Mellifluous Viewport Bitrate Adaptation for 360° Videos Streaming over HTTP/2

Nguyen Duc Thien[1], Tran Trieu Hai[2], Nguyen Viet Hung[1]
[1]Faculty of Information Technology, East Asia University of Technology, Bacninh, Vietnam
[2]Faculty of General Education - Hanoi Open University, Hanoi, Vietnam
Correspondence: Nguyen Duc Thien (e-mail: thiennd@eaut.edu.vn)

Communication: received 05 Jan 2024, revised 11 Mar 2024, accepted 11 Apr 2024
Digital Object Identifier: 10.32913/mic-ict-research.v2024.n2.1251

Abstract: Virtual reality provides an immersive experience for users. There has recently been much research on adaptive methods of playing 360° video over HTTP/2 to excite users. This paper analyzes users' behaviors and perceptions when watching 360° online videos. We also provide some measurements and evaluation of the main features of 360° video streaming. Experiments show that our method improves bitrate by up to 45.8% and buffer size counter-up to 25.0% compared to referenced methods.

Keywords: *VR, HTTP/2, 360° video, QoE, Viewport Adaptive Streaming.*

I. INTRODUCTION

Virtual classrooms are deployed more and more popularly, studied, and applied to many fields, especially in information technology. Virtual reality [1] or 360° video has been widely used in this field to meet human requirements in providing a realistic user experience, including learner experience, student satisfaction assessment, or even about the cost of streaming 360° video, which requires much higher bandwidth and a much higher 60fps frame rate [2–4].

Viewport Adaptive Streaming (VAS) applications were used to reduce bandwidth requirements. A VAS system that also delivers a portion of the currently available video to the user can result in a view with a higher bitrate in the provided area than in other segments [2, 5–7]. As a result, 360° video streaming will be scaled up based on the corresponding argument, as shown in Fig. 1.

On the other hand, since users tend to change their viewing direction when they are watching, this view will change over time during streaming sessions, or status will be updated to infer topics they are interested in. This may affect the viewer's perception described in [8]. Furthermore, bitrates in a VAS system should be adjusted to adapt to different views over time. Besides, studies [9, 10] have demonstrated when throughput being reduced due to network problems, users' viewing experience will be significantly reduced.

Besides, a large number of videos must be cached into versions to ensure smoothness when watching 360° videos in different network conditions, as shown in Fig. 1. Finally, the client will choose the version to download and play the video [9]. However, there is a limitation that the performance of VAS can be reduced due to the large buffer size [11]. Therefore, it is necessary to consider the viewport's adaptability and network condition.

In this paper, our objective is to assess the impact of users' head movement on existing methods, explicitly focusing on the following key aspects:

- Evaluation of current VAS methods: We aim to examine how existing VAS methods perform when there are variations in bandwidth and image quality uniformity within the users' field of view. By doing so, we intend to enhance the smoothness of video playback on the client side.
- Development of an adaptive algorithm: We propose an algorithm that dynamically adapts to changing network conditions, specifically in video streaming over HTTP/2 streams. This algorithm will ensure that the video streaming experience remains optimal by adjusting to fluctuations in available bandwidth.

By conducting this evaluation and developing the adaptive algorithm, we aim to contribute to improving video streaming systems, particularly in scenarios where head movement and varying network conditions can significantly impact the user experience. The remaining of this paper is the paper's content in the following sections.

The first presents related works in part II, part III details the proposed algorithm, and the fourth quarter evaluates

Figure 1: General chunk-based Viewport Adaptive Streaming system

the algorithm in part IV. Finally, the article concludes in part V.

II. RELATED WORK

In this section, we will briefly describe the context of some of the current HTTP video streaming methods of the scalable 360° video streaming method considered in the study.

1. Adaptive streaming over HTTP

Thanks to the strong development of information technology and 5G and 6G network systems today [12], 360° videos are becoming more popular. Moreover, 360° video requires a high resolution and viewport rate to cover an omnidirectional view. Therefore, in research [13], the authors propose an optimization method using video encoding layers, which increases the possibility of viewport adaptation over HTTP. Furthermore, research[2] shows that encoding and attention to regions for streaming are significantly improved to bring immersion to the viewer.

