LETTER *Special Section on Enriched Multimedia—Making Multimedia More Convenient and Safer—*

Adaptive Tiling Selection for Viewport Adaptive Streaming of 360-degree Video

Duc V. NGUYEN[†a)], **Huyen T. T. TRAN**[†], *Nonmembers*, *and* **Truong Cong THANG**[†], *Member*

SUMMARY 360-degree video is an important component of the emerging Virtual Reality. In this paper, we propose a new adaptation method for tiling-based viewport adaptive streaming of 360-degree video. The proposed method is able to dynamically select the best tiling scheme given the network conditions and user status. Experiments show that our proposed method can improve the viewport quality by up to 2.3 dB compared to a conventional fixed tiling method.

key words: *360-degree video, viewport adaptive streaming, adaptive tiling*

1. Introduction

360-degree video (360 video for short) is an integral part of Virtual Reality, which can provide immersive viewing experience to users [1]. However, streaming of 360 video over bandwidth-constrained networks is not an easy task because 360 videos require much higher bandwidth than traditional videos [2]. For effective streaming of 360 video, viewport adaptive streaming has been introduced. The basic idea is to deliver *viewport*, which is the visible part according to current position of the user's head, at high quality, whereas the other parts are delivered at lower quality [3].

It should be noted that, in practice, video adaptation is carried out on the basis of adaptation intervals (called temporal segments in DASH [4]). So, even though the network delay could be very small today (e.g. a few milliseconds), the streaming system can respond to changes after each interval only.

Tiling-based approach is the most popular approach for realizing viewport adaptive streaming [3], [5]. In tiling-based viewport adaptive streaming, the original video is spatially partitioned into regions called *tiles*. Each tile is further encoded into several versions of different quality levels. Given the user's viewport, the tiles overlapping the viewport (called visible tiles) are streamed at high quality while the other tiles at lower quality [3]. Figure 1 shows a tiling example and the visible tiles corresponding to a specific viewport.

In the current literature, typical tiling schemes include 4x3 [6], 6x4 [5], 8x4 [7], and 8x8 [3]. Some studies investigate good tiling schemes from the server's point of view, by considering the tradeoff between coding efficiency and the number of tiles [5], [7]. However, no metric to decide the optimal tiling scheme has been considered.

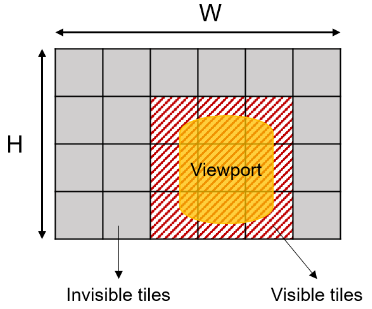

Fig. 1 Illustrations of tiling scheme, viewport, and visible tiles.

In our opinion, the existing studies on optimal tiling scheme miss two important issues. First, the optimal tiling scheme should be mainly considered from the client's point of view. That is, it should be based on the quality performance measured at the client, not at the server.

Second, a tiling scheme so far is fixed during a whole streaming session. Intuitively, when the head-moving speed is small, one should use high tiling granularity (i.e., large number of tiles) as it can reduce the amount of redundant pixels, which are the pixels belonging to high quality tiles but not in the viewport. Meanwhile, redundant pixels in case of low tiling granularity (i.e., small number of tiles) can help cope with a high head-moving speed. Since the user head movement is generally varying throughout a streaming session, using a fixed tiling scheme as in existing studies might lead to non-optimal viewport quality.

In this context, it is important to answer some related questions such as "what is the benefit of adaptive tiling compared to fixed tiling?", "which tiling should be selected given a speed of head movement?", or "if fixed tiling is preferred in a given context, what is the best tiling scheme for the client?". To the best of our knowledge, there is no previous work that investigate these questions. As this is a very complex problem, the aim of this seminal work is essentially to raise its importance for research.

For that purpose, we will first formulate a general problem for adaptive tiling in 360 video streaming. Then, correspondingly a simple solution to that problem is devised. Experiment results show that adaptive tiling can improve the average viewport quality by up to 2.3 dB compared to a fixed tiling solution. It is also found that among fixed tiling schemes, 4x3 tiling achieves the lowest viewport quality and thus should not be used.

The remainder of the paper is organized as follows.

Manuscript received March 26, 2018.
Manuscript revised August 25, 2018.
Manuscript publicized October 19, 2018.

