

Int. J. Advance Soft Compu. Appl, Vol. 13, No. 3, November 2021
Print ISSN: 2710-1274, Online ISSN: 2074-8523
Copyright © Al-Zaytoonah University of Jordan (ZUJ)

Optimization of Video Cloud Gaming Using Fast HEVC Video Compression Technique

Mosa I. Salah¹, Ahmad A. Mazhar², Manar A. Mizher³

¹Faculty of Architecture and Design, Al-Zaytoonah University of Jordan, Jordan
e-mail: mosa.salah@zuj.edu.jo

²SAE, School of Creative Media, Department of Multimedia, Amman Institute,
Jordan
e-mail: ah.mazhar@yahoo.com

³Faculty of Computer Sciences and Informatics, Amman Arab University, Jordan
e-mail: mmizher@aau.edu.jo

Abstract

Cloud computing is a model of technology that offers access to system resources with advanced level of services ability. These resources are measured reliable, flexible and affordable for several kinds of applications and users. Gaming manufacturing is one field that expands the profits of cloud computing as numerous new cloud gaming designs have been presented. Many advantages of cloud gaming have exaggerated the success of gaming based on the improvements on traditional online gaming. Though, cloud gaming grieves from several downsides such as the massive amount of needed video processing and the computational complexity required for that. This paper displays the original system drawbacks and develops a new and original algorithm to speed up the encoding process by reduces the computational complexity by exploiting the block type and location. Enhancements on the video codec led to 12.2% speeding up on the over-all encoding time with slight loss of users' satisfactions.

Keywords: *Cloud gaming, Computational complexity, Motion estimation, HEVC, Video Encoding*

1. Introduction

Digital communication is becoming more popular day after another; thus, video gaming is one field that has been affected accordingly. Online gaming has grown up and it is now more popular worldwide because of this revolution than any time before. This spread of entertainment is found across the globe and in all societies. According to the Entertainment Software Association (ESA), about 60% of Americans play video games regularly [1]. As a normal result of this increase of digital gamers, the revenue of gaming industry has grown-up to reach billions in last few years [2]. Gamer attracting has no limit, so this growth is not the target of game developers. Game developers are struggling to improve their products to attract more game players and enlarge this investment. Exploiting cloud-computing techniques and emerging it with game development is one of the most focused areas that game developers are gambling on. Deploying games on a cloud and fasten the games using the huge capabilities of a cloud. Exploiting cloud computing in gaming results new area of technology that so-called cloud gaming. Clouding are brilliant way to exploit these capabilities to process games faster instead of consoles or personal computers. This technique would end with faster games and wider reach of customers. Using cloud gaming will help client not to process the whole tasks itself. That required the cloud to do the most complex computational tasks and that results a reduction of processes that need to be executed on the client side. These processes may contain rendering high-definition video sequences and mathematical computations that would be conducted on the cloud. Then on the client side, the game will be dealt as a video (GaV) technique [3].

In the field of on-line gaming, the term cloud gaming became wider spread and one of the most important terms in the field. As cloud computing has increased the reliability of computing technology overall, cloud gaming increased the efficiency on gaming by relying on cloud capability to improve gamer experience. In cloud gaming, the cloud can run the game and acts as a game engine. These huge processes can be done remotely on the cloud and then be streamed to the end user's device. Only the desired actions and controlling events are the parts of the processes to be transferred to the client's device as this is a limited device compared to the cloud capability [4, 5]. This clouding exploiting technique, the client's device is free of processing the huge amount of data as usual technique in consoles. Client is only required to process the limited amount of data that is related to controlling and actions.

Rendering, compression and high complexity process reserve more than 80% of the whole game. However, these parts are not required from the client's device which reflects more exploiting of resources. In this case, user needs only a video decoder to show the received game sequences. Users also receive the communication messages and instructions from the game server. The idea behind cloud gaming is useful to eliminate the need of user's hardware high performance. The only part the user needs

to enjoy such kind of games is an Internet connection with a minimum of 5 Mbps connection speed. The speed of the Internet connection is the way that the client's PC requires to conduct a successful enjoyable cloud gaming experience. The good news today is that the internet is now available in most places around the world with almost download speed above the required by most of the cloud games [6]. One disadvantage of the cloud gaming technique is consuming the Internet bandwidth because if the huge amount of data to be communicated, however, reducing the computational load on the user side is still an essential benefit compared to this or other factors.

Cloud gaming introduced many benefits and supported game developers. Game providers and gamers are also parts of the improvement of this technique who also got benefits from this improvement. Apart of the achievement of efficiency, an important advantage that the developers will gain is guaranteeing that no more hacking or illegal copying of their games is threatening their games any more. The game is stored in a safe place and is not downloaded on the client's machine. All clients now are required to access the game on the cloud using their secured accounts and on their own expenses. New kind of services and business models are now to be developed. On the other hand, gamers also can get significant benefits of converting to cloud gaming. One is by accessing their game at any time from any place providing Internet, the second is by saving money as it is not required to keep upgrading their devices hardware in every short period of time.