Besides, Streaming over HTTP has received much attention thanks to the Dynamic Adaptive Streaming over HTTP (DASH) feature. However, the development history is primarily private and not made public [14]. It is, therefore, difficult to compare. The paper [14] showed that using the open-source DASH method provides a basic assessment of the different segment lengths and their impact on the server. HTTP. According to a study [15], Streaming over HTTP (DASH) is a trusted standard for streaming video on the Internet because it provides a rich experience for users that the server side will support video streaming DASH-based content delivery.

2. Adaptive streaming over HTTP/2

With the growing popularity of HTTP/2 video streaming and the dominance of large amounts of video on the Internet, providers have moved toward high-quality video streaming. Besides, thanks to the development of HTTP/2, the protocol demonstrated that the multiplexing feature of HTTP/2 allows audio and video segments to be mixed in streaming [16, 17], bringing more solutions for video streaming over HTTP/2 protocol. The problem of mixed transmissions has been solved using monitoring techniques.

On the other hand, with data growth and high requirements from users, 360° video content is a big challenge [18], as they need to save bandwidth and optimize methods to improve stream quality and increase streaming speed. It is known that random transmission will slow down the process because it takes time to wait for processing. Therefore, researchers have taken advantage of providing more transmission lines to avoid user discomfort. HTTP/2 also shows flexibility for video streaming and improved QoE [19].

Furthermore, a method [20] has also shown that HTTP adaptive streaming plays a vital role in reducing video downtime and improving video quality by adjusting the quality of transmitted video segments according to the conditions of the network. The method also shows that buffering and bit rate factors are also important. Therefore, in this paper, we propose an algorithm to analyze and evaluate the quality of video streaming over HTTP/2 streams.

III. PROPOSED METHOD

In this section, we propose an algorithm that uses a sorting algorithm to select view regions, improving bitrate and cache quality with the HTTP/2 protocol.

1. Version Selection

Figure 2 shows that choosing the version to stream video content depending on user requirements and network conditions is significant. Although different videos are

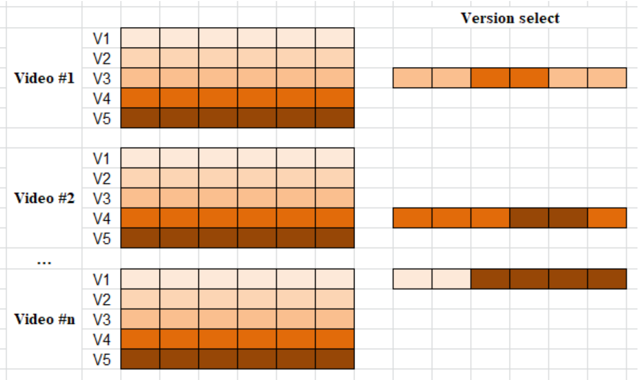


Figure 2: Selecting the streaming version

```
Algorithm 1: Version Selection
Input: $V_n, W_c, R$
Output: $\{v_n\}_{1\le n\le N}$
while $R > 0$ do
    for $v = 1$ to $V - 1$ do
        if $W_c < R$ then
            $W_c \in V_n$
            Calculate to Max $W_c \approx R$.
            $l_n \leftarrow l_n + 1$;
        end
    end
end
return $\{v_n\}_{1\le n\le N}$;
```

divided into different versions, from primary to high level, version transmission depends on network factors. Therefore, we hope that even when network conditions are limited, it still brings the best quality to viewers.

On the other hand, to perform such work, our method not only improves the user's viewport adaptation quality but also divides the video into segments. Each of them will have selected versions, which means that with good network quality conditions, the method that will achieve the best quality level, Version 5 in algorithm 1. However, the network's reality is always changing, and 360° video content is relatively large; studies [2] also show that the influence levels of each region also play an important role. We can improve the quality of important areas and limit unimportant regions. This means that areas within the mind will get much attention, and others will get less attention.

