NOVEL DISTORTION METRIC FOR DEPTH CODING OF 3D VIDEO

Rui Ma[1] *Ngai-Man Cheung*[2] *Oscar C. Au*[1] *Dong Tian*[3]

[1] The Hong Kong University of Science and Technology
[2] Singapore University of Technology and Design
[3] Mitsubishi Electric Research Laboratories

ABSTRACT

In state-of-the-art HEVC-based 3D video codec, multiview video plus associated depth maps are used. In order to achieve better coding performance, instead of the conventional sum of squared errors (SSE), view synthesis optimization (VSO) is proposed and included in the anchor encoder software to calculate view synthesis distortion in rate-distortion optimization (RDO) of depth coding. The anchor VSO achieves high rate-distortion (RD) performance. However, it requires partial rendering and is quite complex and time-consuming. On the other hand, simple SSE metric is fast but RD performance is low. In this paper, we propose a new distortion metric to be used in RDO for depth coding. The complexity of the proposed method is slightly higher than SSE, while its RD performance remains competitive. With a good trade-off between complexity and performance, the proposed method can replace the conventional SSE metric in RDO for depth coding, and can be used as a low-complexity alternative to the anchor VSO.

Index Terms— Distortion metric, rate-distortion optimization, depth coding, view synthesis distortion, 3D video

1. INTRODUCTION

The state-of-the-art video coding standard is High Efficiency Video Coding (HEVC) [1]. It is developed by the Joint Collaborative Team on Video Coding (JCT-VC) of the ITU-T Visual Coding Experts Group (VCEG) and the ISO/IEC Moving Pictures Experts Group (MPEG) as the successor of H.264/AVC. The 3D HEVC extension is an on-going effort for HEVC-based 3D video coding [2]. 3D video is represented in multiview-video-plus-depth (MVD) format, in which a small number of captured views known as base (reference) views together with associated depth maps are coded. The resulting bitstream packets are multiplexed into a 3D video bitstream. After decoding the video and depth data, additional intermediate views between base views suitable for displaying the 3D content on an auto-stereoscopic display can be synthesized using depth-image-based rendering (DIBR) techniques. These intermediate views are called synthesized views. For the purpose of view synthesis, camera parameters are additionally included in the bitstream [3].

Different from texture pictures, depth maps are not directly visible for a viewer. Instead, they are used in rendering of the synthesized views. In particular, the base views are warped to the virtual view locations using depth map data. Hence, lossy coding of depth data will cause distortion in intermediate synthesized views, as the pixels in the base views will be copied to slightly-shifted positions in the synthesized views. Considering this, new distortion metric has been proposed in HEVC-based 3D video coding in addition to the original sum of squared error (SSE) metric, and these metrics are used in rate-distortion optimization (RDO) of depth map coding. Specifically, in the anchor (reference) software [4], an additional metric called synthesized view distortion change (SVDC) is included. SVDC measures the change in distortion when a *reconstructed* depth map block is used instead of the original depth map block during rendering. Partial rendering is required in order to compute SVDC, and this significantly increases the complexity and encoding time. To reduce the complexity, a model-based synthesized view distortion estimation is proposed [5] to combine with SVDC. With this, rendering operations are required only in certain situations. Encoding complexity can be reduced, although it still remains high.

In this paper, we present a new distortion metric in RDO for depth map coding. The proposed method estimates view synthesis distortion without rendering. Since the time-consuming partial rendering process is totally avoided, the encoding time is significantly reduced. The encoding complexity of the proposed method is close to the conventional SSE metric, while the RD performance remains competitive. With a good trade-off between complexity and performance, the proposed method could be used as an low-complexity alternative to the anchor software and as a replacement of SSE metric in RDO for depth coding.