[†] The authors are with the University of Aizu, Aizuwakamatsu-shi, 965–8580 Japan.
a) E-mail: nvduc712@gmail.com

DOI: 10.1587/transinf.2018MUL0001

Section 2 presents our proposed method. The proposed method is evaluated in Sect. 3. Finally, the paper is concluded in Sect. 4.

2. Adaptive Tiling Selection Method

In this section, we first present the problem formulation of adaptive tiling selection. Based on that, a solution to the problem is described.

2.1 General Problem Formulation

In our system, the tiling scheme is decided every adaptation interval. Each adaptation interval consists of L video frames. The original 360 video is represented as a rectangular video with a width of W (pixels) and a height of H (pixels) using Equirectangular projection [8]. There are K available tiling schemes. The tiling scheme C_k $(1 \leq k \leq K)$ is defined as a grid partition of $T_k = M_k \times N_k$ equally sized tiles (i.e. N_k rows and M_k columns). Each rectangular tile has a width of W/M_k and a height of H/N_k. C_k is also denoted by $M_k \times N_k$. Each tile is encoded into V versions. Version v $(1 \leq v \leq V)$ of tile t $(1 \leq t \leq T_k)$ of tiling scheme C_k $(1 \leq k \leq K)$ of the l^{th} frame $(1 \leq l \leq L)$ has a bitrate of $R_t^k(v, l)$ and a distortion of $D_t^k(v, l)$. In this study, the distortion is measured by the Mean Square Error (MSE). MSE and bitrate values of tiles can be provided as metadata [3]. It should be noted that, as shown in our previous study [9], PSNR (which is convertible from/to MSE) is still very effective to represent the viewport quality for users.

Suppose that, at a given time, the server needs to adapt an adaptation interval to meet a bandwidth constraint R^c. Denote $\boldsymbol{P}$ the set of the viewport positions when the user watches the frames of the considered adaptation interval. The tiling selection problem can be formulated as follows.

Find a tiling scheme C_k and a version v_t of each tile t so as to minimize the quality objective VQ which is a function of tiles' distortions and viewport positions.

$$VQ = f(\{D_t^k(v, l), 1 \leq t \leq T_k, 1 \leq l \leq L\}, \boldsymbol{P}) \tag{1}$$

and satisfy the bitrate constraint

$$\sum_{l=1}^{L} \sum_{t=1}^{T_k} R_t^k(v_t, l) \leq R^c. \tag{2}$$

2.2 Optimal Solution

In the following, we will describe the computation of the quality objective and the method to decide the tiling scheme and the version of each tile. The quality objective VQ is computed as follows. First, the viewport distortion $VQ(l)$ of the l^{th} frame $(1 \leq l \leq L)$ is calculated as the weighted average distortion of the visible tiles as follows.

$$VQ(l) = \sum_{t=1}^{T_k} w_t(l) \times D_t^k(v_t, l). \tag{3}$$

Here, the weight $w_t(l)$ indicates how much tile t $(1 \leq t \leq T_k)$ overlaps the viewport at the l^{th} frame. Denote $N(t, l)$ the area of the overlapped area of tile t and N_{vp} the total area of the viewport, the value of $w_t(l)$ is computed as follows.

$$w_t(l) = \frac{N(t, l)}{N_{vp}}. \tag{4}$$

It can be note that the value of $w_t(l)$ depends on the head-moving speed. The quality objective VQ is then computed as the average viewport distortion over all frames of the adaptation interval as follows.

$$VQ = \frac{1}{L} \sum_{l=1}^{L} VQ(l). \tag{5}$$

For selecting the version of each tile, a simple tile selection procedure is applied. Basically, the procedure selects the lowest version for the invisible tiles, and selects the highest possible version for visible tiles. Here, the invisible tiles are also delivered to the client because we have found that some users suddenly changes his/her viewing direction (to the left/right or the back). If the invisible tiles are not sent, the user might experience blank blocks in the viewport. Currently, similar to the previous studies of [5], [6], the visible tiles are determined using the viewport at the first frame of the adaptation interval.

As the problem space is small, a full-search procedure is used to find the optimal tiling scheme as follows.

- **Step 1**: For each tiling scheme
  - Classifying visible tiles and non-visible tiles of the interval.
  - Assigning the lowest version to all non-visible tiles.
  - Finding the highest possible version for all visible tiles that is permitted by the bandwidth constraint R^c.
  - Caclulating the quality objective VQ using (3)(4)(5).
- **Step 2**:
  - Selecting the tiling scheme that achieves the lowest quality objective.
  - Recording the tile versions decided in Step 1 for that tiling scheme.

