

A Novel Two-Level Rate Control Algorithm Based on Visual Attention for 360-Degree Video for Versatile Video Coding Standard

Hoda Roodaki (✉ hroodaki@kntu.ac.ir)

Department of Computer Engineering, K. N. Toosi University of Technology,

Research

Keywords: 360-degree video, Versatile Video Coding (VVC) standard, high-order motion models, motion complexity, λ -domain rate control

Posted Date: November 12th, 2020

DOI: <https://doi.org/10.21203/rs.3.rs-102968/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

RESEARCH

A Novel Two-level Rate Control Algorithm Based on Visual Attention for 360-degree Video for Versatile Video Coding standard

Hoda Roodaki

Correspondence:
hroodaki@kntu.ac.ir
Department of Computer
Engineering, K. N. Toosi
University of Technology, Tehran,
Iran
Full list of author information is
available at the end of the article

Abstract

360-degree video is providing users with an interactive experience to explore the scenes freely. But, because only a small portion of the entire video, called viewport, is watched at every point in time, transmitting the entire video is bandwidth-consuming. Since, the perceptual quality of such video mainly depends on the quality of the viewport, more bandwidth should be assigned to these important parts of the scene. Hence, understanding how people observe and explore 360-degree content is essential. In this paper, we propose a new Two-level rate control algorithm which tries to allocate more bits for encoding the viewport parts of a 360-degree video. The head and eye movements of the observers is used to investigate the visual attention of people to detect the viewports. Then, a Coding Tree Unit (CTU) level rate assignment approach is proposed to assign a proper number of bits to each CTU of the viewport and non-viewport parts. It is assumed that higher motion complexity results in higher bitrates of the encoded video. So, we propose to assign the proper number of bits to each CTU according to its motion complexity. Another novel part of our proposed approach is proposing a new metric to parameterize the motion complexity of each CTU using the high-order motion models in Versatile Video Coding (VVC) standard. Experimental results show that our proposed rate control, on average, achieves 58.27% reduction in bitrate in the Bjøntegaard-Bitrate scales, compared to the standard VCC standard. Furthermore, the proposed scheme provides a significantly better subjective viewing quality compared to the-state-of-the-art methods.

Keywords: 360-degree video; Versatile Video Coding (VVC) standard; high-order motion models; motion complexity; λ -domain rate control

1 Introduction

360-degree video offers a full panoramic field of view and lets observers explore scenes from any direction they choose. During playback, viewers can control the viewing direction by naturally moving their head towards the direction of the view. This use case is becoming more and more popular in platforms such as YouTube 360° video channels. Viewers can look at arbitrary directions within the scene without any limitation confining traditional media. Hence, 360-degree videos are extremely bandwidth-intensive when offered at high quality and are therefore difficult to stream. Even with new compression techniques, this application is currently beyond the capabilities of most receivers. However, in most 360-degree video applications, viewers only watch a small window of the entire scene called the Field

of View (FoV) or viewport [1] at every point in time. Therefore, more bandwidth should be dedicated to these important parts of the scene. After all, the perceptual quality of such video mainly depends on the quality of the viewport. Transferring data comprising scenery not viewed by viewers result in overall low-quality video streaming by dedicating bandwidth to unnecessary data. With this in mind, understanding how people observe and explore 360-degree content is essential. In this paper, we propose a new Coding Tree Unit (CTU)-level rate control scheme based on visual attention for 360-degree content using the Versatile Video Coding (VVC) standard. Versatile Video Coding (VVC) is a video compression standard being developed by the Joint Video Experts Team (JVET) and planned to finalize in mid-2020, and VVC Test Model (VTM) is the reference implementation of VVC specification draft. Our proposed method tries to allocate more bits for encoding the viewports in a 360-degree video that are noteworthy for viewers. For this purpose, a CTU-level rate control is proposed. In the proposed method, the motion complexity of a CTU is considered as a metric to assign a proper number of bits to it. So, we propose a new metric to characterize the motion complexity of the CTU. This metric uses the high-order motion models in VVC standard as explained in the following sections. In addition, among the various existing R- λ model, we use the λ -domain rate control algorithm in this paper as explained in the next section. The main contributions of this paper are as follows.

- Analyzing the head and eye movements of observers to investigate the behaviors of people watching the 360-degree video to extract the viewport and non-viewport parts of the video
- Proposing a Two-level rate control algorithm (viewport-adaptive and CTU-level) for 360-degree video,
- Proposing a new metric to characterize the motion complexity of each CTU to assign a proper number of bits to CTUs

The rest of this paper is organized as follows. In section 2, the related works are reviewed. Section 3 analyzes the λ -domain rate control algorithm. The proposed methods to assign the bits to the viewport and non-viewport parts and the corresponding CTUs are presented in section 4. Section 5 provides the performance evaluation results. Finally, section 6 concludes the proposal.

