FOV-BASED CODING OPTIMIZATION FOR 360-DEGREE VIRTUAL REALITY VIDEOS

Yuanyuan Xu *Taoyu Yang* *Zengjie Tan* *Haolun Lan*

Key Laboratory of Water Big Data Technology of Ministry of Water Resources, Hohai University
College of Computer and Information, Hohai University

ABSTRACT

Panoramic or 360-degree virtual reality videos have high resolution, frame rate, and visual quality that demand efficient coding. Although a user watching a 360-degree video can switch viewing angles, only a portion of the video in the user's Field of View (FoV) is displayed at any time. In this paper, we propose an FoV-based coding scheme for 360-degree videos, which allocates more bits to tiles of the predicted FoV area than other tiles. Taking possible FoV prediction error into account, the proposed scheme aims to minimize the expected weighted distortion of the FoV region, where different weights are given to tiles at different locations representing the influence of projection from spherical domain to the 2D plane. Accordingly, an adaptive tile-level quantization parameter (QP) selection scheme is derived. Simulation results demonstrate the effectiveness of the proposed scheme.

Index Terms— 360-degree video, video coding optimization, field of view (FoV), virtual reality

1. INTRODUCTION

360-degree virtual reality videos allow viewers to switch viewing angles providing an immersive viewing experience. Due to the omnidirectional content and magnifying optical lens in the associated head mounted display (HMD), 360-degree videos have higher visual quality, resolution, and frame rate than traditional videos, which pose challenges for coding and communication of such content. For example, a 360-degree video with a premium quality, 120 frames per second and 24k resolution needs a bandwidth in the range of Gigabits-per-second [1]. Therefore, an efficient coding scheme needs to be designed for 360-degree videos.

Different from traditional videos, 360-degree videos are in the spherical domain. Viewing a 360-degree video is as if sitting in the center of a sphere, where the user wearing HMD can yaw, pitch and roll head to see different content in his/her field of view (FoV). Due to the lack of spherical domain coding methods, spherical contents of 360-degree videos are generally projected to 2D planes and coded using traditional video codec. At the display side, the reconstructed rectangular images are mapped back to a sphere.

A few existing works on 360-degree video coding have considered the effect of projection. In [2], 360-degree video coding is optimized using a weighted-to-spherically uniform quality metric instead of uniformly measured distortion. A rate control method is proposed in [3] which efficiently allocates bits, considering the various pixels in different 2-D formats have different influence in the spherical domain. Coding optimization schemes [4, 5] are designed which consider the distortion in spherical domain. Observing that the entropy or bit-rate of a CTU is correlated with its geometry location, Zhou $et.al$ [6] proposed a new entropy equilibrium optimization method to enhance the coding performance of the 360-degree video.

For 360-degree videos, a user only watches a portion of the video scene with limited horizontal and vertical spans in his/her FoV at any given time. Therefore, FoV prediction information can be utilized to compress 360-degree video coding efficiently. In [7], a machine-learning based on saliency detection method is proposed, and an adaptive coding scheme is designed considering saliency and spatial activity. Coding tools of the high efficiency video coding (HEVC) standard [8] can be used to encode each image into multiple independent tiles. In [1],an FoV-adaptive coding scheme has been designed which only codes tiles for the predicted FoV, a surrounding border and a rotating intra region, whereas the true user's FoV may fall in un-coded tiles, whereas the true user's FoV may fall in un-coded tiles. The existing coding schemes utilizing FoV information rely on correct FoV prediction and do not fully consider the effect of projection. In [9, 10], accuracy of region-of-interest (RoI) prediction is considered in coding optimization of high-resolution videos with pan/tilt/zoom functionality, where slice sizes are optimized to minimize the transmission rate.