2. Chunk Selection

Research by the authors [21] has shown that high level caching is necessary. After dividing the versions, we analyze and divide them into Chunks. Each Chunk will have a different weight, corresponding to this weight is the improved quality. In algorithm 2, we improved the calculation to fill the buffer head, which means that some chunks of version 4 can be used for the version.

```
Algorithm 2: Chunk Weight Selection
for $c = 1$ to $C$ do
    $pivot$ = Buffer[$high$]
    for $low$ to $high$ do
        Calculate Buffer[$high$];
        if Buffer[low]> $pivot$ then
            Buffer[$high$] = Buffer[$low$]
            $low = low + 1$;
        end
    end
end
return Buffer[$high$];
```

TABLE I: Quantization Parameter (QP) of Chunk

Scalable	QP
Level of Enhancement of Chunk 01	38
Level of Enhancement of Chunk 02	32
Level of Enhancement of Chunk 03	28
Level of Enhancement of Chunk 04	24
Level of Enhancement of Chunk 05	20

For algorithm 2, we assume the desire to choose the best value named Buffer. To do this, we divide the versions into chunks as shown above; here, we conduct evaluation and analysis to select the corresponding threshold to achieve the best value, and the quick sort algorithm has been used. We calculate in this step. This method will not only improve time and help the system calculate better based on the buffer threshold, but also calculate and improve quality for regions with the largest number of viewers.

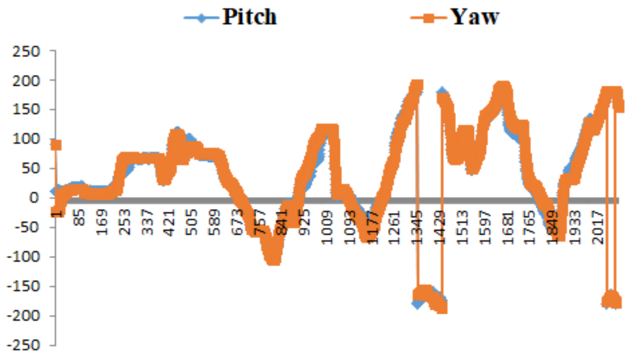
IV. EVALUATION

1. Experimental Settings

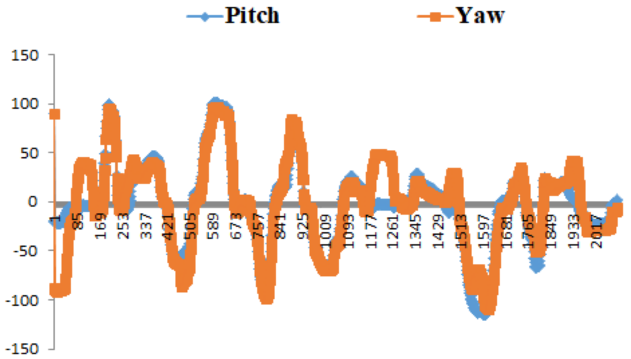
To evaluate the proposed solution, we will build a test system consisting of a server and a client; the server side

TABLE II: Chunk's average bitrate (kbps) for video experiments.

	Average chunk bitrate (kbps)		
Scalable Chunk	Roller Coaster	Diving	Venice
L. Enhance. Chunk 01	55	159	50
L. Enhance. Chunk 02	107	314	114
L. Enhance. Chunk 03	179	505	216
L. Enhance. Chunk 04	389	1037	484
L. Enhance. Chunk 05	633	1418	825



((a)) Head-Movement Trace 01



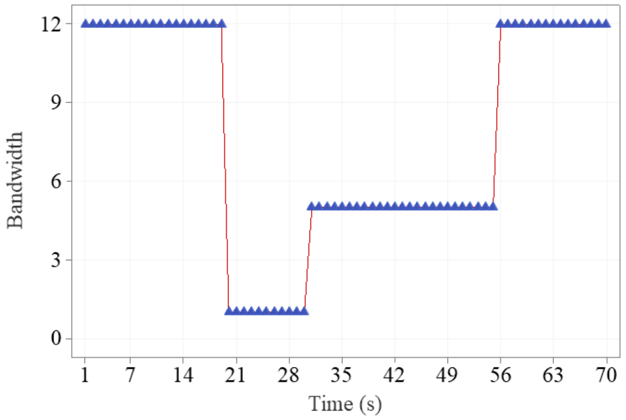
((b)) Head-Movement Trace 02

Figure 3: Head-movement traces under two simulations

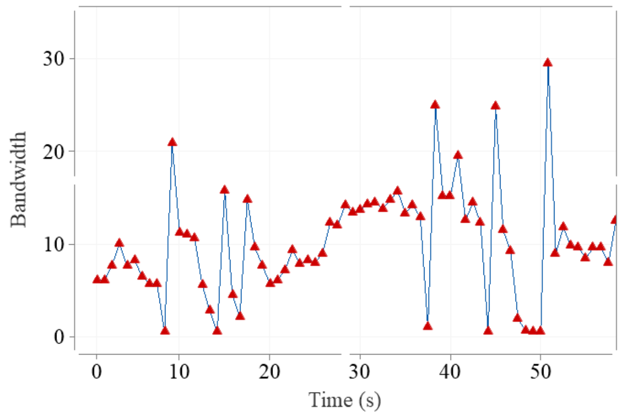
will be analyzed and divided into five different versions. Experiments are run with varying simulations of evaluation.