2 Related Work

Several methods of bitrate allocation for 360-degree videos have been introduced in the literature. In [2], a Quality of Experience (QoE) driven viewport adaptation system is proposed for 360-degree video. This method extracts the user's head movement probability and creates a probabilistic model to show the distribution of viewport prediction error. Then, a QoE-driven optimization framework is used to minimize the estimated distortion. In [3] a framework for viewport-driven rate allocation for 360-degree video is proposed. This method tries to investigate the user view navigation pattern and the spatio-temporal rate-distortion characteristics of a 360-degree video content to maximize the user's perceived quality. The method proposed in [4] tries to predict the viewport using the user's historical head movement. Then, a Quality of Experience rate adaptation method is used to maximize the efficiency under the constraint of the total bitrate. In [5] another rate adaptation

method is proposed that tries to maximize the QoE by predicting the user's viewport with a probabilistic model. In [6] an end-to-end tiled based video streaming framework for 360-degree videos is developed. It uses deep learning to predict the user viewports. Then, a rate adaptation method is used to allocate the proper rate to the tiles according to their viewing probabilities. In [7] a convolutional neural network is used to predict the relationship between the future and historical viewports. Then, an optimization problem is solved to maximize the perceived quality of all users and the quality difference between the viewport and non-viewport parts for each user. The method proposed in [8] uses a Gaussian model to predict the viewport and an optimization algorithm is suggested to preserve the perceived quality of the viewport tiles. As explained, none of the previous approaches use the visual attention of the observers to find the viewport and non-viewport parts. Our proposed approach uses this metric to allocate bitrate to these parts. In addition, the motion complexity of each CTU is investigated as a metric for bit allocation which has not been considered before.

3 λ -domain rate control algorithm

The proposed approach is based on λ -domain rate control for the VVC standard [9] [10]. In λ -domain rate control, the λ parameter is defined as the slope of the R-D curve as shown in equation (1) [10]:

$$\lambda = -\frac{\partial D}{\partial R} = \alpha R^\beta \quad (1)$$

Hence,

$$R = \left(\frac{\lambda}{\alpha}\right)^{\frac{1}{\beta}} \quad (2)$$

where α and β are parameters related to the video content and can be extracted experimentally. Equation (2) shows that the λ parameter determines the bitrate R and the bitrate value can be defined in the λ domain. Hence, λ can be determined according to the target R using the equation (2). In order to achieve the target bitrate for a specific unit such as a frame or a CTU using this model, it is essential to determine the other coding parameters, including Quantization Parameter (QP). When the λ value is set using equations (1) and (2), the QP can be determined using the following equation as suggested in [11].

$$QP = c1 \times \ln(\lambda) + c2 \quad (3)$$

where $c1$ and $c2$ are set to 4.2005 and 13.7122, respectively [12]. The extracted QP value from (3) should be rounded to the nearest integer [12]

4 Proposed Method

Our proposed approach has two main steps to assign a proper rate to a 360-degree video. The main scheme of the proposed method is shown in figure 1. First, the viewport map is generated using the visual attention model of viewers. Then, the proper number of bits are assigned to the viewport and non-viewport parts according to the ratio between the sizes of these two parts. Then a CTU-level rate control is used to assign the suitable number of bits to each CTU of the viewport and non-viewport parts.

Figure 1 The main scheme of the proposed approach

4.1 Viewport extraction

In our proposed method, the extracted viewport map is given as the input of the rate control algorithm. We try to explore the visual attention of people in a 360-degree content by analyzing the head and eye movements to investigate the behaviors of people watching the 360-degree video. So, the results of some experiments that analyze the visual attention of people in a 360-degree video is investigated [13]. This dataset was obtained by processing the eye and head movements of 57 observers wearing a VR headset that is equipped with an eye-tracker. Then, two values are reported for each participant in the experiment that show the gaze positions in longitudes and latitudes, normalized between 0 and 1. Latitude and Longitude are the geographic coordinates that specifies the north–south and east–west position of a point on the sphere. The videos of the dataset are projected to a 2D plane using the Equirectangular Projection (ERP) format. This format is the most widely used projection format for representing a 360-degree video on a 2D plane. The values of longitudes and latitudes should be multiplied by 2π and π respectively to obtain positions on the sphere (u, v) [13]. Then, the m and n points in the 2D plane can be extracted using equations (4) and (5) [14].

$$u = (m + 0.5)/W, \quad 0 \leq m < W \quad (4)$$

$$v = (n + 0.5)/H, \quad 0 \leq n < H \quad (5)$$

where W and H show the width and height of the 2D frames, respectively. A scene from one of the videos of this dataset is shown in figure 2. Table 1 shows some samples of its corresponding gaze points and the converted 2D-points. We have done the above calculations for the presented gaze points for all 57 observers in the dataset and extracted the viewport parts of the whole video on average.

Figure 2 A scene from the test sequence “PortoRiverside” from the dataset presented in [13]

Of course, the proposed bit allocation scheme in this paper is entirely independent from the method that is used to extract the viewport parts. However, using this method to extract the viewports will lead to a better subjective quality.

Table 1 Samples of gaze points and the corresponding converted 2D-points for “PortoRiverside” sequence [13]

Observer	longitude	latitude	Converted 2D point (X)	Converted 2D point (Y)
0	0.447	0.543	2192.77	627.47
1	0.446	0.543	2192.37	627.29
2	0.440	0.544	2188.24	627.00

4.2 Bit allocation Algorithm

After viewport selection, the viewport-adaptive bit allocation for the viewport and non-viewport parts should be done. Some methods are presented in the literature for region-based bit allocation [15]. But in order to assign the bits more efficiently, the main features of video content should be considered. As explained before, in this paper, the VVC standard is used to encode the video contents. The VVC standard adopts the block-based hybrid video coding structure and utilizes a QuadTree with a nested Multi-Type tree coding (QTMT) block structure [16]. Figure 3 shows the block partitioning in the VVC standard. QTMT allows a binary or ternary splitting at all the leaf nodes of a quad-tree. First, the video is divided into some basic processing units named Coding Tree Units (CTU). Then, it is divided into a quad-tree and multi-type tree structure to further partition the quad-tree leaf nodes.