Finally, the online gaming and gaming with worldwide players is easier and with minimal loading on their machines. That is a good chance for participating in tournaments and game sharing [7]. The CloudUnion is a successful example of cloud gaming platform. After users downloaded the client, the CloudUnion streams the game as video in different resolutions. The minimum bandwidth required is 2Mbps which is available in most countries worldwide. However, 6 Mbps will guarantee high quality game streaming [8].

2. Related Works

Cloud gaming term has a unified idea of exploiting the cloud to improve gamers enjoyment, however, structures of exploiting this cloud is implemented in different ways. Three main categories can be illustrated in this field [9]. The first category depends on rendering all the computational operations at the server side. These operations contain gaming logic and video (rendering, capturing, encoding and streaming). This category is known as Remote Rendering GaaS (RR-GaaS). In this category, client is not obligated to have any part of game processing on his device. In this case, client's device is required to receive the video streamed to their device, decode it and display it. This technique achieved high saving of the computational time and load on the client's side. That is really a major benefit that affects gaming

experience. However, suffering from end-to-end delay is a problem that may occur in this scenario.

Client-side responsibility is used more in the second category as more relying on the client device is required. Local Rendering GaaS (LR-GaaS) is the technology used here. The terminal is responsible for the whole processing operations. The operations here include several processes on video such as coding, decoding and streaming. One drawback of this approach is the overloaded demand on the client to perform the real-time rendering of game scenes. The power consumption and time complexity are the two most important factors that affect the game experience. The third category is hybrid processing techniques that designed to conduct one part of the processes on the cloud side while the other part is conducted on the client side. Cognitive Resource Allocation (CRA-GaaS) is the style of this cloud gaming scheme. According to system situation, the cloud gaming architecture would be able to combine the processes that can be distributed to execute through the network. The RR-GaaS is considered the most widespread and developed model over other models. OnLive and Sony (Gaikai) are giants companies and two of the most popular and leading cloud gaming companies that use RR-GaaS.

Many profits are expected from using RR-GaaS cloud gaming for both game creators and users [10]. Inappropriately, many disadvantages are still preventing this approach from being the optimal solution. Computational complexity is one of the main challenges that limiting using this approach because of the huge amount of processing that need to be done on the cloud. Running the game, updating the scene, controlling and encoding are all time-consuming processes that are required from the cloud to be achieved in real time.

As any gaming cloud is expected to serve huge number of players, any minor decrease of the performance will be raised to be sensed huge and serious problem when multiplied by thousands of real-time players at the same time. Obviously, the encoding time reduction will help reducing the processing and responding delay and power consumption. Encoding time reduction is vital as it will not only affect processing time itself, but also will reduce clock frequency which decreases power consumption [11].

Several scenarios have been proposed to improve the computational cost. In [12], the side information from the game engine was exploited to decide either to process or skip it based on some criteria. It considered the locations of macroblocks and performed compression decision based on it. Foreground and background macroblocks are checked and used to reduce the computational complexity. This research showed that up to 18% of the game encoding time was saved.

In [13], an investigation of speeding up the cloud gaming operations at the server side has been conducted. The mechanism focused mainly on accelerating power intensive

process of video encoding using any available information about a game object on the game engine. The results showed the motion estimation is the most time-consuming process and the most power wasting step. Applying recent video codecs such as H.264 lead to about 8.86% acceleration of the whole gaming process.

The computational complexity is severely challenging the RR-GaaS architecture of game clouding. Games are developed every day in all sides: graphics, actions and resolution. Each one of these factors plays an important key on rendering and encoding time increase. However, numerous achievements have been done on some scenarios to decrease the encoding time for H.264/AVC and HEVC video codecs. Some other works concentrated on the computational complexity of the cloud gaming system and how to keep it at the lowest. As a main principal of this research, is to invent an improvement to the cloud gaming system, this section will discuss these works and review main discussed ideas.

The real-time video encoding is one limitation since the H.264/AVC has been launched. Since most games in current market are based on online, it require live streaming and other effective factors such as: encoding, decoding and power consumption. Therefore, these components are monitored and researched to be improved. In [13-18], ideas to reduce the computational complexity of H.264/AVC and HEVC were proposed. The concepts in these works focused on two main steps and can be summarized into: motion estimation and mode decision. They appealed the two factors affect the overall computational time severely. Some of these research techniques have been recommended to reduce the number of motion estimation search and then save the total encoding time. Other part of these works focused on the mode decision in diverse settings to achieve important speeding up of the encoding total time, also the computational complexity of real-time applications using H.264/AVC encoding system. Final set of the above research have developed methods by altering encoding parameters to adapt the computational complexity of the encoder to the accessible resources.

The previous ideas have been proposed mainly to reduce the computational complexity of video encoding process in general. They are assumed to automatically reduce the computational complexity of the cloud gaming overall. Even though, this is correct somehow, but it is still not capable to consider the special features of the cloud gaming. For example, they did not show any considerations about using the side information that normally come with compressed bitstream in games. Improved solutions are more likely to consider all encoded information from the streamed and gained from game scenes.