In this paper, we propose an FoV-based coding scheme for 360-degree videos, which takes into account both the influence of projection and FoV information. Considering the possible FoV prediction error, the coding optimization problem of 360-degree videos has been formulated aiming to minimize the expected weighted distortion of the FoV region, where different weights are given to tiles at different locations to rep-

The corresponding author is Yuanyuan Xu. This work is supported by National Natural Science Foundation of China under Grant No.61801167, and the Fundamental Research Funds for the Central Universities under Grant No.B200202189.

resent the influence of projection from the spherical domain to the 2D plane. Accordingly, an adaptive tile-level quantization parameter (QP) selection scheme is derived, which allocates more bits to tiles of the predicted FoV area than other tiles. Simulation results demonstrate the effectiveness of the proposed scheme.

2. CODING OPTIMIZATION PROBLEM FOR 360-DEGREE VIDEOS

For 360-degree videos, only distortion in the FoV region needs to be considered in the coding optimization. Although efficient saliency detection scheme [7] can provide information about where most users may be looking at, prediction results can deviate from the real FoV region. Taking possible FoV prediction error into account, the expected distortion in FoV region is considered during coding optimization. Furthermore, the influence of projection can be represened by assigning different weights to tiles at different locations. Therefore, coding optimization problem of 360-degree video can be formulated as minimize the expected weighted distortion in the FoV region, subject to a total bitrate constraint.

In this paper, we consider a tile-level video coding optimization. The equirectangular projection (ERP), one of most commonly used projection format, is employed to map video content in spherical domain to 2D plane. If each frame is encoded into $N \times M$ tiles, the size of each tile is $360°/N \times 180°/M$. The weighted FoV distortion for $tile(i,j)$, a tile located at i^{th} row ($i \in \{0,1,..,M-1\}$) and j^{th} column ($j \in \{0,1,..,N-1\}$), can be defined as follows.

$$D_{i,j}^{wf} = p_{i,j}^{FoV} w_i D_{i,j}, \tag{1}$$

where $p_{i,j}^{FoV}$, w_i, and $D_{i,j}$ are the probability of being in the FoV region, the weight related to projection, and distortion of $tile(i,j)$, respectively. The coding optimization problem can be formulated as minimizing the expected weighted distortion in the FoV region, subject to a total bitrate constraint as follows.

$$\begin{aligned} min \quad & \sum_{j=0}^{N-1}\sum_{i=0}^{M-1} D_{i,j}^{wf} = \sum_{j=0}^{N-1}\sum_{i=0}^{M-1} p_{i,j}^{FoV} w_i D_{i,j}, \\ subject\ to & \sum_{j=0}^{N-1}\sum_{i=0}^{M-1} R_{i,j} < R_c, \end{aligned} \tag{2}$$

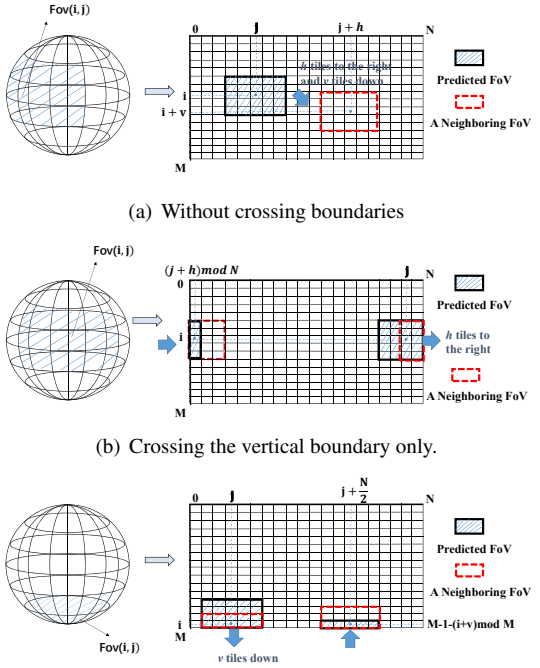
where $R_{i,j}$ and R_c are bitrate of tile located at i^{th} row and j^{th} column and total bitrate constraint, respectively.