Figure 3 Illustration of a QuadTree with nested Multi-Type tree coding (QTMT) block structure [17]

Our proposed approach for bit allocation has two parts, viewport-adaptive and CTU-based rate allocation. In the viewport-adaptive part, the total number of bits of the viewport and non-viewport parts is allocated. Then, at the next step, the rate of each CTU is assigned.

4.3 Viewport-adaptive rate allocation

In this step, first, the ratio between the rate of the viewport and non-viewport part will be extracted. For this purpose, the number of CTUs in the viewport and non-viewport parts are counted. The ratio between the rate of the viewport and non-viewport parts can be extracted by dividing the number of CTUs in the viewport (VP) by the number of CTUs in the non-viewport (non-VP) parts.

$$K = \frac{\#CTUs\ in\ VP}{\#CTUs\ in\ non-VP} \quad (6)$$

The total bitrate B_{total} will be the sum of the bitrate of the viewport and non-viewport parts:

$$B_{total} = B_{VP} + B_{non-VP} \quad (7)$$

where B_{VP} and B_{non-VP} is the total number of bits of the viewport and non-viewport parts, respectively. Using (6), we have:

$$B_{total} = (K + 1) \times B_{non-VP} \quad (8)$$

4.4 CTU-level rate allocation

For the non-viewport parts, the total bitrate would be calculated by multiplying the number of CTUs of non-viewport parts by the bitrate of each CTU.

$$B_{non-VP} = B_{CTU_{non-VP}} \times \#CTUs \text{ in } (non-VP) \quad (9)$$

where $B_{CTU_{non-VP}}$ is the bitrate of each CTU in the non-viewport part. From equations (8) and (9), we have:

$$B_{CTU_{non-VP}} = \frac{B_{total}}{(K + 1)} \times \#CTUs \text{ in } (non-VP) \quad (10)$$

But, the bitrate of CTUs in the viewport parts should be assigned more accurately. Since the quality of these parts affect the perceived quality of the viewers significantly. According to the studies in the related literature, the “motion complexity” of the video content is one of the most important features that affected the total bitrate of a video [18]. It is assumed that higher motion complexity results in higher bitrates of the encoded video.

Accordingly, in this work, the motion complexity of each CTU in viewport parts is considered as a metric to assign a proper number of bits to the CTUs of this part. For this purpose, first, the CTU motion complexity concept should be parameterized by defining the proper parameters that describe it. We start by explaining the basic observations to parametrize the CTU motion complexity.

4.4.1 Observations

In the video coding process, each frame is split into blocks and then the translation model is used to describe the changes between them. For instance, if $Blk(t + 1)$ shows the coordinates of the top-left corner of a block at frame $(t + 1)$ and $Blk(t)$ shows the coordinates of the top-left corner of the corresponding one in frame (t) as the predicted and reference frames respectively, dx and dy are the translation of the center of the block described as:

$$Blk(t + 1) = Blk(t) + \begin{bmatrix} dx \\ dy \end{bmatrix} \quad (11)$$

The translation models work well when the motion between frames is rather small and the cameras and objects do not rotate or shear. The high-order motion models, such as affine, bilinear, and perspective motion models could characterize more complex motions such as rotation and zooming in natural videos more efficiently [19]. Among these models, affine motion is much simpler and can better characterize complex motions and improve coding efficiency [20]. The affine motion model

includes transformations such as rotation and shear. This model is described as follows [20]:

$$Blk(t+1) = \begin{bmatrix} r_1 & r_2 \\ r_3 & r_4 \end{bmatrix} \times Blk(t) + \begin{bmatrix} dx \\ dy \end{bmatrix} \quad (12)$$

where $\begin{bmatrix} dx \\ dy \end{bmatrix}$ shows the formation matrix for rotation and shear. In the VVC standard, during the rate-distortion optimization process at the encoder side, the rate of each block is calculated using the translational and affine models separately. Then, the model with the lower rate is used during the encoding process as shown in Figure 4.

Figure 4 Motion model predictor in VVC standard

In theory, the selection of the affine model during the rate-distortion optimization process shows that the complex motions of CTUs are not well described in conventional models. Hence, we propose that the percentage of affine motion model selection for the blocks of each CTU would be considered as a metric to show the motion complexity of that CTU. In order to justify this hypothesis, we compared this proposed metric with one of the most known ones currently in use. In [18], it is suggested to use the following equation to find the level of motion and motion complexity.

$$LevelofMotion = \frac{Bitrate_P + Bitrate_B}{2 \times 10^6 \times 0.89(QP_P + QP_B)} \quad (13)$$

where $Bitrate_P$ and $Bitrate_B$ are the numbers of bits that are used for I , P and B frames and QP_P and QP_B are the average quantization parameters of P and B frames respectively.

For a natural video, the bitrate of the coded P-frame depends on the motion across the video frames in the scene. Since for a P-frame a higher motion in the video scene leads to a larger difference in the pixel values across the frames, more bits are required. Indeed, the most critical parameter affecting the rate and distortion trade-off in coding a video is quantization parameter. So, these parameters affect the motion complexity of a video.

In order to show the efficiency of our proposed metric, we extract the motion complexity of 6 test sequences shown in Table 2, using our proposed metric and the metric proposed in [18]. This dataset is comprised of 4K video sequences in 3840×1920 pixels resolution and equirectangular format.

Then, the correlation coefficient between these two metrics are extracted. The correlation coefficient is a measure of the linear dependence between two variables. The correlation coefficient is calculated between these two values using the following equation.