With our current implementation, the average calculation time of the proposed method is less than 1ms. Thus, it is able to apply in real-time adaptation.

3. Experimental Results

In our experiment, we use a 360-degree video named *Timelapse*, which is provided in [10]. The video is of Equirectangular format, having a duration of 60 seconds, a resolution of 3840x1920 (4K), and a frame rate of 24 fps. The Field of View (FoV) of the viewport is 90 horizontal degrees x 90 vertical degrees. We consider $K = 4$ tiling schemes of 4x3, 6x4, 8x4, and 8x8. Each tile is encoded into 7 versions corresponding to 7 quantization parameter values of 24, 28, 32, 36, 40, 44, and 48 using HEVC. The adaptation interval and the buffer size are both set to 1 second. We use 10 head movement traces which are recorded from 10 different users watching the considered video [10]. The CDFs of the angular speed per interval of each trace are shown in Fig. 2. It can be seen that the head movements vary among the users. To clearly see the effect of tiling, we assume that the viewport positions during each adaptation interval are known in advance. Also, the network bandwidth is constant during each streaming session. Three bandwidth values of 2 Mbps, 4 Mbps, and 6 Mbps, are used. The network delay is set to 10ms.

The proposed method is compared to the conventional method (essentially Step 1 above) in which the tiling scheme is fixed during the streaming session. As PSNR has been proved as most suitable metric for evaluating 360 video [9], viewport PSNR, which measures the quality of a rendered viewport in the sphere domain, is adopted as the performance metric in this study. It is calculated as the PSNR between the rendered viewport and the original viewport. Besides, it is possible that the boundaries between tiles might be visible and cause negative impacts to user experience. This issue will be studied in our future work.

Figure 3 shows the selected tiling schemes and the average angular speeds of 7^{th} – 55^{th} adaptation intervals of our method under trace #6 when the bandwidth is 4 Mbps. It can be seen that our method can dynamically adapt the tiling scheme. Specifically, a higher number of tiles is selected when the user head movement speed decreases (e.g., 7^{th} – 10^{th} intervals). On the other hand, a small number of tiles is used when the head movement speed is high. Figure 4 shows the percentage of tiling schemes decided by the proposed method with the 10 head traces when the bandwidth is 4 Mbps. It can be seen that the selected tiling schemes are strongly correlated to the movement of each trace. For example, 4x3 tiling is not selected in case of traces #10 and #2 as these traces have very low movement speed. Meanwhile, in case of trace #6 where the angular speed is mostly in the range of 30–70 (degree/sec), 6x4 tiling is the most selected scheme. When the movement speed spreads out evenly between 0 and 90 (degree/sec) as in case trace #9, the portions of the tiling schemes are very similar.

Tables 1, 2, and 3 show the gain of adaptive tiling compared to fixed tilings when the bandwidth is 2Mbps, 4Mbps, and 6Mbps respectively. Note that the last two rows in these tables summarize the average and maximum values of improvement among the 10 traces. Figure 5 shows the average viewport PSNR of adaptive tiling and fixed tilings, where the average is computed over the three bandwidth values.

It can be seen that, by adapting the tiling scheme during a streaming session, our proposed method always achieves higher viewport quality than the conventional method. Especially, the improvement is consistent over all three bandwidth values. In general, the improvement compared to 4x3 tiling is highest, while improvements compared to 6x4, 8x4, and 8x8 tilings are similar. As seen in Table 2, the proposed method can improve the average viewport PSNR by up to

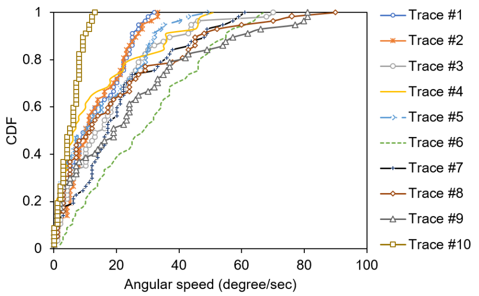

Fig. 2 CDFs of the angular speed per interval of 10 head traces.