2.1. The Expected FoV Distortion

In this paper, we consider the case of an FoV region with a fixed size occupying $n \times m$ tiles. The FoV region can be characterized by the center of viewport. To simplify this problem, we consider $N \times M$ FoV patterns, where the center of viewport in each FoV pattern aligns with the center of a particular tile. As in [11], we assume these FoV patterns follow two-dimensional Gaussian distribution, where the mean of Gaussian distribution is the center of the predicted FoV pattern. $FoV(i,j)$ is used to represent an FoV pattern whose center aligns with the center of $tile(i,j)$. If the predicted FoV is $FoV(\mu_i, \mu_j)$, the probability of another FoV pattern $FoV(i,j)$ can be expressed as follows.

$$p(FoV(i,j)) = \int_j^{j+1}\int_i^{i+1} \frac{1}{\sqrt{2\pi * \sigma^2}} e^{(-\frac{(\frac{x-\mu_i}{M}-0.5)^2+(\frac{y-\mu_j}{N}-0.5)^2}{2\sigma^2})} dxdy, \tag{3}$$

where $\sigma = 0.167$.



(a) Without crossing boundaries

(b) Crossing the vertical boundary only.

(c) Crossing the horizontal boundary

Fig. 1. Three cases of a neighboring FoV pattern for $FoV(i,j)$

Since there are $N \times M$ FoV patterns, each tile could be contained in multiple FoV patterns. To obtain a viewing probablity of $tile(i,j)$, all the neighboring $n \times m$ FoV patterns that contain $tile(i,j)$ need to be considered. Since movement in the spherical domain may cause movement accross image boundaries in ERP, a neighboring FoV pattern $FoV(i',j')$ which is v tiles down and h tiles to the right is discussed in following three cases.

1. *Without crossing boundaries.* In this case, the center of $FoV(i',j')$ is in the neighboring area of $tile(i,j)$, as shown in Fig. 1(a).

2. *Crossing the vertical boundary only.* If $tile(i,j)$ is near the left or right border of the 2D plane, the center of a

neighboring FoV pattern $FoV(i', j')$ may cross one of the vertical boundaries and appears on the other side of the 2D plane. An example of this case is illustrated in Fig. 1(b). Please note that the condition of crossing both vertical and horizontal boundaries will be discussed in the next case.

3. *Crossing the horizontal boundary.* If $tile(i, j)$ is located near the South or the North Pole, the center of a neighboring FoV pattern $FoV(i', j')$ may cross one of the horiontal boundaries in 2D plane and appears in the area that is 180 degrees apart from $tile(i, j)$ horizontally, as shown in Fig. 1(c).

The accumulative probability of $tile(i, j)$ being viewed can be represented as follows.

$$p_{i,j}^{FoV} = \sum_{h=-\frac{n}{2}}^{\frac{n}{2}} \sum_{v=-\frac{m}{2}}^{\frac{m}{2}} p(FoV(i', j')), \tag{4}$$

where

$$i' = \begin{cases} i+v, & 0 \le i+v < M \\ M-1-(i+v)mod \quad M, & i+v < 0 \; or \; i+v \ge M \end{cases}; \tag{5}$$

$$j' = \begin{cases} (j+h)mod\, N, & 0 \le i+v < M \\ j+\frac{N}{2}, & i+v < 0 \; or \; i+v \ge M \end{cases}. \tag{6}$$

2.2. The Projection Weight

In this work, we consider the ERP format. In ERP, rows near poles have higher sampling density on the sphere than rows near the equator [2]. Therefore, tiles at different locations in the 2D plane have different importance.