Table 2 Properties of the test sequences

Sequence	Original Resolution
PortoRiverside	3840 × 1920
Sofa	3840 × 1920
Cows	3840 × 1920
Fountain	3840 × 1920
UnderwaterPark	3840 × 1920
Touvet	3840 × 1920

$$R = \frac{\sum_{i=1}^N (X_i - \bar{X}) \times (Y_i - \bar{Y})}{\sum_{i=1}^N (X_i - \bar{X})^2 \times \sum_{i=1}^N (Y_i - \bar{Y})^2} \quad (14)$$

where X_i and Y_i denote the two sample scores. \bar{X} and \bar{Y} show the samples means and N represents the total number of tests considered in the evaluation process. The value of motion complexity extracted from [18] and our proposed metric and the correlation coefficient between them are shown in Table 3. As we can see the correlation coefficient between these two values is observed to be high at 75%. On the other hand, as stated in [13], we have large camera movements in “UnderwaterPark” and “Touvet” video sequences and high-order motion models, such as affine, could characterize such complex camera motions including rotation and panning more efficiently. The results depicted in Table 3 show that the motion complexity of these two sequences calculated by our proposed method is higher compared to the other sequences. This shows that our proposed metric performs far better in obtaining true motion complexity by defining it as the percentage of affine model selection.

Table 3 The value obtained for level of motion from the method described in [18] and the proposed method and the corresponding correlation coefficient

Sequences	Level of Motion extracted from our approach	Level of Motion extracted from the method proposed in [18]
Sofa	0.52	0.23
Cows	0.60	0.83
UnderwaterPark	0.65	3.58
Touvet	0.66	1.57
Correlation Coefficient	75%	

According to this observation, we consider the “percentage of the affine model selection for each CTU” as a metric for measuring the motion complexity of the CTU in this work.

4.4.2 CTU-based rate allocation in viewport section

As explained before, our proposed rate control at CTU level tries to assign a greater number of bits to CTUs with higher motion complexities. The allocated bits to each CTU are obtained using the following equation.

$$B_{CTU_{VP}} = \frac{B_{VP} - B'_{VP}}{\sum W_{CTU} - VP} \times W_{CTU_{VP}} \quad (15)$$

where B_{VP} is the viewport bitrate extracted during the previous step and the B'_{VP} is the assigned bitrate to encoded CTUs in this viewport. So, $B_{VP} - B'_{VP}$ shows the available number of bits for the current CTU. $w_{CTU_{VP}}$ is the weight of each CTU in the bit allocation process. In our proposed approach, the motion complexity (calculating by the proposed metric) is used as a weight value in equation (15). In this case, CTUs with higher motion complexity achieve higher bitrate compared to CTUs with lower motion complexity.

5 Simulation Results

Several experiments were performed to show the efficiency of our proposed rate control for 360-degree video sequences. The results show the effectiveness of the proposed approach to allocate proper rates to the viewport and non-viewport parts of 360-degree videos. Results have been obtained using the VVC standard reference software, *VTM – 4.0* [21]. As stated in the previous section, in earlier video coding standards only the translation motion model is used for motion compensation prediction. However, many types of complex motions such as rotation can typically be observed in video sequences. In the VVC standard and its corresponding reference software, a simplified 4-parameter affine transform motion compensation prediction is applied. We have used this coding tool in VTM in order to parameterize the motion complexity and extract the weight of the CTUs bitrate.

5.1 Extraction of the viewport and non-viewport parts

As explained before, six video sequences of the dataset proposed in [13] are used for our experiments. The dataset contains the associated gaze fixation and head trajectory data. This data is obtained by processing the eye and head movements of 57 observers gathered from a free-viewing experiment using a VR headset equipped with an eye-tracker. In this dataset, for each observer, the gaze positions in longitudes and latitudes are reported. We have used equations (4) and (5) to extract the corresponding points transferred to a 2D plane. Then, we have divided the extracted values by 128 to find the related CTUs to the viewport and non-viewport parts. Hence, the K value in equation (6) can be derived at this point. Then, the bitrate for each CTU in the non-viewport part can be extracted from equation (10). In order to extract the bitrate of each CTU in the viewport region, equation (15) is used. The weight of bitrate for each CTU in this equation is calculated using the concept of “motion complexity” defined as the percentage of the affine model selection for CTUs of the viewport part. Using the value of “motion complexity” as the weight value for CTUs in equation (15), the bitrate of each CTU of the viewport part is calculated. Table 4 shows the number of bits for each CTU of the Frame 1 of two video sequences, for $QP = 22$, obtained from equation (15).

By calculating the total number of bits for viewports, the remaining bits could be assigned to non-viewport parts. Once the number of allocated bits for each CTU in the viewport and non-viewport sections is initialized, the corresponding QP is computed using λ -domain rate control as explained in equations (1), (2) and (3). In order to show the efficiency of our proposed method, we have compared the performance of our proposed method against the method that uses the λ -domain rate control simply to find the QP values for viewport and non-viewport parts.

Table 4 The number of extracted bits for each CTU of the viewport part of the Frame 1 using equation (15)

Sequences	CTU Number	Number of bits (bit)
PortoRiverside	1	3659807.31
	2	942677.64
	3	314225.88
	4	157112.94
	5	94267.76
	6	62845.17
	7	44889.41
	8	33667.05
	9	26185.49

Table 5 shows the extracted QP values for the viewport and non-viewport parts in our proposed method and the anchor method for PortoRiverside test sequence at four various target bitrates.

