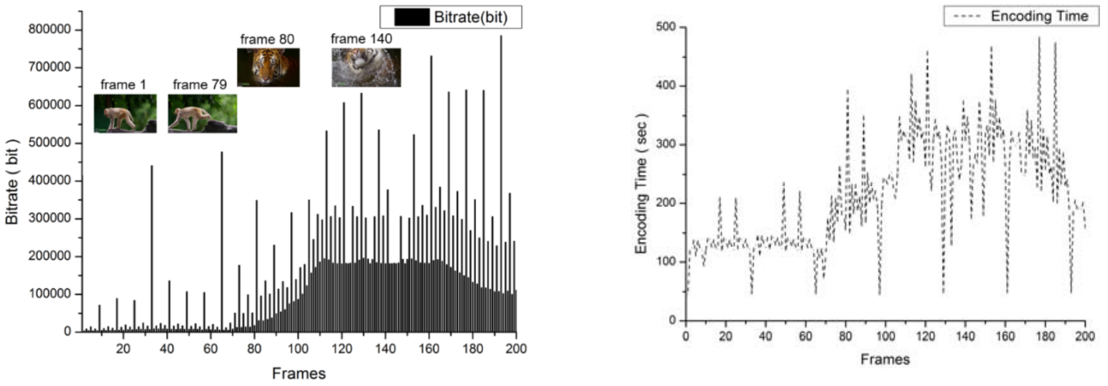


Figure 6. Results of encoding.

6.1. Video Content Type Decision

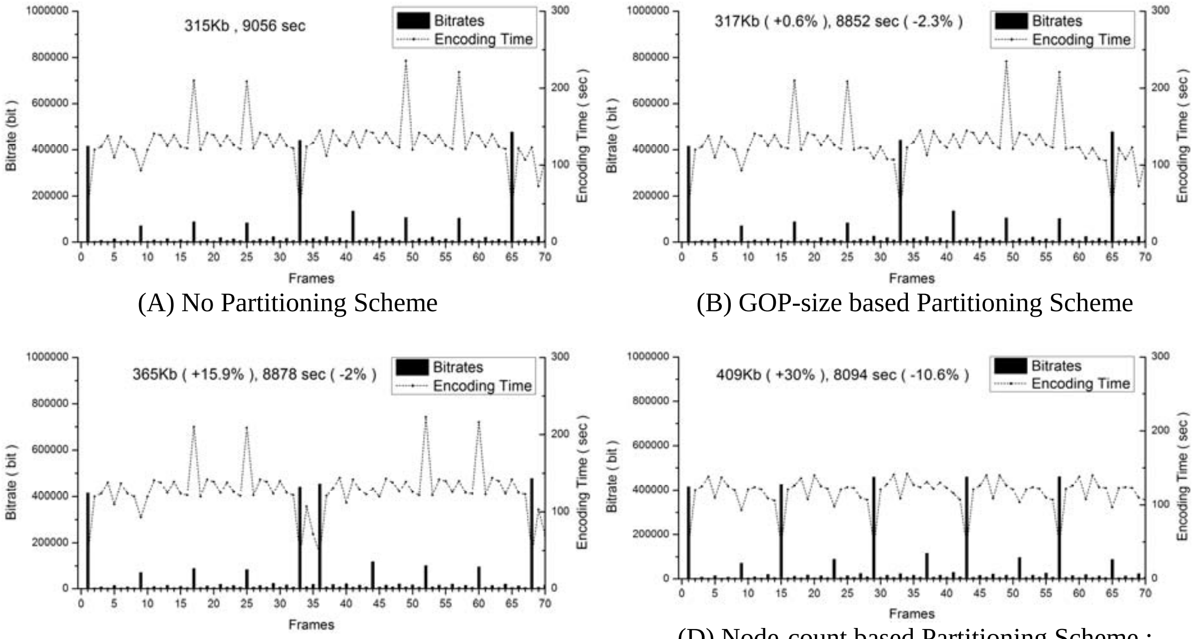
In order to define the content type of the video, we used the following method. First, we performed encoding on a number of videos that contain different types of content and collected bit rate and encoding time results. We then visually confirmed the characteristics of the content by playing each video. We then collected relevant information. Finally, the video content type was defined using the collected information. The specific characteristics of each of the defined content types are described below.