In [3], a pixel-level projection weight is defined as the area of a small unit on the sphere divided by the area of the corresponding area in the 2D plane. Thus, the weight for a pixel at the i^{th} row and j^{th} column is

$$w_i^{pixel} = cos\frac{(i+0.5-H/2)\pi}{H}, \tag{7}$$

where H is the number of pixels in each column. Based on the pixel-level weight, a tile-level projection weight can be defined as the average weight of all the pixels in the tile. The projection weight w_i for $tile(i, j)$ can be defined as:

$$w_i = \frac{M}{H} \sum_{k=0}^{\frac{H}{M}-1} cos(\frac{(k+0.5)\pi}{H} + i\pi/M - \pi/2) \tag{8}$$

3. THE WEIGHTED EXPECTED FOV-BASED CODING OPTIMIZATION

With the expected FoV distortion derived in Eq. (4) and projection weight in Eq. (8), the unconstrained problem in Eq. (2) can be solved using a Lagrange approach as follows.

$$\frac{\partial J}{\partial R_{i,j}} = \frac{\partial \sum_{j=0}^{N-1} \sum_{i=0}^{M-1} p_{i,j}^{FoV} w_i D_{i,j}}{\partial R_{i,j}} + \lambda = 0, \tag{9}$$

where λ is the Lagrange multiplier. Different tiles are independently encoded. Since the probability of a tile being in the FoV and its projection weight are independent of the coding procedure, using $\lambda_{i,j} = -\frac{\partial D_{i,j}}{\partial R_{i,j}}$ Eq. (9) is equivalent to

$$-p_{i,j}^{FoV} w_i \lambda_{i,j} + \lambda = 0 \tag{10}$$

According to [12], the relationship between quantization parameter (QP) and the Lagrange multiplier λ can be emprically obtained as

$$QP = 4.2005ln(\lambda) + 13.7122. \tag{11}$$

Based on the relationship between tile-level Lagrange multipliers and a global Lagrange multiplier in Eq. (10), we propose a coding scheme that adapts the QP of each tile considering the possibility of being in the FoV region and projection weight. With a base QP denoted as QP_{base} corresponding to the global Lagrange multiplier, the QP value of each tile can be calculated as

$$QP_{i,j} = QP_{base} - 4.2005ln(p_{i,j}^{FoV} w_i). \tag{12}$$

4. EXPERIMENTAL RESULT

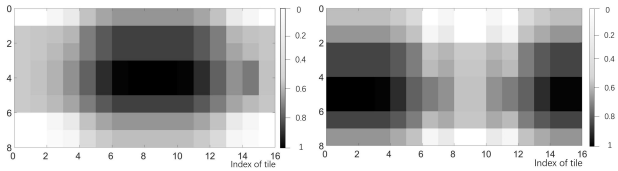
4.1. Simulation Setup

The proposed scheme is implemented in the HEVC reference software HM 16.16. Ten 360-degree video sequences in ERP format are selected from dataset [13]. 15 frames of these video sequences are encoded using QP values of {22, 27, 32, 37} under *Random Access* and *All Intra* configurations. Each frame is encoded into 16×8 tiles. The FoV region has a fixed size of $115° \times 115°$, which consists of 5×5 tiles. The first viewport of each sequence provided in JVET configuration file or dataset [13] is selected as the predicted FoV region.

During coding, QP for each tile is calculated according to Eq. (12). At the decoder side, viewports of 50 users are generated according to two-dimensional Gaussian model with the predicted viewport as the mean value. Only tiles associated with each user's viewport are decoded and displayed. The weighted-spherical PSNR (WS-PSNR) [14] of the FoV region is used as a quality metric. Bjontegaard delta-rate (BD-rate) [15] is used to evaluate the coding performance of the proposed scheme compared with the HM anchor.

4.2. Results and Discussion

Fig. 2 illustrates the tile-level coefficient $p_{i,j}^{FoV} w_i$ in Eq. (1) representing the importance of different tiles. The value of this coefficient decreases as the distance between tile and



(a) Predicted pattern: $FoV(4,9)$ (b) Predicted pattern: $FoV(5,1)$

Fig. 2. Illustration of Tile-level Coefficient $p_{i,j}^{FoV} w_i$.

viewport center gets larger. Fig. 2(a) shows a regular case, while Fig. 2(b) illustrates the value distribution when the center of predicted FoV is near the left border of the image.