Table 5 The extracted QP values for viewport and non-viewport parts in our proposed method and the anchor method

Sequences	Proposed Method		Anchor Method	
	viewport	Non-viewport	viewport	Non-viewport
PortoRiverside	19	22	22	22
	24	27	27	27
	29	32	32	32
	34	37	37	37

Then, the viewport and non-viewport parts of these video sequences have been coded using the extracted QP values for the proposed method and the anchor method. This coding performance comparison is illustrated in Table 6. As shown in the results, the bitrate is reduced significantly using our proposed method.

Table 6 The coding performance analysis of the proposed approach against the anchor method

Sequence	BD-Bitrate
PortoRiverside	-66.02%
Sofa	-56.66%
Cows	-59.33%
Fountain	-45.52%
UnderwaterPark	-58.27%
Touvet	-63.81%
Average	-58.27%

It should be noted that in the extraction of the results shown above, we tried to track the required target bitrate accurately. It means that, we try to increase the bitrate of viewport parts and decrease the bitrate of non-viewport parts, in a way that, the total bitrate would not be increased considerably. Figure 5 shows the comparison of the required target bitrates and the total bitrates of the viewport and non-viewport parts using our proposed approach for Cows video sequence. As we can see, the obtained bitrate using our method follows the required target bitrate precisely.

Figure 5 Comparison of the target extracted bitrate using our proposed method and the required target bitrate for Cows video sequence

Finally, we have compared the results of the viewport parts of our proposed method and the anchor method subjectively for more clarifications. Figure 6 show

the results. As we can see, our method achieves the highest subjective quality in the mentioned parts. In the anchor method the details of the picture is completely blurred.

Figure 6 Comparisons of subjective quality of (a) anchor method and (b) our proposed method for Fountain sequence

Moreover, we compared the results with one of the similar methods [7] in which, the rate allocation problem is formulated as an optimization problem that tries to maximize the received quality of the video. Then a steepest descent algorithm is presented to solve the optimization problem. We have compared the results of our proposed method and this method as shown in Figure 7. The subjective results of our proposed method is much better than the anchor one in viewport and non-viewport parts.

Figure 7 Comparison of (a) the results of the method proposed in [7] and (b) the result of our proposed method

6 Conclusion

This paper proposes a two-level rate control method to assign a proper number of bits to a 360-degree video using VVC standard. First, the proper number of bits is assigned to viewport and non-viewport parts according to their number of CTUs. The, the motion complexity of the CTUs is takes into account to allocate a proper number of bits to each CTUs of viewport parts. So, a new metric is proposed to parameterize the motion complexity of CTUs which is the percentage of the affine model selections in each CTU. We have extracted this value using the coding tools of VTM, the reference software of the VVC standard. Experimental results show that our proposed method can achieve 58.27% reduction in bitrate in the Bjøntegaard-Bitrate scale compared to the standard Versatile Video Coding(VVC) standard with a better subjective viewing quality compared to the-state-of-the-art methods.

Author details

Department of Computer Engineering, K. N. Toosi University of Technology, Tehran, Iran.

References

1. Corbillon, X., Simon, G., Devlic, A., Chakareski, J.: Viewport-adaptive navigable 360-degree video delivery. *IEEE International Conference on Communications (ICC)* (2017)
2. Xie, L., Xu, Z., Ban, Y., Zhang, X., Guo, Z.: 360probdash: Improving qoe of 360 video streaming using tile-based http adaptive streaming. *Proceedings of the 25th ACM international conference on Multimedia* (2017)
3. Chakareski, J., Aksu, R., Corbillon, X., Simon, G., Swaminathan, V.: Viewport-driven rate-distortion optimized 360° video streaming. *IEEE International Conference on Communications* (2018)
4. Hu, Y., Liu, Y., Wang, Y.: Qoe-driven viewport adaptive streaming for 360 video. *IEEE International Conference on Multimedia & Expo Workshops (ICMEW)* (2019)
5. Hu, H., Xu, Z., Zhang, X., Guo, Z.: Optimal viewport-adaptive 360-degree video streaming against random head movement. *IEEE International Conference on Communications (ICC)* (2019)
6. Park, S., Bhattacharya, A., Yang, A., Dasari, M., Das, S.R., Samaras, D.: Advancing user quality of experience in 360-degree video streaming. *IFIP Networking Conference (IFIP Networking)* (2019)
7. Zou, J., Li, C., Liu, C., Yang, Q., Xiong, H., Steinbach, E.: Probabilistic tile visibility-based server-side rate adaptation for adaptive 360-degree video streaming. *IEEE Journal of Selected Topics in Signal Processing* **14**, 161–176 (2020)
8. Yuan, H., Zhao, S., Hou, J., Wei, X., Kwong, S.: Spatial and temporal consistency-aware dynamic adaptive streaming for 360-degree videos. *IEEE Journal of Selected Topics in Signal Processing* **14**, 177–193 (2020)