To explain each type of content, we present the encoding results of a particular video as an example. The video used for encoding was a 4K video composed of 200 frames and the 32 GOP size. Random access mode was used to encode the video

6.2. Static-Type Content

We implemented the encoding of static-type contentusing a partitioning scheme based on the number of nodes and a partitioning scheme based on GOP. We described the characteristics of each partitioning scheme according to the results. We performed encoding successively in a single node without distributed processing of the PA to measure the performance of the video's partitioning aspects. Figure 7 shows the encoding result of video that is partitioned by applying each of the partitioning schemes with a 70 frame video composed of static-type content. Figure 7-A presents bit rate and encoding time distribution of the encoding implementation results where static- type content was not partitioned. Frames 1, 32, and 64 in the figure show the location of the I-frame. Since the I-frame is basically intra-coded, it has shorter encoding time compared to other frames that are inter-coded. Meanwhile, in terms of bit rate distribution, the I-frame forms a considerably high bit rate compared to other frames. Hence, in the distributed encoding process of static-type content, I-frames can be identified as a critical factor that affects the overall bit rate and encoding time. Figure 7-C shows the encoding implementation result where the partitioning scheme based on the number of nodes is applied using two nodes. C forms two PAs that have a length of 35 frames. Since the 36th frame is encoded as an I-frame, an increase in bit rate occurs. In C, the bit rate is 365Kb and the encoding time is 8,878 seconds, which shows a 15.9% increase in bit rate and a 2% decrease in encoding time compared to A. Figure 7-B shows the encoding results of applying a partitioning scheme based on GOP. Unlike Figure 7-C, B shows no change in I-frame following the partitioning, and the inter-coding process also shows little change. When comparing the measured bit rate and encoding time, we see the bit rate increased 0.6% and the encoding time decreased 2.3% compared to A.

Moreover, we implemented encoding using a partitioning scheme based on nodes and changed the number of nodes to five. This was to account for the characteristics of the partitioning scheme based on nodes, which variably forms a video's PA based on the number of nodes rather than the partitioning scheme based on GOP. Figure 7-D shows the encoding results of the partitioning scheme based on five nodes. In D, it is clear that five I-frames are generated as five PAs are formed. Hence, changes in bit rate and encoding time follow. The encoding results indicate a bit rate of 409Kb and encoding time of 8,094 seconds. Comparing with A, the bit rate showed an increase of 30%, while the encoding time showed a decrease of 10.6%. Even though it is possible to say that efficiency increased in D compared to C in terms of encoding time, the bit rate increased to up to 30% overall. As a result, it cannot be said to be efficient. Finally, in partitioning static-type content, although the partitioning scheme was based on the number of nodes and can diminish the encoding time compared to the partitioning scheme based on GOP, it had a disadvantage in that it made the bit rate more inefficient.



(A) No Partitioning Scheme

(B) GOP-size based Partitioning Scheme

(C) Node-count based Partitioning Scheme : 2Nodes

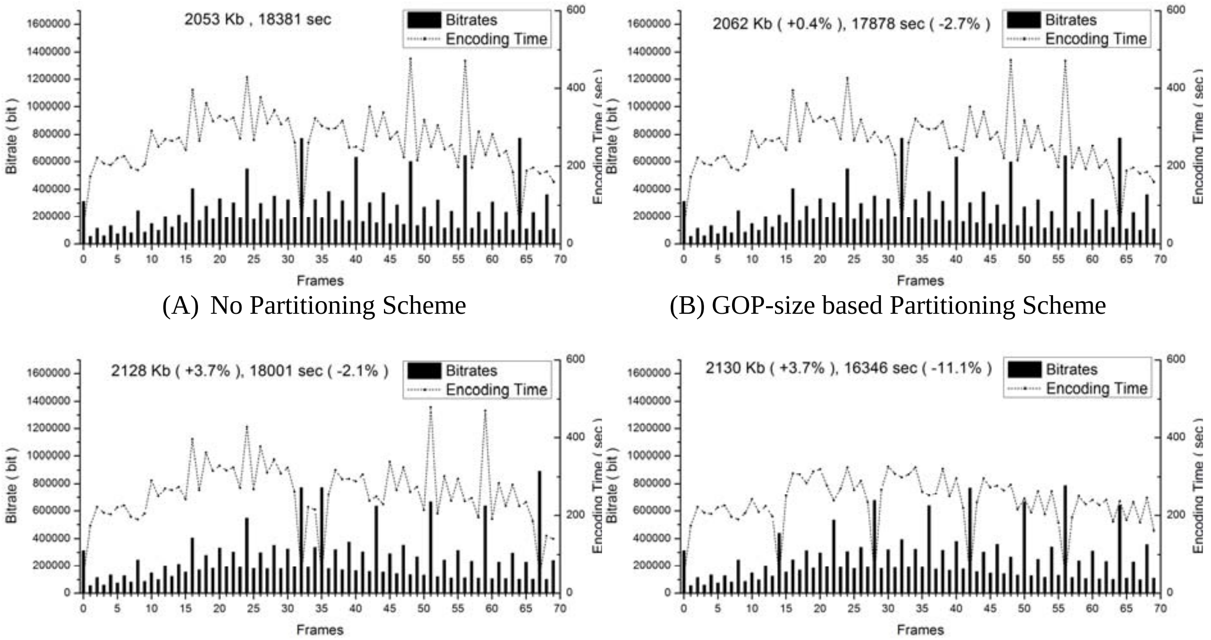
(D) Node-count based Partitioning Scheme : 5Nodes