Table 1. The BD-rate saving of proposed work and the work in [5] compared with HM anchor

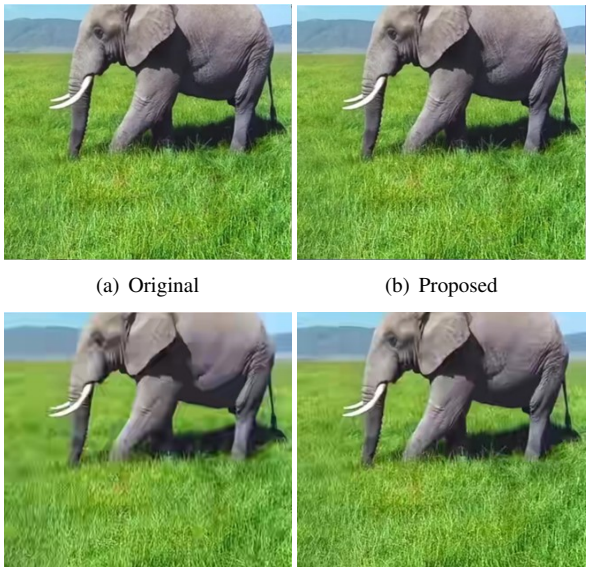
Resolution	Sequence	Proposed		Li *et al.* [3]	
		Random Access	All Intra	Random Access	All Intra
3840 × 1920	Angelfalls	-3.45%	-4.41%	-0.1%	-0.69%
	Broadway	-2.83%	-3.41%	0.18%	-0.1%
	Canolafield	-5.03%	-5.76%	-0.11%	-0.37%
	Harbor	-3.16%	-3.22%	-0.1%	-0.24%
	Newyork	-3.24%	-3.49%	0.12%	-0.41%
	Trolley	-3.15%	-3.32%	-0.23%	-0.31%
	Gaslamp	-6.12%	-6.35%	-0.18%	-0.34%
	Kiteflite	-3.08%	-3.21%	-0.09%	-0.22%
3840 × 2048	Elephants	-5.26%	-5.74%	-0.25%	-0.41%
	Diving	-3.13%	-3.27%	0.04%	-0.18%
Average		-3.84%	-4.2%	-0.07%	-0.32%

(a) Original (b) Proposed

(c) HM 16.16 (d) Li *et al.* [3]

Fig. 3. The subjective visual quality comparison showing a cropped FoV area in the 4^{th} frame of "Angelfalls": (a) original; (b) proposed (bitrate: 14,637 kbps); (c) HM 16.16 (bitrate: 40,514 kbps); (d) Li *et al.* [3] (bitrate: 31,738 kbps).

To verify the effectiveness of the proposed scheme, it is compared with HM 16.16 and the coding scheme in [3]. HM 16.16 codes each tile with a fixed QP, while the scheme in [3] adjusts QP only considering projection effect. To achieve a fair comparison, a tile-level QP adjustment scheme is implemented for work [3]. The BD-rate savings of the two schemes



(a) Original (b) Proposed

(c) HM 16.16 (d) Li *et al.* [3]

Fig. 4. The subjective visual quality comparison showing a cropped FoV area in the 4^{th} frame of "Elephants": (a) original; (b) proposed (bitrate: 55,658 kbps); (c) HM 16.16 (bitrate: 122,527 kbps); (d) Li *et al.* [3] (bitrate:115,594 kbps).

are presented in Table 1. The proposed scheme can achieve up to 6.12% and 6.35%, and an average of 3.84% and 4.2% BD-rate savings compared with the HM 16.16 anchor under *Random Access* and *All Intra* configurations, respectively. Since the scheme in [3] only considers the influence of projection, tiles at the same row have the same priority, achieving an average of 0.07% and 0.32% BD-rate saving compared with HM 16.16 anchor under *Random Access* and *All Intra* configurations, respectively. The perforamnce gain under *All Intra* configuration is larger than that in *Random Access* configuration.