The computational complexity of cloud gaming has been well-thought-out in several researches. Using run-time graphics rendering context, the real-time video encoding process has been monitored and analyzed in [19]. The idea founded mainly to select a set key-frames and to exploit the 3-D Warping algorithm to interpolate internal non-key frames. In [20], the depth map from the game engine has been exploited to prevent the huge motion estimation process. In [21], the depth information has been also used but this time to increase the motion selection step of the codec H.264/AVC. The authors of [22] suggested an idea to adjust parameters based on the cloud competences. This will exploit the available complexity to meet the complexity provided by the cloud gaming. In [23], authors invented a ranking method to reduce the power needed for processing. The main idea based on removing objects that less important to the scene.

3. Proposed Method

In this paper, a new idea was devised to use the game engine side information. That is achieved by either skip or process this information according to some critical conditions. The proposed idea considers macroblock location so it is more comprehensive than the previous ones. Both macroblocks: foreground and background macroblocks would be processed and performed based on its location. For each macroblock in a foreground object, the movement is detected and compared to the predicted motion vector. As a result of this prediction, three scenarios can be achieved: skip the motion estimation process, perform the motion estimation process with some restrictions, or perform the normal motion estimation process. The background macroblocks are also considered in this research. The motion estimation for these macroblocks is accelerated. A background motion vector is used to compare with the motion vector that already predicted. The three scenarios of motion estimation process will also be applied here for the difference between the background motion vectors and the predicted ones.

The conventional block diagram cloud gaming system depends on performing the gaming operations on both sides, the client and the cloud. The client side is in charge of only collecting user interactions and then decoding the received bitstream from the server side. However, the server side is accountable to run the game logic, render the game objects and deal with frame by frame encoding with the video encoder. All these computational tasks are to be done on the server side. In such cases, the power consumption is a huge challenge on both the cost and ability if the game provider deals with huge number of players. One more challenging in such cases, the server side performs the compression process without any information about the game objects. Although some side information about objects sizes and locations are available, it is

not exploited to facilitate the compression operations. Somehow, this information can be used to skip some of the complex encoding operations.

Considering the observations above, we suggest a new technique to exploit the object's information available at the encoder side to reduce the computational time with minimum bitrate increase and as low as of the quality degradation. To achieve the goal, A fast H.265 HEVC encoding proposed in [17] has been exploited to speed up the process. In addition, two paths have been used to detect and improve the encoding process, foreground and background MBs baths are considered in the study. The enhanced H.265 codec used is described in details in the research in [17], however, the paths are described in the following 3.1 and 3.2 subsections.

3.1 Foreground Macroblocks

As an early step, each MB position is checked. The MBs that are located inside the boundary of any game object is measured as foreground macroblocks. After a macroblock is recorded as a foreground MB, its movement is checked to study if it is less than the Low Threshold (L-Th), if so, it will be categorized as closed MB. However, if it is more than L-Th, it would be examined for a second time to check if its less than the High Threshold (H-Th), if so, it will be now categorized as average. If the movement is above the H-Th, the MB will be classified as far MB.

The next stage is to conduct the motion estimation (ME) for the examined MB. The ME of each MB is skipped if it is categorized as close. However, the MB will be going through a improvement ME if it is categorized as average. Finally, if the MB is categorized as far, that means the MB needs to go through the normal ME process.

3.2 Background Macroblocks

The MB is considered a background MB if it is not positioned inside any boundaries of any game objects. As the worst case, the game engine would not offer any information about the background nor its movement. For this case, our proposed algorithm is designed to work closely with such cases. However, if the game engine offers the required information, the algorithm is modified to exploit this information. Though, this will improve the overall processing time, the proposed algorithm can work even with no game engine's feedback using a novel technique.

Initially, the background MB that is currently in process requests to be checked if it is also positioned in the background of the reference frame. In addition, the current MB is tested if it was located behind any other objects in the previous encoded frame. After checking the previous conditions, the MB is not a background MB in the previous frame, the normal ME is run as full search method. Then, a ME for the current MB is performed based on the background ME with bearing in mind the motion vectors for

all MBs. In situation of an early MBs in a frame, where no motion vectors to be measured, the normal ME will be used. After insufficient MBs, motion vectors will be extracted and used through similar steps as in foreground MBs.

Figure 1 displays the phases that our algorithm practices at the video encoder side. This process is achieved for each MB and for all obtainable modes of each MB. Based on the case of a MB, the ME can be skipped for some modes but to be completed for another. The first step in the proposed algorithm is to check if the MB is positioned inside a game object border, if so, another examination is conducted to check in which threshold this MB is categorized. The ME is chosen based on the level of the threshold. However, if the MB is located inside an earlier frame’s background and not in a game object, two different thresholds are examined, and the ME is chosen consequently. Finally, if the MB is not measured in any of the two previous cases, a normal ME is conducted.