Figure 7. Results of static-type content encoding with video partitioning.

6.3. Dynamic-Type Content

Dynamic-type content refers to content whose characteristics are opposite those of the static-type. That is, dynamic-type content is composed of objects that have complex shapes and many alterations. Hence, dynamic-type content is composed of complicated scenes when visually assessed, and it has the characteristic of having a high bit rate and long encoding time in the encoding process. For example, the interval between frame 80 and 200 in Figure 6 forms a relatively high bit rate and encoding time compared to static type content. When observing the video's complexity by playing an interval of the video where a tiger causes a huge water spray as it rapidly sways its tail in the water, a dramatic change in the objects was confirmed. Hence, we defined the type of content that has visual characteristics similar to frames 80 to 200 as the dynamic type. We performed dynamic-type content encoding by applying a partitioning scheme based on the number of nodes and a partitioning scheme based on GOP. Figure 8 shows the encoding result of applying each of the partitioning schemes on a 70-frame dynamic-type video content. Figure 8 shows the encoding result of applying each of the partitioning schemes on a 70-frame dynamic-type video content. Figure 8-A shows the result of stand-alone encoding implementation on dynamic-type content without partitioning. It can be confirmed that the dynamic-type content is had high bit rate distribution overall.

However, unlike the previously mentioned static type content, the bit rate of the I-frame does not show a distinctive difference compared to other b-frames and p-frames. For example, the interval between frames 15 to 25 in A is composed of frames that have a bit rate similar to or higher than that of the I-frame. In particular, frame 17 and frame 25 have bit rate higher than the I-frame, which is caused by a large change in the dynamic-type object composition. This increases the computation complexity of inter-coding process, resulting in a high bit rate. Hence, when partitioning dynamic-type content, the overall bit rate and encoding time are not only affected by the I-frame but are also influenced by the b-frame and p-frame. Figure 8-B shows the result of encoding dynamic-type content by applying the partitioning scheme based on GOP. Comparing the encoding results with A, the bit rate increased by 0.4%, and the encoding time decreased by 2.7%. This result does not differ much from that of the previously measured static-type content. Hence, we can say that the partitioning scheme based on GOP has little difference in efficiency in terms of content type. Figure 8-C shows the result of a node-based partitioning scheme using two nodes. Comparing the encoding performance with A, the bitrate increased by 3.7%, and the encoding time decreased by 2.1%.



(A) No Partitioning Scheme (B) GOP-size based Partitioning Scheme

(C) Node-count based Partitioning Scheme : 2Nodes (D) Node-count based Partitioning Scheme : 5Nodes