The subjective visual quality comparisons of the FoV area under the configuration of *All Intra* are presented in Fig. 3 and Fig. 4. We crop a region with the size of 440 $\times$376 from the sequence of "Angelfalls" and "Elephants", respectively. Though with a lower bitrate, the proposed scheme can achieve better visual quality in the FoV area than HM 16.16 and the work in [3].

5. CONCLUSION

In this paper, a tile-level FoV-based coding optimization scheme has been proposed for 360-degree videos. The optimization problem has been formulated aiming to minimize the expected weighted distortion of the FoV region, considering projection weights and possible FoV prediction error. Accordingly, a derived tile-level QP adjustment scheme efficiently allocates bits according to tiles' priority. Simulation results demonstrate the effectiveness of the proposed scheme.

6. REFERENCES

[1] Y. Liu, L. Sun, Y. Mao, and Y. Wang, "Low-Latency FoV-Adaptive Coding and Streaming for Interactive 360° Video Streaming," in *Proceedings of the 28th ACM International Conference on Multimedia(ACM MM)*, 2020, pp. 3696–3704.

[2] Y. Sun and L. Yu, "Coding Optimization Based on Weighted-to-Spherically-Uniform Quality Metric for 360 Video," in *Proceedings of 2017 IEEE Visual Communications and Image Processing (VCIP)*, 2017, pp. 1–4.

[3] L. Li, N. Yan, Z. Li, S. Liu, and H. Li, "λ-domain perceptual rate control for 360-degree video compression," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, pp. 130–145, 2019.

[4] Y. Li, J. Xu, and Z. Chen, "Spherical Domain Rate-Distortion Optimization for 360-Degree Video Coding," in *Proceedings of 2017 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2017, pp. 709–714.

[5] Y. Liu, H. Guo, and C. Zhu, "Spherical Position Dependent Rate-Distortion Optimization for 360-Degree Video Coding," in *Proceedings of 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2019, pp. 992–996.

[6] Y. Zhou, L. Tian, C. Zhu, X. Jin, and Y. Sun, "Video Coding Optimization for Virtual Reality 360-Degree Source," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, pp. 118–129, 2020.

[7] G. Luz, J. Ascenso, C. Brites, and F. Pereira, "Saliency-Driven Omnidirectional Imaging Adaptive Coding: Modeling and Assessment," in *Proceedings of 2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2017, pp. 1–6.

[8] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of The High Efficiency Video Coding (HEVC) Standard," vol. 22, pp. 1649–1668, 2012.

[9] A. Mavlankar and B. Girod, "Video Streaming with Interactive Pan/Tilt/Zoom," in *High-Quality Visual Experience. Signals and Communication Technology*, chapter 19, pp. 431–455. 2010.

[10] A. Mavlankar and B. Girod, "Spatial-random-access-enabled video coding for interactive virtual pan/tilt/zoom functionality," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 5, pp. 577–588, 2011.

[11] H. Yuan, S. Zhao, J. Hou, X. Wei, and S. Kwong, "Spatial and Temporal Consistency-Aware Dynamic Adaptive Streaming for 360-Degree Videos," vol. 14, pp. 177–193, 2020.

[12] D. Zhang, J. Xu, B. Li, and H. Li, "QP Refinement According to Lagrange Multiplier for High Efficiency Video Coding," in *Proceedings of 2013 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2013, pp. 477–480.

[13] "Viewport-adaptation-induced immersive video quality assessment database," https://vision.nju.edu.cn/11/e6/c29466a463334/page.htm.

[14] Z. Chen, Y. Li, and Y. Zhang, "Recent Advances in Omnidirectional Video Coding for Virtual Reality: Projection and Evaluation," *Signal Processing*, vol. 146, pp. 66–78, 2018.

[15] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," *ITU-T VCEG-M33, April*, 2001.