- First, we divided the experiment into two different head movements in Fig. 3.
- Second, we experiment with actual bandwidth and our designed bandwidth to test the proposed algorithm further in Fig. 4.
- Third, we simulate a system algorithm that typically limits intentional selection and instead runs automatically.

On the other hand, we use three videos, Rollercoaster, Diving, and Venice from [22]. Each 360° video will be converted to CubeMap format using 360Lib software [23], with the viewport set to 70s. The videos will be divided into 24 chunks; each chunk has the same resolution of 480x480, and each video will be divided into five versions, with each version being V_n $(0 < n < 6)$, V_1 the lowest version, the highest version is V_5, Let R is the network bandwidth, and W_c the weight of each chunk $(1 \leq c \leq 24)$. We use the HEVC [24] standard for encoding. The parameters we calculated for testing are in Tables I and II.



((a)) Bandwidth 01



((b)) Bandwidth 02

Figure 4: Bandwidth traces under two simulations

2. Performance Analysis

In this section, we create a standard algorithm (called No) and an improved algorithm (called proposed) to analyze the evaluations and three referenced methods including WZUQI [2], GLVP [4], and BL [25] with the following descriptions:

- No: we perform the normal setting as the Proposed algorithm. However, No will choose the version corresponding to the network conditions without chunk selection, while our algorithm will calculate the chunk. The chunk value then arranges the chunks in order and selects the appropriate chunk based on network conditions.
- WZUQI [2]: WZUQI is the algorithm focusing on the video's central center. Therefore, for our problem, the limitation is that it is not fixed but moves in the direction of the head, so for this algorithm, with continuously changing motion, the algorithm results are relatively limited.
- GLVP [4]: GLVP is the algorithm that focuses on

TABLE III: Average bitrate (Kbps) of reference vs. the proposed method

Videos	**Head movement trace #1**									
	WZUQI		**No**		**GLVP**		**BL**		**The Proposed**	
	BW #1	**BW #2**	**BW #1**	**BW #2**	**BW #1**	**BW #2**	**BW #1**	**BW #2**	**BW #1**	**BW #2**
RollerCoaster	468,60	**469,28**	202.63	202.46	503.75	494.26	454.94	459.13	555.46	547.36
Diving	231.77	198.56	505.54	509.74	343.87	343.68	433.10	415.69	519.70	513.19
Venice	222.10	185.26	232.95	233.13	462.14	454.36	436.86	428.24	514.00	514.77
Average	**307.49**	**284.37**	**313.71**	**315.11**	**436.59**	**430.77**	**441.63**	**434.36**	**529.72**	**525,11**

TABLE IV: Average bitrate (Kbps) of reference vs. the proposed method

Videos	**Head movement trace #2**									
	WZUQI		**No**		**GLVP**		**BL**		**The Proposed**	
	BW #1	**BW #2**	**BW #1**	**BW #2**	**BW #1**	**BW #2**	**BW #1**	**BW #2**	**BW #1**	**BW #2**
RollerCoaster	444.09	**444.15**	189.78	189.60	495.06	486.46	408.12	396.11	543.55	508.92
Diving	304.20	478.48	451.29	478.48	285.94	293.70	367.58	362.88	545.82	478.93
Venice	282.81	233.56	218.30	218.22	442.29	434.93	366.22	362.99	460.16	479.06
Average	**343.70**	**385.39**	**286.46**	**295.43**	**407.76**	**405.03**	**380.64**	**373.99**	**516.51**	**488.97**

TABLE V: Average total buffer size (s) of all test videos of reference vs. the proposed method under Head-movement trace 01