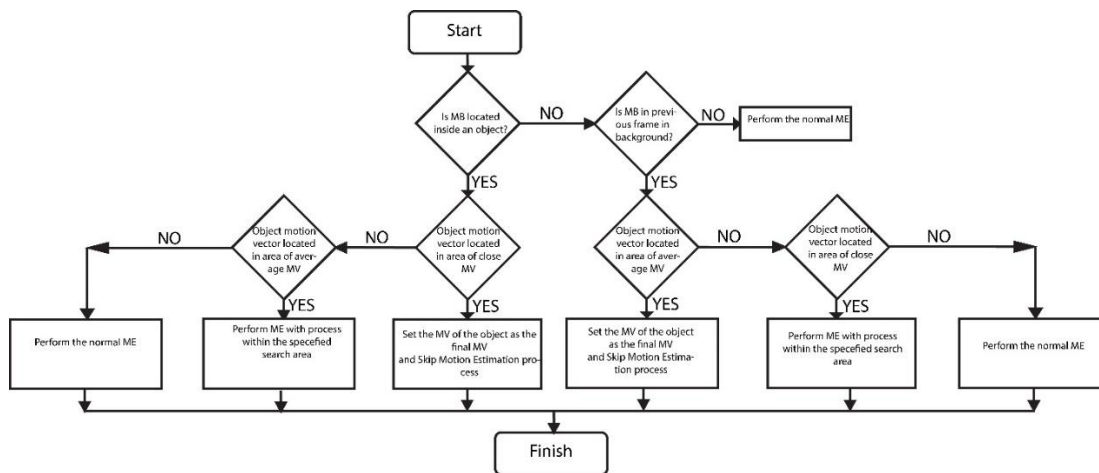


Fig. 1 : Flowchart of the proposed algorithm

The set of values of the thresholds is an important factor that performs a vital role in the outcome of the proposed algorithm. Though, these values can be specified in the initial steps for any specific encoder and depending on the video type or the desired output. The values of the threshold parameters are shown in section 4, we proposed and used these parameters for the experiments we have done using the HEVC codec.

4. Experimental Results

To estimate the performance of the proposed algorithm, many experiments were accomplished. The HEVC video codec was chosen as a reference software. The

experiments considered several types of video sequences that cover diverse bitrates and resolutions. The chosen parameters of the HEVC codec are shown as in Table 1.

Table 1 : Parameter setting of the encoder

Profile	Baseline
Level	3
Number of coded frames	150
Number of reference frames	5
Search range	256
Rate Distortion Optimization (RDO)	On
Quantization Parameters	24, 36, 38, 44 and 46

For a fair testing, we conducted the tests and informed the amount of gotten acceleration compared to a state-of-the-art research [13]. On contrary, we mentioned the undesirable effects of the proposed algorithm on the bitrate rise and quality loss sides. The subjective and objective quality metrics were used to ensure the improvement gained by the proposed method.

4.1 Subjective Measurements

The subjective quality assessment was performed based on the Double Stimulus Continuous Quality Scale Method, ITU-R [24]. The situations for subjective and objective methods are used. The users' demographic, the methodology and evaluation of tests are also discussed.

Three different types of popular games were used for evaluation, Aquarium Toy, Death Ball Toy and Truck Toy. All these games were tested for both scenarios, the conventional method using HEVC and our proposed method using the improved HEVC [17]. All expected necessary information and encoding settings of the games have been chosen, evaluated, and sent to the video encoder. In addition, the part of game, the background, is also measured and sent to the video encoder. However, no further information is exploited from the game engine. In term of capturing the game video, the popular software called FRAPS was used as game scene capturer [25].

The game capturing criteria were accomplished using 30fps and a quarter pixel motion estimation. For video encoding, the reference software of HEVC was used with the determined thresholds that were mentioned above.

Firstly, we asked 30 users as participants in the study. The participants were students at college levels and they have no experience in video compression or assessment. The participants were asked questions to categorize them based on the demography, the demographic analysis is represented in Table 2.

Secondly, we asked participants to score their evaluations on nine different sets of video sequences from the three selected games. Each game has been encoded with three different bitrates as shown in Table3.

Table 2 : Demographics of subjective participants

Gaming experience				
Bad	Poor	Fair	Good	Excellent
2	2	7	8	13
Weekly hours game play				
≤ 5	6-10	11-20	21-30	>30
8	9	5	4	4
Gaming platform (already played on)				
PC	Console	Tablet	Cell phone	
2	8	2	18	

Table 3 : Selected bitrates for MOS

Game Name	Rate 1 (Kbps)	Rate 2 (Kbps)	Rate 3 (Kbps)
AquariumToy	3200	2800	1800
DeathBallToy	3400	2800	2000
TruckToy	3000	2800	2200

Each set of coded sequences have been coded using the traditional encoding method and our proposed method. To ensure the results are accurate, we encoded a 60 second video sequence with 2800 frames for each bitstream. Participants had to rate the quality as Excellent, Good, Fair, Poor or Bad. They asked to score in each set of video sequences. The video sequences have been displayed at their original resolution to disregard any possible distortion that may occur as a result of any scaling. The participants' watching distance was determined to be four times than the screen size as recommended in [24].