2. PREVIOUS WORK

In the 3D HEVC anchor software [4], a new view synthesis optimization (VSO) encoding option can be used for depth coding. In VSO, the distortion metric for depth data is changed from the conventional SSE to synthesized view distortion change (SVDC) [3]. SVDC is defined as the change in distortion ΔD when a reconstructed depth map block is used instead of the original depth map block during rendering:

$$\begin{aligned}\Delta D &= \tilde{D} - D \\ &= \sum_{(x,y)\in I} [\tilde{S}(x,y) - S_{Ref}(x,y)]^2 - \\ &\quad \sum_{(x,y)\in I} [S(x,y) - S_{Ref}(x,y)]^2.\end{aligned} \tag{1}$$

Here I represents the set of all sample pixels in the synthesized view. S_{Ref} denotes a reference texture rendered from original video and original depth. S denotes a texture rendered from a depth map s_D consisting of encoded depth data in already encoded blocks and original depth data in the other blocks. $\tilde{S}$ denotes a texture rendered from a depth map $\tilde{s}_D$, with $\tilde{s}_D$ different from the depth map s_D in that *reconstructed* depth data is used in the current block instead of the original depth data. The video pictures used in rendering

S and $\tilde{S}$ are reconstructed video pictures, if they are available, otherwise, original video pictures are used. SVDC is selectively used in steps related to the mode decision, coding unit partitioning, motion parameter inheritance and merging. To measure SVDC, partial rendering is used in the anchor software. With SVDC, higher RD performance can be achieved, but the complexity of the encoding would significantly increase.

Model based synthesized view distortion estimation [5] is proposed to combine with SVDC to reduce the complexity. This model based estimation computes view synthesis distortion (VSD) defined as follows

$$\begin{aligned} VSD = & \sum_{(x,y)\in B} (\frac{1}{2} \cdot \alpha \cdot |s_D(x,y) - \tilde{s}_D(x,y)| \cdot \\ & [|\tilde{s}_T(x,y) - \tilde{s}_T(x-1,y)| + |\tilde{s}_T(x,y) - \tilde{s}_T(x+1,y)|]^2), \end{aligned} \tag{2}$$

where B denotes the current block. s_D and $\tilde{s}_D$ denote the original and reconstructed depth data, respectively. $\tilde{s}_T$ denotes the reconstructed texture. α is the coefficient determined by camera parameters relating the depth difference to disparity difference. As rendering is not required, VSD can be computed with low complexity.

The anchor software combines SVDC and VSD to obtain a trade-off between complexity and performance. In particular, VSD is used in lieu of SVDC in intra-mode pre-selection and residual quadtree partitioning. However, the time-consuming partial rendering process is still used in many situations, and the encoding complexity remains high. In this paper, an alternative distortion metric without rendering is proposed. This significantly reduces the encoding time since the time-consuming partial rendering can be totally avoided. The proposed method is as simple as SSE metric, while the RD performance remains competitive.

Other algorithms have been proposed to estimate the synthesis quality in other contexts. Nguyen and Do [6] analyzed the rendering quality of image-based rendering (IBR) algorithms and used Taylor series expansion to derive the upper bound of the mean absolute error (MAE) in the synthesis output. Liu et al. [7] approximated errors due to depth map artifacts using a linear model of average magnitude of mean-squared disparity errors over an entire frame and a motion sensitivity factor computed from the energy density. An autoregressive model was proposed by Kim et al. [8] to estimate the synthesis distortion at the block level and was shown to be effective for rate-distortion optimized mode selection. A distortion model as a function of the view location was also proposed by Velisavljevic et al. [9] for bit allocation. Takahashi [10] proposed an optimized view interpolation scheme based on frequency domain analysis of depth map error. Cheung et al. [11] proposed to estimate the synthesis quality using power spectral density (PSD).