Videos	**Bandwidths**	**WZUQI**	**No**	**GLVP**	**BL**	**Proposed**
RollerCoaster	01	4.683	4.991	6.409	6.349	6.409
	02	4.676	4.988	6.417	6.350	6,417
Diving	01	4.872	4.549	6.367	6.358	6.383
	02	4.778	4.529	6.329	6.328	6.341
Venice	01	4.699	4.890	6.169	6.143	6.169
	02	4.692	4.890	6.172	6.140	6.172
Average Buffer		**4.734**	**4.806**	**6.310**	**6.278**	**6.315**

TABLE VI: Average total buffer size (s) of all test videos of reference vs. the proposed method under Head-movement trace 02

Videos	**Bandwidths**	**WZUQI**	**No**	**GLVP**	**BL**	**Proposed**
RollerCoaster	01	444.094	189.777	495.061	408.120	543.547
	02	444.150	189.595	486.461	396.113	508.918
Diving	01	304.200	451.289	285.942	367.580	445.818
	02	264.239	478.475	293.704	362.878	478.934
Venice	01	282.812	218.301	442.287	366.219	460.162
	02	233.556	218.220	434.929	362.990	479.055
Average Buffer		**328.842**	**290.943**	**406.397**	**377.317**	**486.072**

improving and estimating the typical view of the VAS system. In general, the algorithm is the same as currently mentioned. Still, for the proposed method, in addition to estimating, the algorithm needs to focus on calculating load calculation and buffer filling.

- BL [25]: BL is a proposed tiling-based 360° video streaming method that maintains smooth viewport quality despite network bandwidth fluctuations and user head movements. BL dynamically adjusts the number of extension tiles based on viewport prediction errors to ensure a smooth viewport bitrate across video segments. In addition, the total bitrate allocated to tiles is chosen to accommodate sudden changes in network bandwidth. Finally, a tile selection algorithm helps to reduce intra-segment bitrate variations.

On the other hand, according to recent research, the most recent scalable 360° video streaming methods adapt the video to variations in network throughput and user behavior. None of the methods listed above perform well in either measured parameters: buffer size or average view bitrate. Our solution is to create a stable connection as network bandwidth changes while improving user experience. Furthermore, improving these two parameters will fine-tune the video streaming quality and expertise. Proper consideration and adjustment of these parameters help ensure that videos play smoothly and are of good quality on viewing devices.

a) Viewport bitrate

In this paper, we measured the viewport bitrate parameter to evaluate the human-perceived quality of the same viewport when streaming 360° videos over the same HTTP/2 stream. This parameter also shows that the higher the value of the bit rate, the better the perceived quality of the service. However, we also see that increasing the quality too high also affects the user's perception when it goes up and down suddenly.

Experiments show that the average bit rate of the proposed method improves better than the referenced methods in Table III and IV. For head movements #1, the proposed method improves up to 40.8% compared to method No. Besides, the most negligible improvement is 17.3% for the BL method. Meanwhile, for head movements #2, our method can improve up to 44.5%, while the lowest improvement is 17.2% in the GLVP method.

b) Buffer size

The purpose of our experiment is to understand cache activation and cache impact parameters. A bigger storage buffer means faster and better data streaming for users. This parameter also indicates that buffering has a net impact on video streaming because the quicker the buffer transmits, the more data will be downloaded.

Furthermore, we also found that this effect holds under strong bandwidth fluctuations and normal throughput. In practical testing, we found that the view bit rate fluctuates greatly under practically zero network fluctuations and does not change much with normal throughput. However, our algorithm is stable in these two cases of fluctuations.

Experiments show that five methods have buffer sizes between 4.7s and 6.4s. Although, in this test, the maximum allowed cache threshold is 7s. However, neither technique has reached the maximum threshold; this is considered a limitation for future expansion. However, the experiment shows that although the proposed method has not reached the maximum threshold, filling the buffer high and close to the threshold value is relatively good. Compared to the proposed method, the reference methods have deficient buffer levels, reaching 5,529s for head movement #1 and 5,535s for head movement #2. While measuring, the proposed method reaches the most minor threshold of 6,159s. Overall, with the current buffer level, the proposed method can improve up to 25%; we can see that the buffer level is shown in Tables V and VI.

In summary, the method has also improved and expanded on the commonly used methods. Further research on multi-directional motion is needed to gain a broader perspective.