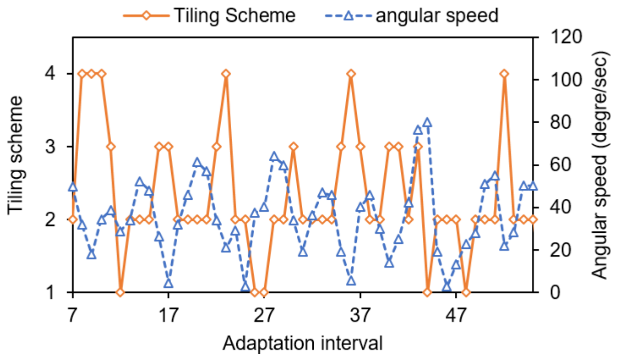

Fig. 3 Selected tiling schemes (1: 4x3, 2: 6x4, 3: 8x4, 4: 8x8) of the proposed method under trace #6 when the bandwidth is 4 Mbps.

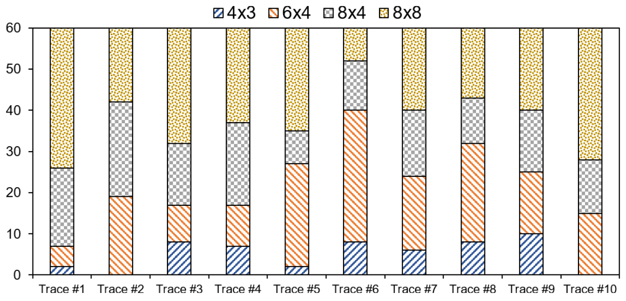

Fig. 4 Percentage of each selected tiling scheme of the proposed method when the bandwidth is 4 Mbps.

Table 1 Quality gain of adaptive tiling over fixed tiling when the bandwidth is 2 Mbps. The last two rows summarize the average and max values of improvement.

Trace	Quality gain (dB)			
	vs. 4x3	vs. 6x4	vs. 8x4	vs. 8x8
#1	1.6	0.7	0.6	0.3
#2	1.6	0.7	0.3	0.4
#3	1.2	0.7	0.4	0.4
#4	0.9	0.7	0.3	0.4
#5	1.4	0.5	0.5	0.6
#6	1.1	0.3	0.4	0.6
#7	1.3	0.5	0.5	0.6
#8	1.9	0.5	0.5	0.8
#9	1.4	0.7	0.5	0.6
#10	1.9	0.5	0.5	0.4
Average	1.4	0.6	0.4	0.5
Max	1.9	0.7	0.6	0.8

Table 2 Quality gain of adaptive tiling over fixed tiling when the bandwidth is 4 Mbps. The last two rows summarize the average and max values of improvement.

Trace	Quality gain (dB)			
	vs. 4x3	vs. 6x4	vs. 8x4	vs. 8x8
#1	1.9	0.9	0.7	0.3
#2	2.1	0.8	0.4	0.3
#3	1.6	1.0	0.6	0.4
#4	1.2	0.8	0.4	0.4
#5	1.8	0.6	0.8	0.7
#6	1.3	0.4	0.7	0.9
#7	1.7	0.8	0.8	0.9
#8	2.2	0.6	0.7	0.8
#9	1.6	0.8	0.7	0.7
#10	2.3	0.8	0.4	0.2
Average	1.8	0.8	0.6	0.5
Max	2.3	1.0	0.8	0.9

Table 3 Quality gain of adaptive tiling over fixed tiling when the bandwidth is 6 Mbps. The last two rows summarize the average and max values of improvement.

Trace	Quality gain (dB)			
	vs. 4x3	vs. 6x4	vs. 8x4	vs. 8x8
#1	1.5	0.8	0.6	0.2
#2	1.9	0.8	0.4	0.3
#3	1.2	0.8	0.5	0.4
#4	1.0	0.6	0.3	0.4
#5	1.5	0.5	0.7	0.7
#6	1.2	0.3	0.7	0.8
#7	1.3	0.7	0.7	0.8
#8	1.6	0.5	0.5	0.6
#9	1.2	0.7	0.6	0.6
#10	1.9	0.9	0.3	0.1
Average	1.4	0.7	0.5	0.5
Max	1.9	0.9	0.7	0.8

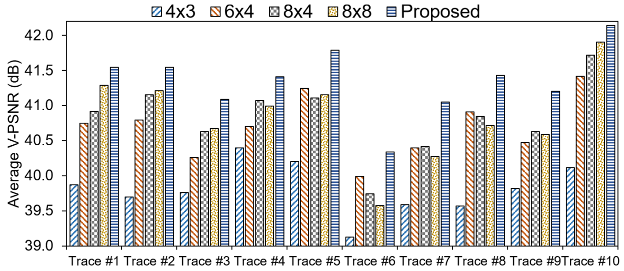

Fig. 5 Average viewport PSNR per trace of our proposed method and the conventional method (averaged over three bandwidth values).