3. PROPOSED DISTORTION METRIC

3.1. New distortion metric in RDO for depth maps

As mentioned in Section 1, depth maps are not directly visible. Instead, they are used in rendering of synthesized views, where base views are warped to the virtual view locations. In particular, disparities are computed based on depth values, and are used to determine the amount of pixel shift from base views to virtual views in the warping process. The relationship between depth value $s_D(x,y)$ and disparity $p(x,y)$ at the position (x,y) is

$$p(x,y) = s \cdot s_D(x,y) + o. \tag{3}$$

Here s and o are the scaling factor and the offset, respectively. s and o are determined by camera parameters as follows

$$s = \frac{f \cdot b}{255}\left(\frac{1}{Z_{near}} - \frac{1}{Z_{far}}\right), \quad o = \frac{f \cdot b}{Z_{far}}, \tag{4}$$

where f is the focal length. b is the baseline between base view and synthesized view. Z_{near} and Z_{far} represent the nearest and farthest depth value of the scene, respectively.

In standard 3D test sequences, the cameras are rectified and arranged linearly, and there exists only horizontal disparity. Considering an original depth pixel $s_D(x,y)$, the corresponding disparity determined by (3) is $p(x,y)$. During rendering, the corresponding texture pixel $\tilde{s}_T(x,y)$ will be shifted horizontally by $p(x,y)$ in the synthesized view S. Thus,

$$S\left(x + p(x,y), y\right) = \tilde{s}_T(x,y). \tag{5}$$

Similarly, given the reconstructed depth pixel $\tilde{s}_D(x,y)$ and disparity $\tilde{p}(x,y)$, the corresponding texture pixel $\tilde{s}_T(x,y)$ will be shifted horizontally by $\tilde{p}(x,y)$ in the synthesized view $\tilde{S}$. Thus,

$$\tilde{S}(x + \tilde{p}(x,y), y) = \tilde{s}_T(x,y). \tag{6}$$

The difference between S and $\tilde{S}$ caused by $\tilde{p}(x,y)$ is then computed as

$$Dist(x,y) = [\tilde{S}(x + \tilde{p}(x,y), y) - S(x + \tilde{p}(x,y), y)]^2. \tag{7}$$

Substitute (5) and (6) into (7), we derive

$$Dist(x,y) = [\tilde{s}_T(x,y) - \tilde{s}_T(x + \Delta x, y)]^2, \tag{8}$$

where we define

$$\tilde{s}_T(x + \Delta x, y) = S(x + \tilde{p}(x,y), y). \tag{9}$$

Here Δx is an unknown shift to be determined. Note that in the distortion function (8), Δx is the only unknown variable. So the problem is changed to find the value of Δx. Using (5), we can easily get

$$\begin{aligned} & \tilde{s}_T(x + \Delta x, y) \\ & \quad = S(x + \Delta x + p(x + \Delta x, y), y). \end{aligned} \tag{10}$$

Compare (9) and (10), it's observed that

$$\tilde{p}(x,y) = \Delta x + p(x + \Delta x, y). \tag{11}$$

So we have

$$\begin{aligned} \Delta x = & \ \tilde{p}(x,y) - p(x + \Delta x, y) \\ = & \ [\tilde{p}(x,y) - p(x,y)] - \\ & \ [p(x + \Delta x, y) - p(x,y)] \\ = & \ \Delta p(x,y) - [p(x + \Delta x, y) - p(x,y)]. \end{aligned} \tag{12}$$

In the above equation, the left side is the texture shift to be determined. The right side contains two terms. The first term is the disparity difference between the original and reconstructed depth data. The second term is the disparity difference between neighbouring pixels in the original depth map. Usually, disparity difference of neighbouring pixels is very small or equal to zero, since the depth map is smooth in most places. So we set the second term equal to zero. Thus,

$$\Delta x \approx \Delta p(x,y), \tag{13}$$

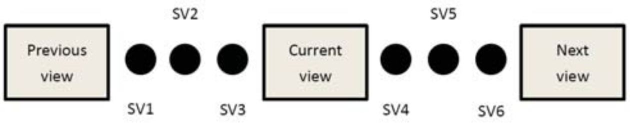

Fig. 1. Synthesized views

and the distortion function (8) is changed into

$$Dist(x,y) = [\tilde{s}_T(x,y) - \tilde{s}_T(x + \Delta p(x,y), y)]^2. \tag{14}$$

Finally, the total distortion of a block is calculated as

$$\begin{aligned} Dist &= \sum_{(x,y)\in B} Dist(x,y) \\ &= \sum_{(x,y)\in B} [\tilde{s}_T(x,y) - \tilde{s}_T(x + \Delta p(x,y), y)]^2 \cdot c, \end{aligned} \tag{15}$$

where c is an adjusting constant. We propose to use (15) to estimate view synthesis distortion.