V. CONCLUSIONS

In this paper, we studied 360° video streaming with extensive content via HTTP/2 protocol. The proposed method performs significantly more than the conventional method based on bit rate and buffer parameters. Overall, the proposed method improves significantly compared to the referenced methods; for head movements, the bit rate ratio can improve from 17.2% to 45.8%. Besides, the proposed method to enhance the buffer can achieve a maximum improvement of up to 25.0%.

REFERENCES

[1] T. Stockhammer. MPEG Immersive Media. Accessed: 2023-12-27. [Online]. Available: https://www.itu.int/en/ITU-T/studygroups/2017-2020/16/Documents/ws/201701ILE/S2P2-1701-01-19-MPEG-Immersive-Media-Thomas-Stockhammer.pdf

[2] V. H. Nguyen, N. N. Pham, C. T. Truong, D. T. Bui, H. T. Nguyen, and T. H. Truong, "Retina-based quality assessment of tile-coded 360-degree videos," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 9, no. 32, 2022.

[3] Huewei. (2017) Virtual reality/augmented reality white paper. [Online]. Available: http://www-file.huawei.com/-/media/CORPORATE/PDF/ilab/vr-ar-en.pdf

[4] H. Nguyen, T. N. Dao, N. S. Pham, T. L. Dang, T. D. Nguyen, and T. H. Truong, "An accurate viewport estimation method for 360 video streaming using deep learning," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 9, no. 4, pp. e2–e2, 2022.

[5] C. Ozcinar, A. De Abreu, and A. Smolic, "Viewport-aware adaptive 360° video streaming using tiles for virtual reality," in *2017 IEEE International Conference on Image Processing (ICIP)*, Beijing, China, Sep. 2017, pp. 2174–2178.

[6] D. Nguyen, N. V. Hung, N. T. Phong, T. T. Huong, and T. C. Thang, "Scalable multicast for live 360-degree video streaming over mobile networks," *IEEE Access*, vol. 10, pp. 38 802–38 812, 2022.

[7] N. V. Hung, P. H. Thinh, N. H. Thanh, T. T. Lam, T. T. Hien, V. T. Ninh, and T. T. Huong, "Lvsum-optimized live 360 degree video streaming in unicast and multicast over mobile networks," in *2023 IEEE 15th International Conference on Computational Intelligence and Communication Networks (CICN)*, 2023, pp. 29–34.

[8] W. L. S. Ramos, M. M. Silva, E. R. Araujo, A. C. Neves, and E. R. Nascimento, "Personalizing fast-forward videos based on visual and textual features from social network," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020, pp. 3260–3269.

[9] T. C. Thang, H. T. Le, A. T. Pham, and Y. M. Ro, "An evaluation of bitrate adaptation methods for HTTP live streaming," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 4, pp. 693–705, 2014.

[10] H. T. T. Tran, N. P. Ngoc, A. T. Pham, and T. C. Thang, "A Multi-Factor QoE Model for Adaptive Streaming over Mobile Networks," in *2016 IEEE Globecom Workshops (GC Wkshps)*, Dec 2016, pp. 1–6.

[11] D. V. Nguyen and H. T. T. Tran and T. C. Thang, "Impact of delays on 360-degree video communications," in *2017 TRON Symposium (TRONSHOW)*, 12 2017, pp. 1–6.

[12] F. Muheidat, K. Dajani, and L. A. Tawalbeh, "Security concerns for 5g/6g mobile network technology and quantum communication," *Procedia Computer Science*, vol. 203, pp. 32–40, 2022, 17th International Conference on Future Networks and Communications / 19th International Conference on Mobile Systems and Pervasive Computing / 12th International Conference on Sustainable Energy Information Technology (FNC/MobiSPC/SEIT 2022), August 9-11, 2022, Niagara Falls, Ontario, Canada. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050922006123

[13] X. Chen, D. Wu, and I. Ahmad, "Optimized viewport-adaptive 360-degree video streaming," *CAAI Transactions on Intelligence Technology*, vol. 6, no. 3, pp. 347–359, 2021.

[14] S. Lederer, C. Müller, and C. Timmerer, "Dynamic adaptive streaming over http dataset," in *Proceedings of the 3rd multimedia systems conference*, 2012, pp. 89–94.

[15] J. Kua, G. Armitage, and P. Branch, "A survey of rate

adaptation techniques for dynamic adaptive streaming over http," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1842–1866, 2017.