Table 4 Number of traces in which a fixed tiling scheme achieves the highest performance compared to other fixed tilings.

Tiling Scheme	4x3	6x4	8x4	8x8
Number of Traces	0	3	3	4

2.3 dB compared to 4x3 tiling, and up to 0.8 ∼ 1.0 dB compared to other tilings.

Table 4 shows the number of head movement traces in which a fixed tiling scheme achieves the highest performance compared to the other fixed tilings. It can be noted that, though having the highest coding efficiency, 4x3 tiling scheme does not achieve the highest viewport PSNR for any traces. This suggests that 4x3 tiling is not effective and thus should not be used. This finding in fact cannot be found if the tiling is considered from the server's point of view. We can also see that the tiling schemes of 6x4, 8x4, and 8x8 have similar number of traces where they achieve the highest viewport PSNR. From this result, 6x4 tiling seems to be the best choice of fixed tiling, due to its high PSNR value and lowest number of tiles. A related tiling approach is using overlapped tiles (e.g. [11]) to cope with user head movements. Here, one may adjust both the size and the overlapped parts of tiles. This approach is reserved for our future work.

4. Conclusion

In this paper, we have presented an adaptation method for viewport adaptive streaming of 360 video that is able to dynamically adapt the tiling scheme based on the user's head movements and the network bandwidth. It was shown that adaptive tiling can improve the average viewport quality by up to 2.3 dB. For future work, we will investigate optimal tiling selection scheme when applying other tile selection methods.

References

[1] H.T.T. Tran, N.P. Ngoc, C.T. Pham, Y.J. Jung, and T.C. Thang, "A subjective study on QoE of 360 video for VR communication," Proc. 19th IEEE MMSP, Luton, UK, pp.1–6, Oct. 2017.

[2] X. Corbillon, G. Simon, A. Devlic, and J. Chakareski, "Viewport-adaptive navigable 360-degree video delivery," Proc. 2017 IEEE ICC, Paris, France, pp.1–7, May 2017.

[3] D.V. Nguyen, H.T.T. Tran, A.T. Pham, and T.C. Thang, "A new adaptation approach for viewport-adaptive 360-degree video streaming," Proc. 19th IEEE ISM, Taichung, Taiwan, pp.38–44, Dec. 2017.

[4] T.C. Thang, Q.-D. Ho, J.W. Kang, and A.T. Pham, "Adaptive streaming of audiovisual content using MPEG DASH," IEEE Trans. Consum. Electron., vol.58, no.1, pp.78–85, 2012.

[5] M. Graf, C. Timmerer, and C. Mueller, "Towards bandwidth efficient adaptive streaming of omnidirectional video over HTTP: Design, implementation, and evaluation," Proc. 8th ACM MMSys '17, Taipei, Taiwan, pp.261–271, June 2017.

[6] A. Zare, A. Aminlou, M.M. Hannuksela, and M. Gabbouj, "HEVC-compliant tile-based streaming of panoramic video for virtual reality applications," Proc. 24th ACM International Conference on Multimedia, MM '16, Amsterdam, Netherlands, pp.601–605, 2016.

[7] H. Ahmadi, O. Eltobgy, and M. Hefeeda, "Adaptive multicast streaming of virtual reality content to mobile users," Proc. ACM Multimedia 2017, Mountain View, California, USA, pp.170–178, ACM, 2017.

[8] Y. Ye, E. Alshina, and J. Boyce, "JVET-E1003: Algorithm descriptions of projection format conversion and video quality metrics in 360Lib," Joint Video Exploration Team ITU-T SG 16 WP3 ISO/IEC JTC 1/SC 29/WG 11 5th Meet., Geneva, Switzerland, Joint Video Exploration Team, 2017.

[9] H.T.T. Tran, C.T. Pham, N.P. Ngoc, A.T. Pham, and T.C. Thang, "A study on quality metrics for 360 video communications," IEICE Trans. Inf. & Syst., vol.E101-D, no.1, pp.28–36, Jan. 2018.

[10] X. Corbillon, F. De Simone, and G. Simon, "360-degreee video head movement dataset," Proc. 8th ACM MMsys, MMSys '17, Taipei, Taiwan, pp.199–204, ACM, 2017.

[11] D. Ochi, Y. Kunita, K. Fujii, A. Kojima, S. Iwaki, and J. Hirose, "HMD viewing spherical video streaming system," Proc. 22nd ACM Multimedia, Orlando, Florida, USA, pp.763–764, ACM, 2014.