3.2. Further modification of the distortion function

The new distortion function in (15) considers only a single virtual view location. However, the decoded depth data is used to generate the synthesized views for multiple virtual view locations between base views. Therefore, we modify (15) by considering the distortion of the synthesized views at 6 different virtual view locations. Fig. 1 shows the 6 synthesized views, denoted by SV1 to SV6, respectively. Each three of them are placed with equal interval between the current and neighboring base views.

Let the disparity between current view and SV2 be Δp. Since disparity is proportional to the distance between the current base view and synthesized view as suggested by (3) and (4), disparities of all 6 virtual view locations can be calculated. They are

$$\{\frac{3}{2}\Delta p, \Delta p, \frac{1}{2}\Delta p, -\frac{1}{2}\Delta p, -\Delta p, -\frac{3}{2}\Delta p\},$$

following the synthesized view numbers. Based on these disparities, the distortion function (15) is modified to

$$\begin{aligned} Dist = \sum_{(x,y)\in B} (& [\tilde{s}_T(x,y) - \tilde{s}_T(x + \frac{3}{2}\Delta p(x,y), y)]^2 + \\ & [\tilde{s}_T(x,y) - \tilde{s}_T(x + \Delta p(x,y), y)]^2 + \\ & [\tilde{s}_T(x,y) - \tilde{s}_T(x + \frac{1}{2}\Delta p(x,y), y)]^2 + \\ & [\tilde{s}_T(x,y) - \tilde{s}_T(x - \frac{1}{2}\Delta p(x,y), y)]^2 + \\ & [\tilde{s}_T(x,y) - \tilde{s}_T(x - \Delta p(x,y), y)]^2 + \\ & [\tilde{s}_T(x,y) - \tilde{s}_T(x - \frac{3}{2}\Delta p(x,y), y)]^2) \cdot c. \end{aligned} \tag{16}$$

Finally, we propose to use the modified distortion metric in (16) to estimate view synthesis distortion.

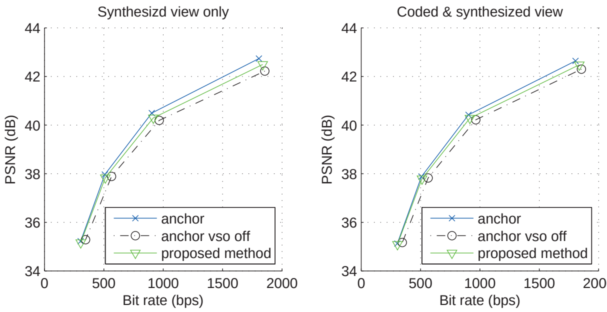

Fig. 2. RD performance comparison - sequence ”Balloons”

4. EXPERIMENTAL RESULTS

The performance of the proposed method is evaluated by comparing it to the 3D HEVC anchor software [4]. Two modes of the anchor software are evaluated. The first one is the anchor VSO mode, which combines SVDC and model-based estimation VSD. The second is the anchor with VSO turned off for depth maps, where conventional SSE metric is used.