Figure 8. Results of dynamic-type content encoding with video partitioning

This result indicates a big difference in efficiency in terms of bit rate when compared with the previous static-type content results. This is because the I-frame has a low impact on the overall bit rate in dynamic-type content, unlike static-type content. It also showed similar efficiency in terms of encoding time. Figure 8-D shows the result of encoding by applying the partitioning scheme based on a node with five nodes. Comparing the encoding results with A, the bit rate increased by 3.7%, and the encoding time decreased by 11.1%. Comparing this result with the result of the static-type content implementation, an efficiency increase is confirmed in terms of both of bit rate and encoding. Moreover, compared with the partitioning scheme that used two nodes, the one with five nodes showed a considerable decrease in encoding time despite a similar bit rate. This is because inter-coding has a relatively efficient allocation of I-frames in cases of complicated frames because five PAs are formed. For example, five PAs are formed in D and frame 15 is intra-coded into I-frame. However, as was mentioned in A, as frame 15 expressed relatively complicated scenes, it was one of the frames that originally formed a high bit rate that implemented inefficient inter-coding. Hence, even though D is encoded at a high bit rate because frame 15 is intra-coded, there is no large increase in bit rate since the frame originally had a high bit rate. However, the encoding time showed a large difference. Comparing the encoding results of frame 15 in A and D, the encoding time was 270 seconds and 45 seconds, respectively.To sum up these characteristics, a partitioning scheme based on the number of nodes can produce changes in efficiency according to content type, and it can implement an efficient partition for dynamic-type content.

6.4. Efficiency Change

The efficiency change of the two partitioning schemes according to the content type described in the previous section can be summarized as follows. First, the partitioning scheme based on GOP showed consistent encoding performance without a large change of efficiency in terms of content type. Hence, it can guarantee relatively stable encoding performance when partitioning video that is composed of diverse content types. However, a partitioning scheme based on number of nodes showed different efficiency with different content type. It had inefficient encoding performance during the process of partitioning static-type content because on increase in high bit rate occurred. However, when partitioning dynamic-type content, it showed efficient encoding performance overall, as the decrease of encoding-time exceeded the increase of bit rate. This characteristic are confirmed in Figure 9, which presents the change of bit rate and encoding time of the two partitioning schemes according to the content type expressed as a percentage. The video length used in the picture was 70 frames and the resolution was 4K. We did not implement distributed encoding to examine the change in encoding time and bit rate caused by the partitioning of the video itself. Hence, the encoding time in Figure 9 is the sum of the encoding time of each of the PAs. Figure 9-A shows the change of bit rate and encoding time of the two partitioning schemes on static-type content. The partitioning scheme based on GOP did not produce any change in bit rate and encoding time regardless of the number of nodes. However, in the partitioning scheme based on the number of nodes, the encoding results changed according to the changing number of nodes. Although the encoding time decreases as the number of nodes increases, the overall efficiency of the encoding was low due to the increase in the high bit rate. Figure 9-B shows the performance change of the two partitioning schemes on dynamic-type content. The encoding results of the partitioning scheme based on GOP showed a similar change to those of the static-type content. Hence, the partitioning scheme based on GOP can be said to have a relatively small change in encoding efficiency based on the content type. However, the partitioning scheme based on the number of nodes showed a different result from that of static-type content. In particular, in case of dynamic-type content with two nodes and four nodes, the bit rate decreased more than the static-type content, and the encoding time decreased even more. As a result, the efficiency greatly increased. However, with a steady increase in the number of nodes, even though the encoding time decreased, the efficiency eventually decreased as the bit rate increased. The efficiency change of the two partitioning schemes based on content type can be summarized as follows. The partitioning scheme based on GOP is able to provide relatively stable encoding efficiency regardless of the content type. However, the partitioning scheme based on the number of nodes had different levels of efficiency according to content type. In particular, it had high efficiency following the increasing number of nodes in dynamic-type content. However, if the number of nodes exceeded a certain level, the bit rate increased, which eventually caused less efficient encoding. Hence, even though a partitioning scheme based on the number of nodes provides unstable encoding efficiency overall, it has the characteristic of being able to provide optimal encoding efficiency for dynamic-type content.

To sum up, applying a partitioning scheme for distributed encoding that does not account for video content cannot guarantee optimal encoding performance, and it can create a problem of lower encoding efficiency.