In the final stage, the Mean Opinion Score (MOS) was used as the subjective quality metric for both the conventional and proposed methods. The MOS for each combination methods and setting is calculated as an average of all MOS of all participants. The results of all MOS are shown in Table 4. As shown clearly, the MOS of the proposed method is about 0.82% less than the conventional coding one.

Table 4 : Subjective results - MOS

	AquariumToy			DeathBallToy			TruckToy			Average Change (%)
	Rate 1	Rate 2	Rate 3	Rate 1	Rate 2	Rate 3	Rate 1	Rate 2	Rate 3	
Conventional	4.71	4.52	4.97	4.89	4.76	4.53	4.85	4.63	4.69	-
Proposed	3.71	3.86	3.31	3.99	4.01	4.07	4.09	4.07	3.99	0.82

Moreover, the results of the proposed method showed reliable performance over the three different rates of the experiments. The average quality of the proposed method reduces negligibly by less than 0.3 dB compared to the conventional full method. Both the PSNR and the dB are an evidence on the high performance of the proposed method. As the motion estimation process is mainly performed for finding the best block, the entropy coding is used to calculate the residual between current and reference block.

4.2 Objective Measurements

The conventional compression method is shown in Figure 2 and the proposed method is shown in Figure 3. The Torque2D is defined in [23], this method is used as the testing game engine. The same three games as in subjective metric also used for the objective testing. For each game, an object has been chosen and sent to the video encoder. In addition to the selected object, the background is taken into attention for speeding up. Though, the information about the background is not provided by the game engine. The game scenes are captured in order to be sent to the encoder by FRAPS software [25, 26].

The capturing frame rate was set to 30fps. The HEVC reference software of full search is used for video encoding. To conduct the experiments, the thresholds are defined as follows: T1=22, T2=12, T3=8 and T4=4. The stated thresholds are to be determined; an initial simulation process needs to be conducted first on a set of video sequences outside the specified evaluation pool.

The simulation process should contain different contents and bitrates. The thresholds for any other video codecs need to be adjusted using the initial simulation process again. However, the threshold parameters are not variable after the simulation process. For performance, the proposed method was evaluated and tested in different settings. Table 1 displays the parameter settings that used in the video encoder in order to conduct the performance calculations. The captured videos are coded using Baseline with five different QP values and 150 coded frames.

The proposed method disclosed significant speed up compared to the conventional scheme. The total encoding time for the three selected video scenes are shown in Figure 4. As it shown in the figure, the total encoding time after implementing the proposed method is speeded up by up to 12.2% over the selected QPs. The significant speeding up of the ME for the proposed method is shown in Figure 5. As its clearly shown, the speed of the ME is increased by up to 32.9% and significantly outclassed the conventional encoding method.

The Rate-Distortion (RD) performance for the proposed method, conventional method and the comparison are shown in Figure 6. Significantly, the proposed method has achieved a close RD performance to the conventional method with negligible amount of increment. The proposed method adopted a technique to decrease the quality degradation occurred as a result of the miss-prediction. As a conclusion, losing less than 0.55 PSNR with achieving up to 12.2% speeding up compared to the conventional method should be a negligible issue.