9. Li, Y., Chen, Z.: Rate control for vvc. Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 11th Meeting (2018)
10. Liu, Z., Li, Y.: Ahg10: Adaptive lambda ratio estimation for rate control in vvc. Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 12th Meeting (2018)
11. Li, B., Li, H., Li, L., Zhang, J.: Lambda domain rate control algorithm for high efficiency video coding. *IEEE Transactions on Image Processing* **23**, 3841–3854 (2014)
12. Wen, J., Fang, M., Tang, M., Wu, K.: R-(lambda) model based improved rate control for hevc with pre-encoding. *Data Compression Conference (DCC)* (2015)
13. David, E.J., Gutiérrez, J., Coutrot, A., Silva, M.P.D., Callet, P.L.: A dataset of head and eye movements for 360° videos. *Proceedings of the 9th ACM Multimedia Systems Conference* (2018)
14. Ye, Y., Alshina, E., Boyce, J.: Algorithm descriptions of projection format conversion and video quality metrics in 360lib version 5. Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 8th Meeting (2017)
15. Meddeb, M., Cagnazzo, M., Pesquet-Popescu, B.: Region-of-interest based rate control scheme for high efficiency video coding. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2014)
16. Fu, T., Zhang, H., Mu, F., Chen, H.: Fast cu partitioning algorithm for h.266/vvc intra-frame coding. *IEEE International Conference on Multimedia and Expo (ICME)* (2019)
17. An, J., Huang, H., Zhang, K., Huang, Y.-W., Lei, S.: Quadtree plus binary tree structure integration with jem tools. Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 2nd Meeting (2016)
18. Roodaki, H., Irvani, Z., Hashemi, M.R., Shirmohammadi, S.: A view-level rate distortion model for multi-view/3d video. *IEEE Transactions on Multimedia* **18**, 14–24 (2015)
19. Seferidis, V.E., Ghanbari, M.: General approach to block-matching motion estimation. *Optical Engineering* (1993)
20. Li, L., Li, H., Liu, D., Li, Z., Yang, H., Lin, S., Chen, H., Wu, F.: An efficient four-parameter affine motion model for video coding. *IEEE Transactions on Circuits and Systems for Video Technology* **28**, 1934–1948 (2018)
21. Versatile Video Coding (VVC). <https://jvet.hhi.fraunhofer.de> Accessed October 2020

7 Declarations

7.1 Abbreviations

VVC: Versatile Video Coding
 CTU: Coding Tree Unit
 VTM: VVC Test Model
 JVET: Joint Video Experts Team

7.2 Ethical Approval and Consent to participate
 Not applicable.

7.3 Consent for publication
 Not applicable.

7.4 Availability of data and materials
 The datasets supporting the conclusions of this article are included within the article and its additional files.

7.5 Competing interests
 The author declares she has no competing interests.

7.6 Funding
 The author received no specific funding for this work

7.7 Authors' contributions
 Hoda Roodaki works all things about this paper. The author read and approved the final manuscript.

7.8 Acknowledgments
 Not applicable.

7.9 Authors' information
 Hoda Roodaki was born in Tehran, Iran in 1982. She received the B.S. and M.S. degrees in Computer engineering from the University of Tehran, and Sharif University of Technology, in 2005 and 2007 and the Ph.D. degree in Computer Architecture from university of Tehran, Iran in 2014. Since 2015, she has been an Assistant Professor with the Computer Engineering Department, K. N. Toosi University, Tehran, Iran. Her research interests include Multi-view/3D and 360-degree video coding, Scalable video coding, video quality assessment and cloud gaming.

Figures

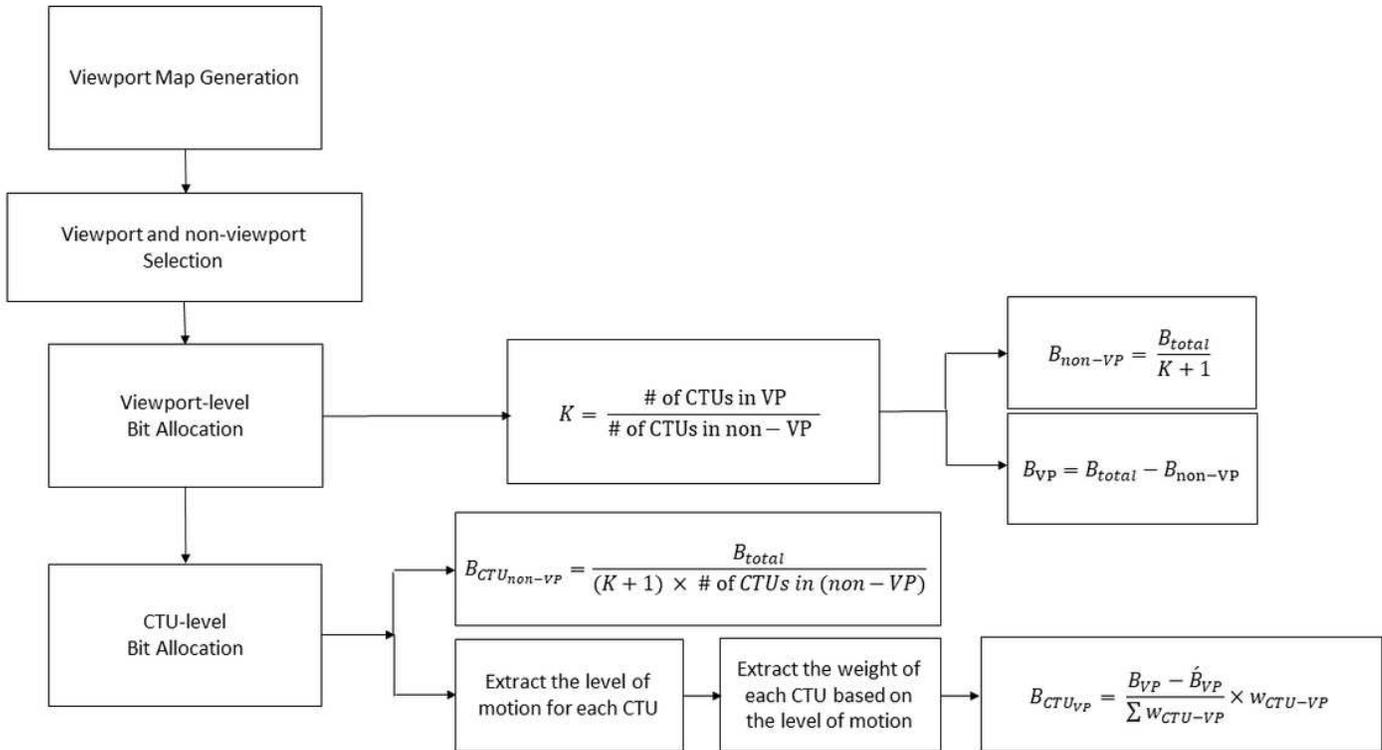


Figure 1

The main scheme of the proposed approach



Figure 2

A scene from the test sequence "PortoRiverside" from the dataset presented in [13]