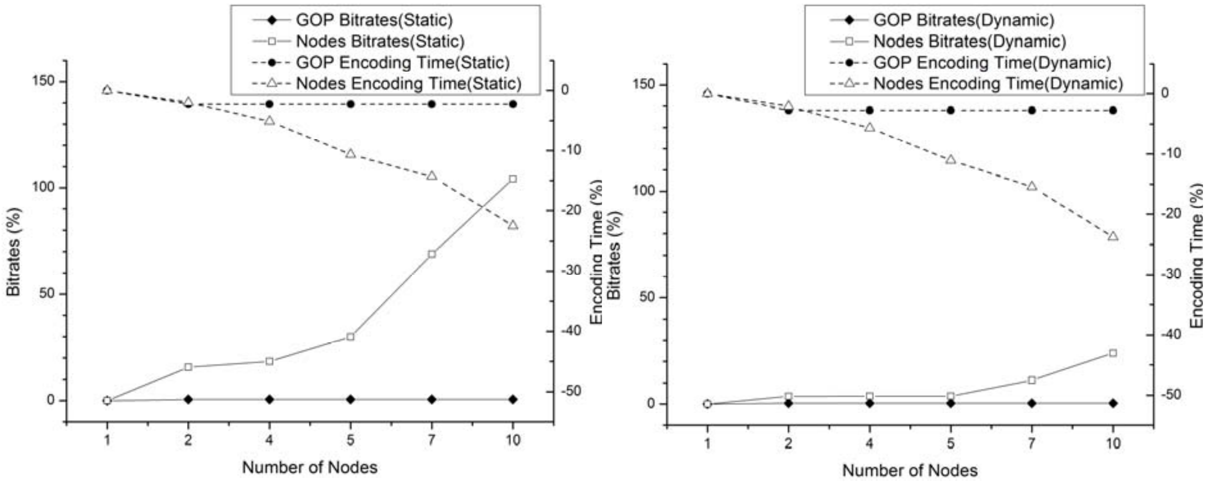


Figure 9. Encoding results of content type with video partitioning

7. CONCLUSION AND FUTURE WORK

This paper described the effect of video partitioning schemes when performing distributed encoding, and it listed several points that video partitioning schemes should consider for efficient encoding. We classified the video partitioning schemes into (a) a video partitioning scheme based on the number of nodes and (b) a partitioning scheme based on GOP. We also pointed out that these two types of video partitioning schemes are problematic in that they do not account for the image itself during the partitioning process. Therefore, we measured the change in encoding efficiency of the partitioning schemes according to content type. The measurement results indicated changes in efficiency according to content type, and we described the potential problems that can occur when the content types are not considered. As for the future research, we are currently working on the partitioning schemes that take into account the image's content type and developing a content type differentiating algorithm that uses color difference components of video and then uses variable partitioning schemes based on this.

ACKNOWLEDGEMENTS

This work was supported by Kyonggi University‘s Graduate Research Assistantship 2014.

REFERENCES

[1] Sullivan, Gary J, *et al.*, “Overview of the High Efficiency Video Coding (HEVC) Standard”, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649-1668, Dec 2012.

[2] Pereira, Rafael, *et all.*, “An Architecture for Distributed High Performance Video Processing in the Cloud”, *Proc. of 2010 IEEE 3rd International Conference on Cloud Computing*, 2010, pp. 482-489

[3] http://hadoop.apache.org/

[4] Lao, Feng, *et al.*, “Parallelizing Video Transcoding Using Map-Reduce-Based Cloud Computing”, *Proc. of 2012 IEEE International Symposium onCircuits and Systems*, pp. 2905-2908, May 2012.

[5] http://hevc.hhi.fraunhofer.de/

[6] Myunghoon Jeon, *et al*., “Toward video partitioning schemes in distributed video processing for HEVC”, *International Conference on Internet (ICONI)*, Dec 2014.

BIOGRAPHIES OF AUTHORS

Myunghoon Jeon received his B.S and M.S. degrees in Computer Science from Kyonggi University, Korea in 2004 and 2011 respectively. He is a Ph.D. student working at department of Kyonggi University

Byoung-Dai Lee is an assistant professor at the department of Kyonggi University, Korea. He received his B.S. and M.S. degrees in Computer Science from Yonsei University, Korea in 1996 and 1998 respectively. He received his Ph.D. degree in Computer Science and Engineering from University of Minnesota, twin cities, U.S.A. in 2003. Before joining the Kyonggi University, he worked at Samsung Electronics, Co., Ltd as a senior engineer from 2003 to 2010. During the period, he has participated in many commercialization projects related to mobile broadcast systems. His research interests include sensor networks, open mobile platform, and mobile multimedia broadcast.