The experiments were conducted using test sequences and test conditions specified in common test conditions [12]. Multiview video sequences with associated depth maps are used, with two resolution class 1024x768 and 1920x1088. Four texture QP values (40, 35, 30, 25) for independent view are tested. Depth QP values are fixed with texture QP values defined in [12]. As specified in common test conditions, synthesized views are rendered between coded views using the decoded (reconstructed) texture and decoded depth. The generated synthesized views are compared to synthesized views that are rendered using the original texture and original depth. The PSNR values of the synthesized and actually coded views are calculated. Then, together with the overall bit rates, the average bit rate savings for different synthesized views is computed using Bjøntegaard delta rate (BD-rate) [13]. The encoding time is also evaluated, and compared with the anchor VSO.

In Fig. 2, the RD curves of sequence ”Balloons” are evaluated for different methods. ”Anchor vso off” represents the anchor software with VSO turned off for depth maps, where SSE metric is used instead. Average PSNR values of synthesized views and actually coded views plus synthesized views are calculated, respectively. The result shows that while the performance of our method is not as good as anchor VSO, it outperforms conventional SSE.

Table 1 and Table 2 show the percentage of BD-rate gains and losses against the anchor (VSO on) for the proposed method and anchor VSO off, respectively. We can see that with the VSO turned off, the conventional SSE metric provides rather low performance with 17.4% loss in synthesized view and 12.9% loss in coded and synthesized views. The proposed method gives 7.3% loss in synthesized view and 5.3% loss in coded and synthesized views. With the proposed method, 58% of the coding performance gap between SSE and the anchor method can be recovered.

Importantly, the proposed method has very low complexity. Table 3 lists the encoding time percentage against the anchor. The proposed method takes 86.1% of the anchor VSO encoding time on average, while the conventional SSE takes 81.5%. Note that SSE is almost the simplest metric. The results indicate that the complexity of proposed method is slightly higher than the simplest metric with a 5.7% increase. It is very important to notice that VSO is only

Table 1. BD-rate evaluation of proposed method

	synthesized view only	coded & synthesized view
Balloons	5.2%	3.8%
Kendo	7.3%	5.6%
Newspapercc	12.3%	9.6%
GhostTownFly	10.1%	6.8%
PoznanHall2	6.2%	4.6%
PoznanStreet	3.2%	2.5%
UndoDancer	6.9%	4.5%
1024x768	8.2%	6.3%
1920x1088	6.6%	4.6%
average	**7.3%**	**5.3%**

Table 2. BD-rate evaluation of anchor VSO off

	synthesized view only	coded & synthesized view
Balloons	14.8%	12.9%
Kendo	24.3%	22.3%
Newspapercc	20.2%	14.3%
GhostTownFly	14.4%	9.3%
PoznanHall2	20.2%	14.3%
PoznanStreet	8.5%	5.8%
UndoDancer	19.4%	11.4%
1024x768	19.7%	16.5%
1920x1088	15.6%	10.2%
average	**17.4%**	**12.9%**

one module in the encoding pipeline of depth maps, with other modules such as motion estimation. Table 4 lists the percentage of VSO processing time in total encoding time and the average is about 18.6%. The proposed distortion metric, on the other hand, takes only 6.0% of the total encoding time on average. That translates into a significant 72.1% reduction when compared to VSO processing time. Note that for other modules in the encoding pipeline (e.g., motion estimation), many acceleration ideas have been proposed and can be applied to depth encoding. Therefore, VSO shall become an important bottleneck in the encoding and our proposed method can accelerate this bottleneck. Note also that the proposed method can be combined with other mode prediction techniques to further reduce the complexity of RD optimized mode decision in depth map coding.

Moreover, in HEVC codec, the encoder is much more complex than the decoder. Consequently, encoding time is hundreds times of decoding time. Saving encoding time becomes significantly important. The proposed method can reduce VSO processing time by 72.1% and the total encoding time by 13.9% with reasonable RD performance. If other modules in the encoding such as motion estimation are optimized, the percentage reduction in total encoding time using the proposed method would likely increase considerably. From this point of view, the proposed method offers a good trade-off between encoder complexity and RD performance, and it can be used as a low-complexity alternative to the anchor VSO.