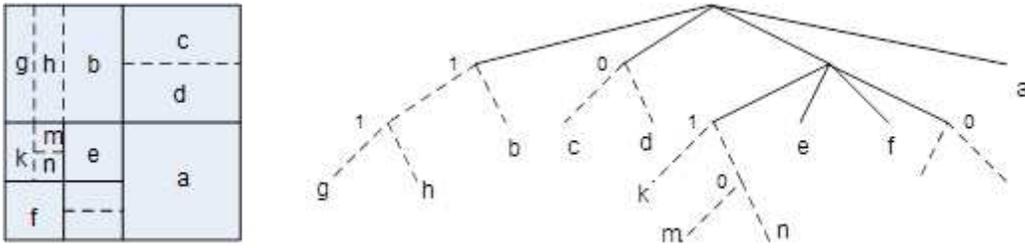


Figure 3

Illustration of a Quad Tree with nested Multi-Type tree coding (QTMT) block structure [17]

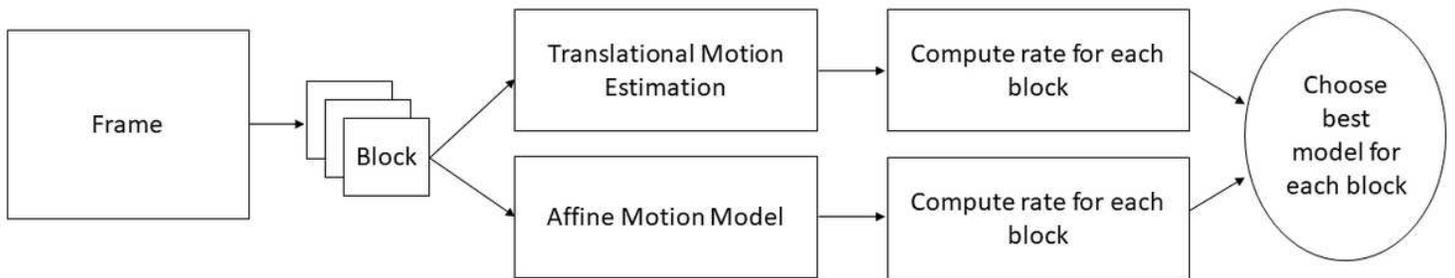


Figure 4

Motion model predictor in VVC standard

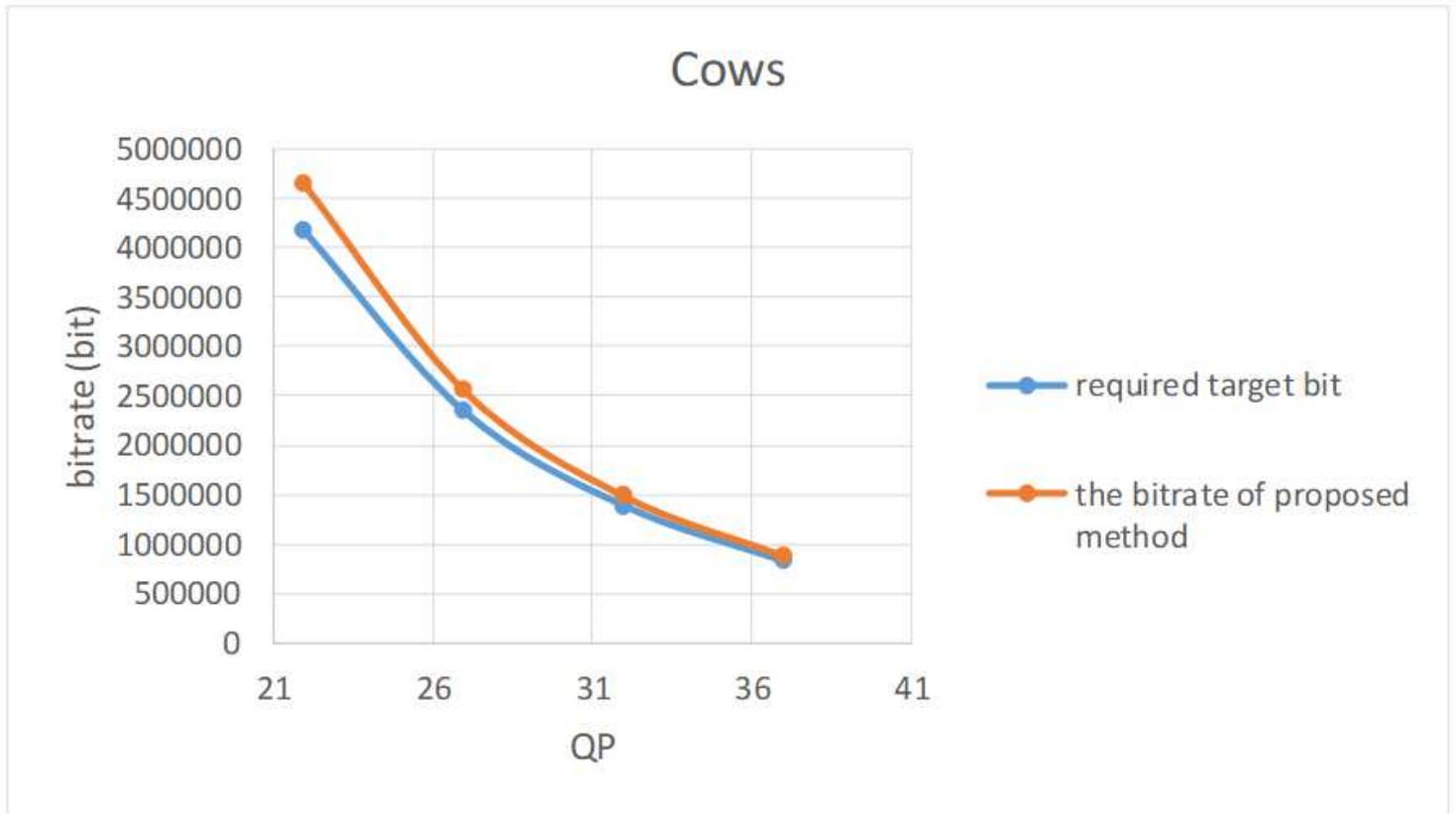


Figure 5

Comparison of the target extracted bitrate using our proposed method and the required target bitrate for Cows video sequence