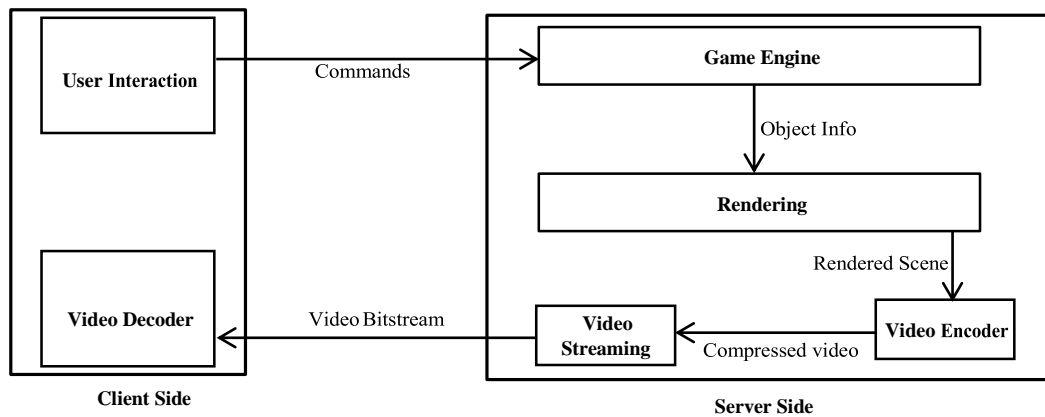


Fig. 2 : Conventional Compression Method

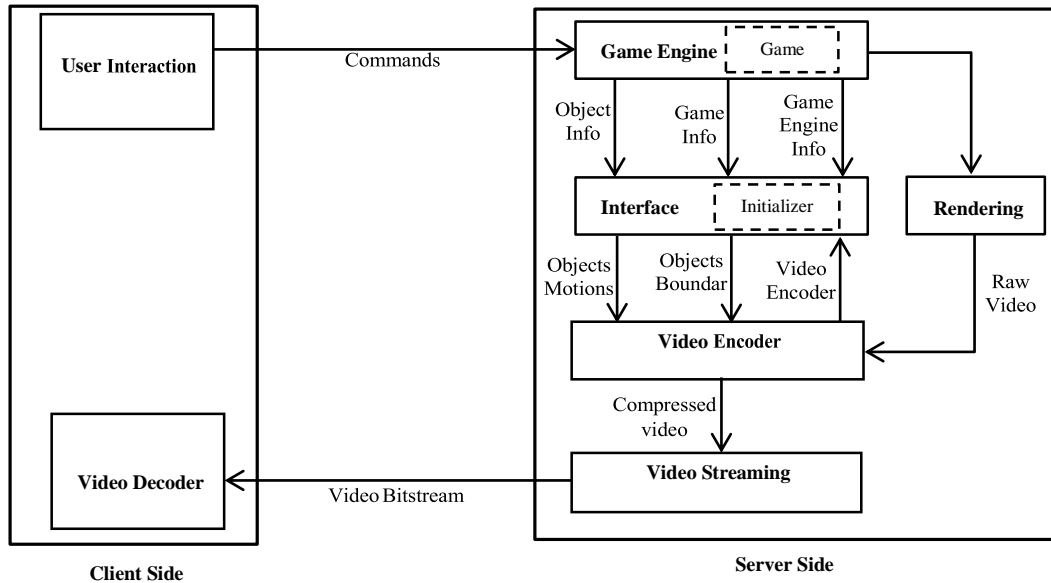


Fig. 3: Diagram of the Proposed Method

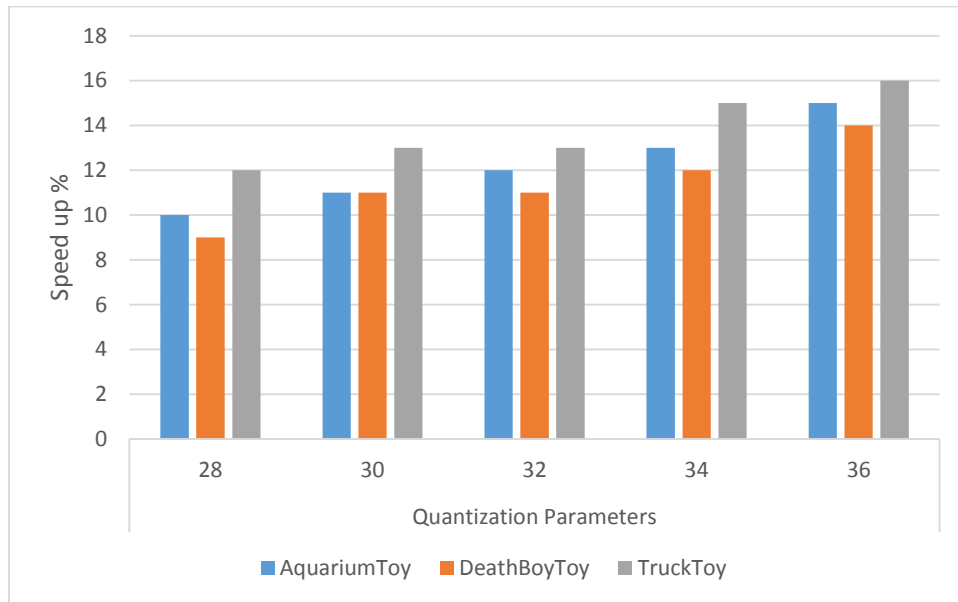


Fig. 4 : Encoding Time Speedup Compared to the Conventional Method

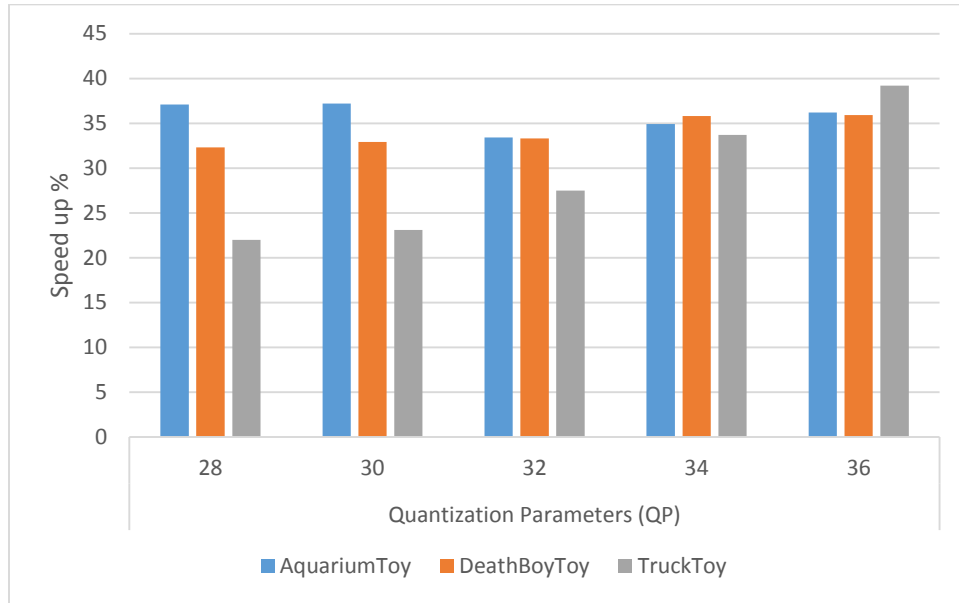


Fig. 5 : Motion Estimation Compared to the Conventional Method

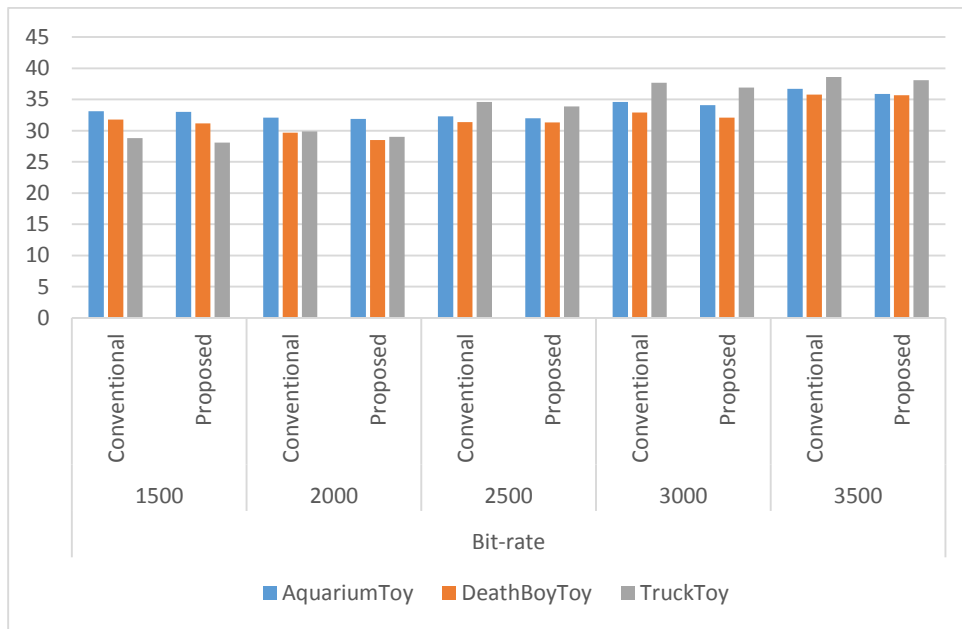


Fig. 6 : Rate Distortion Performance Compared to the Conventional Method

5. Conclusion

A new enhanced cloud gaming encoding technique was proposed in this research. The information of a game object inside the game engine was exploited to achieve this acceleration. In particular situations, this game object information was used in the video encoder to skip the time-consuming motion estimation process. Furthermore, the macroblocks are categorized either as foreground or background macroblocks. In case the macroblock is a background one, some measures to be checked, if met, the motion estimation is accelerated and the quality improvement also achieved. The experimentations showed that the proposed method is talented to speed up the motion estimation by up to 32.9% and the encoding time up to 12.2%. One significant profit of the proposed method is the ability of applying it inside any game