[16] H. Wu, X. Li, G. Wang, G. Cheng, and X. Hu, "Resolution identification of encrypted video streaming based on http/2 features," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 19, no. 2, feb 2023. [Online]. Available: https://doi.org/10.1145/3551891

[17] V. H. Nguyen, D. T. Bui, T. L. Tran, C. T. Truong, and T. H. Truong, "Scalable and resilient 360-degree-video adaptive streaming over http/2 against sudden network drops," *Computer Communications*, vol. 216, pp. 1–15, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S014036642400001X

[18] M. Mahmoud, S. Rizou, A. S. Panayides, N. V. Kantartzis, G. K. Karagiannidis, P. I. Lazaridis, and Z. D. Zaharis, "A survey on optimizing mobile delivery of 360° videos: Edge caching and multicasting," *IEEE Access*, 2023.

[19] A. Kumar, S. Banerjea, and M. Pandey, "Churn-tolerant cdn at the edge for adaptive video streaming: towards multi-connection approach using http/2," *International Journal of Information Technology*, vol. 15, no. 7, pp. 3491–3497, 2023.

[20] M. Nguyen, H. Amirpour, F. Tashtarian, C. Timmerer, and H. Hellwagner, "Performance analysis of h2br: Http/2-based segment upgrading to improve the qoe in has," *Multimedia Tools and Applications*, pp. 1–35, 2023.

[21] N. V. Hung, T. D. Chien, N. P. Ngoc, and T. H. Truong, "Flexible http-based video adaptive streaming for good qoe during sudden bandwidth drops," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 10, no. 2, pp. e3–e3, 2023.

[22] X. Corbillon, F. De Simone, and G. Simon, "360-degree video head movement dataset," in *Proceedings of the 8th ACM on Multimedia Systems Conference*, ser. MMSys'17. Taipei, Taiwan: Association for Computing Machinery, 2017, p. 199–204.

[23] JVET, "360lib," https://jvet.hhi.fraunhofer.de/, October 2016. [Online]. Available: https://jvet.hhi.fraunhofer.de/

[24] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, "Overview of shvc: Scalable extensions of the high efficiency video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 20–34, 2015.

[25] H. L. D. Huong, D. V. Nguyen, T. T. Huong, P. N. Nam, and T. C. Thang, "Smooth viewport bitrate adaptation for 360 video streaming," in *2019 6th NAFOSTED Conference on Information and Computer Science (NICS)*. IEEE, 2019, pp. 512–517.

NGUYEN DUC THIEN received the B.Sc.degree in Informatics of Vietnam National University, Hanoi, in 2005, the M.Sc. degree Master of Data transmission and computer network of Pots and Telecommunications Institute of Technology, Vietnam, in 2011.

He is a lecturer in the Faculty of Information Technology at the East Asia University of Technology, Trinh Van Bo Street, Nam Tu Liem District, Hanoi, Vietnam.

His research interests include multimedia communications, network security, artificial intelligence, and applications.

Email: thiennd@eaut.edu.vn

TRAN TRIEU HAI received a B.Sc.degree in Bachelor of Information Technology, Faculty of Information Technology, Hanoi National University, Vietnam, in 2004 and an M.Sc. degree from Master of Information Management in Information Management, Shu-te University, Taiwan, in 2013.

He is also an IT Lecturer at the Faculty of General Education - at Hanoi Open University, Nguyen Hien Street, Hai Ba Trung District, Hanoi, Vietnam.

His research interests include Network Security, Artificial Intelligence, Natural Language Processing, and applications.

Email: haitt@hou.edu.vn

NGUYEN VIET HUNG received a B.Sc.degree in Bachelor of Informatics pedagogy from the Faculty of Engineering Technology of Ha Tinh University, Vietnam, in 2012, the M.Sc. degree Master of Information Technology from the Faculty of Information Technology of Ho Chi Minh City University of Technology, Vietnam, in 2016, and has been learning the Ph.D. degree in Telecommunications Engineering from the Hanoi University of Science and Technology, Vietnam.

He is also a lecturer in the Faculty of Information Technology at the East Asia University of Technology, Trinh Van Bo Street, Nam Tu Liem District, Hanoi, Vietnam.

His research interests include multimedia communications, network security, artificial intelligence, QoE/QoS guarantee for network services, Virtual Reality, Natural Language Processing, Data Mining, and applications.

Email: hungnv@eaut.edu.vn