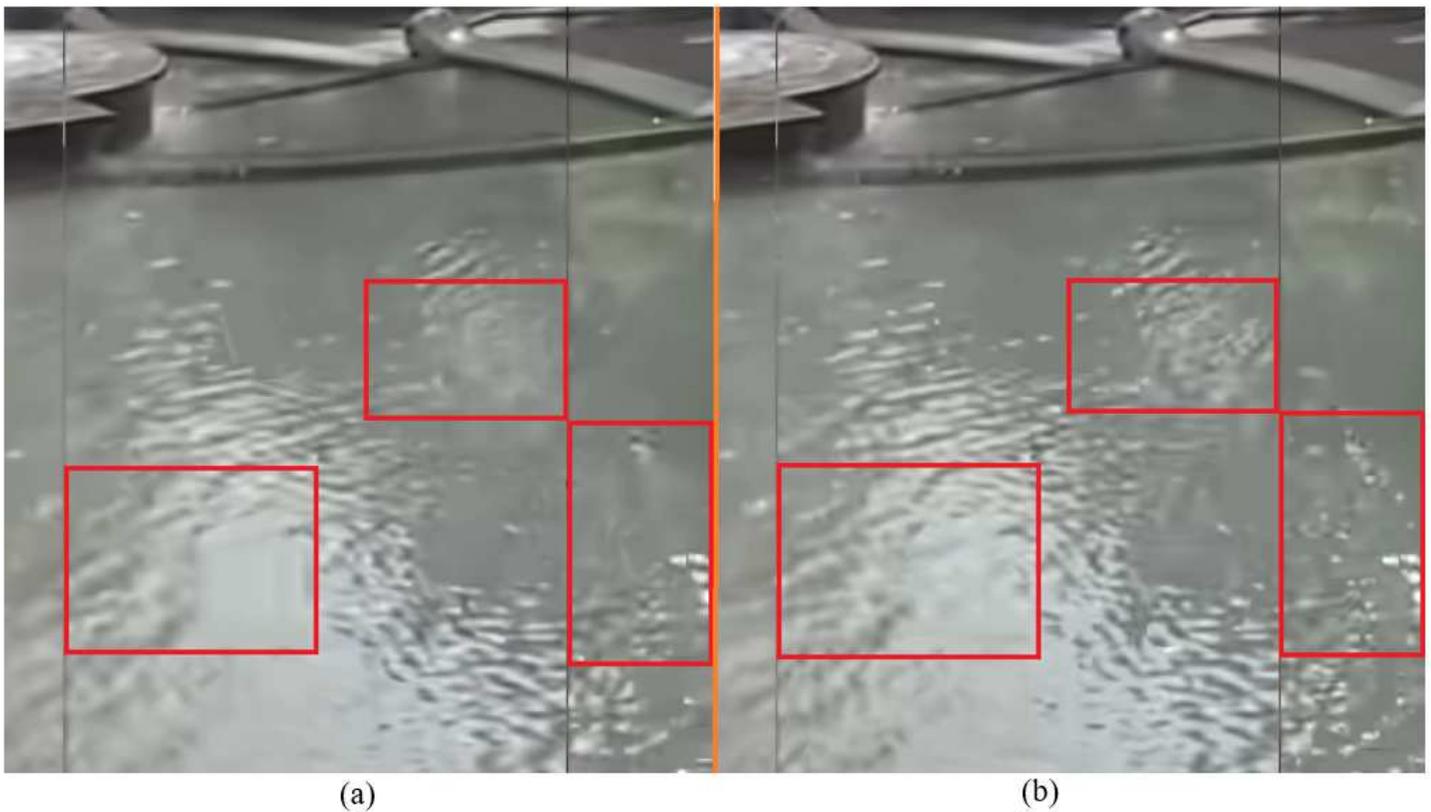


Figure 6

Comparisons of subjective quality of (a) anchor method and (b) our proposed method for Fountain sequence



Non-viewport part



Viewport part

(a)



Non-viewport part



Viewport part

(b)

Figure 7

Comparison of (a) the results of the method proposed in [7] and (b) the result of our proposed method