object and with any video encoder. That is applicable with minimal required changes. Nevertheless, supporting different type of game objects such as 3D objects may require more research and could be one of the future work areas.

References

- [1] "Industry Facts," Entertainment Software Association (ESA). [Online]. Available: <http://www.theesa.com/facts/index.asp>.
- [2] "Gartner's Press Release," Gartner Inc. [Online]. Available: <http://www.gartner.com/newsroom/id/2614915>.
- [3] Mazhar, A.: Efficient Cloud Gaming Scheme Using Scene Objects Adaptation. International Journal of Computer Science and Information Technology (IJCSIT), Vol. 9, No 2. 2017
- [4] W. Cai, H. Chan, X.Wang, And V. Leung, "Cognitive Resource Optimization For The Decomposed Cloud Gaming Platform," Ieee Trans. Circuits Syst. Video Technol., Vol. 25, No. 12, Pp. 2038_2051, Dec. 2015.
- [5] D. Mishra, M. El Zarki, A. Erbad, C.-H. Hsu, And N. Venkatasubramanian, "Clouds + Games: A Multifaceted Approach," Ieee Internet Comput., Vol. 18, No. 3, Pp. 20_27, May 2014.
- [6] S.-P. Chuah, C. Yuen, And N.-M. Cheung, "Cloud Gaming: A Green Solution To Massive Multiplayer Online Games," Ieee Wireless Commun., Vol. 21, No. 4, Pp. 78_87, Aug. 2014.
- [7] W. Cai "Survey on Cloud Gaming: Future of Computer Games" 2169-3536 2016 IEEE. Volume 4 2016.