Table 3. Encoding time percentage against the anchor VSO

	Proposed method	anchor VSO off
Balloons	88.1%	83.8%
Kendo	87.7%	84.1%
Newspapercc	85.5%	75.0%
GhostTownFly	85.9%	80.7%
PoznanHall2	85.3%	83.2%
PoznanStreet	84.6%	80.4%
UndoDancer	85.9%	83.7%
1024x768	87.1%	80.9%
1920x1088	85.4%	82.0%
average	**86.1%**	**81.5%**

Table 4. Percentage of VSO processing time in total encoding time

	Proposed method	anchor VSO	Time reduction compared with anchor VSO
Balloons	6.5%	18.5%	69.4%
Kendo	5.9%	16.5%	68.0%
Newspapercc	8.7%	22.0%	66.4%
GhostTownFly	5.4%	17.9%	73.7%
PoznanHall2	4.2%	17.5%	79.6%
PoznanStreet	7.1%	20.7%	70.9%
UndoDancer	4.6%	17.1%	76.8%
1024x768	7.0%	19.0%	68.9%
1920x1088	5.3%	18.3%	75.3%
average	**6.0%**	**18.6%**	**72.1%**

5. CONCLUSIONS

We presented a new distortion metric for depth map coding in 3D video. A simple model is used to estimate view synthesis distortion, thus the time-consuming rendering process can be avoided. Experiments demonstrated that the proposed method can save 72.1% of VSO processing time. The complexity of the proposed method is slightly higher than conventional SSE metric, but it can recover 58% of the RD performance gap between SSE metric and the anchor VSO. The proposed method can be used as a low-complexity alternative to the anchor VSO and as a replacement of SSE metric for depth coding.

6. REFERENCES

[1] G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, “Overview of the high efficiency video coding (HEVC) standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, December 2012.

[2] H. Schwarz, C. Bartnik, and S. Bosse et al., “3D video coding using advanced prediction, depth modeling, and encoder control methods,” *IEEE Intl. Conf. on Image Processing*, Octorber 2012.

[3] G. Tech, K. Wegner, Y. Chen, and S. Yea, “3D-HEVC test model 2,” ISO/IEC JTC1/SC29/WG11, M27310, October 2012.

[4] "Anchor software: 3DV-HTM 4.0," http://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware.

[5] B. T. Oh, J. Lee, and D. S. Park, "3D-CE8.h results on view synthesis optimization using distortion in synthesized views by samsung," ISO/IEC JTC1/SC29/WG11, M24830, May 2012.

[6] H. T. Nguyen and M. N. Do, "Error analysis for image-based rendering with depth information," *IEEE Trans. Image Processing*, vol. 18, no. 4, pp. 703–716, 2009.

[7] Y. Liu, Q Huang, S. Ma, D. Zhao, and W. Gao, "Joint video/depth rate allocation for 3d video coding based on view synthesis distortion model," *Image Commun.*, vol. 24, no. 8, pp. 666–681, Sept. 2009.

[8] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map coding with distortion estimation of rendered view," in *Proc. SPIE Visual Information Processing and Communication (VIPC)*, 2010.

[9] V. Velisavljevic, G. Cheung, and J. Chakareski, "Bit allocation for multiview image compression using cubic synthesized view distortion model," in *Proc. IEEE International Workshop on Hot Topics in 3D*, 2011.

[10] K. Takahashi, "Theoretical analysis of view interpolation with inaccurate depth information," *IEEE Trans. Image Processing*, vol. 21, no. 2, pp. 718–732, 2012.

[11] N.-M. Cheung, D. Tian, A. Vetro, and H. Sun, "On modeling the rendering error in 3D video," in *Proc. IEEE Int'l Conf. Image Processing (ICIP)*, 2012.

[12] D. Rusanovskyy, K. Müller, and A. Vetro, "Common test conditions of 3DV core experiments," ISO/IEC JTC1/SC29 /WG11, M26349, July 2012.

[13] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," 13th VCEG-M33 Meeting, April 2001.