- [8] Yeng-Ting Lee, Kuan-Ta Chen, Yun-Maw Cheng, and Chin-Laung Lei, "World of Warcraft Avatar History Dataset" In Proceedings of ACM Multimedia Systems, 2011.
- [9] W. Cai, M. Chen, and V. Leung, "Toward Gaming as a Service," *IEEE Internet Comput.*, vol. 18, no. 3, pp. 12–18, 2014.
- [10] R. Shea, J. Liu, E. Ngai, and Y. Cui, "Cloud gaming: architecture and performance," *IEEE Netw.*, pp. 1–23, 2013.
- [11] T. D. Burd and R. W. Brodersen, "Processor Design for Portable Systems," *J. VLSI Signal Process. Syst.*, vol. 13, no. 2–3, pp. 203–221, 1996.
- [12] A. A. Mazhar and M. A. Mizher, "Efficient Video Encoding Acceleration for Cloud Gaming," *Journal of Theoretical and Applied Information Technology*. Vol 98, No. 18, 2020.
- [13] M. Semsarzadeh, M. Hemmati, A. Javadtalab, A. Yassine, and S. Shirmohammadi, "A Video Encoding Speed-up Architecture for Cloud Gaming," in *Cloud Gaming Systems and Networks in conjunction with IEEE International Conference on Multimedia & Expo*, 2014.
- [14] M. Khan, M. Shafique, M. Grellert, and H. Jörg, "Hardware-software collaborative complexity reduction scheme for the emerging HEVC intra encoder," in *Design, Automation and Test in Europe (DATE)*, 2013, pp. 125–128.
- [15] J. Zhou, D. Zhou, and S. Goto, "Alternating asymmetric search range assignment for bidirectional motion estimation in H.265/HEVC and H.264/AVC," *J. Vis. Commun. Image Represent.*, vol. 25, no. 5, pp. 1275–1286, Jul. 2014.
- [16] W. Lin, K. Panusopone, D. M. Baylon, M. Sun, and Z. Chen, "A Fast Sub-Pixel Motion Estimation Algorithm for H.264/AVC Video Coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 2, pp. 237–242, 2011.
- [17] An Efficient Fast HEVC Encoding Technique, A. Mazhar, "Journal of Theoretical and Applied Information Technology", Volume 90, Issue 2. 2016.
- [18] Performance evaluation of H. 265/MPEG-HEVC VP9 and H. 264/MPEG AVC video coding, A Mazhar, "The International Journal of Multimedia & Its Applications" (IJMA) 8 (1), 35-44, 2016.
- [19] S. Shi, C. Hsu, K. Nahrstedt, and R. Campbell, "Using graphics rendering contexts to enhance the real-time video coding for mobile cloud gaming," in *ACM Multimedia*, 2011, pp. 103–112.

- [20] N. Tizon, C. Moreno, and M. Preda, "ROI based video streaming for 3D remote rendering," in IEEE International Workshop on Multimedia Signal Processing (MMSP), 2011, pp. 1–6.
- [21] G. Cheung, A. Ortega, and T. Sakamoto, "Fast H. 264 mode selection using depth information for distributed game viewing," *Vis. Commun. Image Process.*, 2008.
- [22] M. R. Hossein zadeh Taher, H. Ahmadi, and M. R. Hashemi, "Power-Aware Analysis of H.264/AVC Encoding Parameters for Cloud Gaming," in *Cloud Gaming Systems and Networks in conjunction with IEEE International Conference on Multimedia & Expo*, 2014.
- [23] M. Hemmati, A. Javad talab, A. A. Nazari Shirehjini, S. Shir mohammadi, and T. Arici, "Game as video: Bit rate reduction through adaptive object encoding," in *ACM Workshop on Network and Operating Systems Support for Digital Audio and Video*, 2013, pp. 7–12.
- [24] ITU-R, "Methodology for the subjective assessment of the quality of television pictures".
- [25] "FRAPS, Real-time video capture & benchmarking." [Online]. Available: <http://www.fraps.com>.
- [26] "Torque2D Game Engine." [Online]. Available: <http://www.garagegames.com/products/torque-2d>.

Notes on contributors



Mosa I. Salah had finished his master degree from Al-The Arab Academy for Banking and Financial Sciences. Majoring in computer information systems since 2006, he has been a teacher and Researcher at ZUJ at the department of Computer Science and Multimedia Technology. His research interests, mainly in information retrieval, image processing and multimedia technology, he have successfully granted some research grants from the ZUJ.

Mr. Mosa is very active; he is Deputy Director of E-Learning & Open Educational Resources Center, also he

serves as a reviewer and technical committee member in different Journal and Conferences.

Ahmad A. Mazhar is an Associate Professor at the Department of Multimedia, SAE Amman, School of Creative Media, Amman, Jordan. His main teaching and research interests include Multimedia and its applications. He has published several publications in the field of video processing and cloud gaming.

Manar A. Mizher is an Assistant Professor at the faculty of computer Science and Informatics, Amman Arab University, Amman, Jordan. Her main teaching and research interests include computing theory and programming, mobile applications and networking. She has published several research articles in international journals and participated in international conferences.