NEW DISTORTION MODEL FOR DEPTH CODING IN 3DVC

Tao Zhang, Xiaopeng Fan, Debin Zhao, Wen Gao

Dept. of Computer Science and technology, Harbin Institute of Technology, Harbin, China
{tzhang, fxp, dbzhao}@hit.edu.cn, wgao@jdl.ac.cn

ABSTRACT

Video plus the corresponding depth maps is now the most popular data format for 3D video coding, as it is convenient to synthesize an arbitrary intermediate virtual view without the need to display the depth map for end users. How to compress the depth map efficiently is one of the key problems in 3DVC. In this paper we propose a new distortion model for depth map coding by considering the view synthesis distortion caused by depth distortion and the disparity rounding problem. Simulation results show that the proposed scheme can achieve 34% bit-rate saving for depth map coding, or 1.28 dB gain in terms of PSNR on average for synthesized virtual views.

Index Terms—depth map coding, view disparity, view synthesis, rate-distortion optimization, 3DVC

1. INTRODUCTION

With the development of the three-dimensional television (3DTV) and free-viewpoint television (FTV), people can get immersive perception of 3-D scene. Recently, Call for Proposals (CfP) [1] on 3D Video Coding (3DVC) [2] technology has been issued by ISO/IEC JTC1/SC29/WG11 (MPEG), to provide efficient compression and high quality view reconstruction of an arbitrary number of dense views. In order to represent the 3-D scene, a new data format (MVD) that includes multi-view video and the corresponding depth maps has been proposed and widely used. It is convenient to synthesize intermediate virtual views by using depth image based rendering (DIBR) technique [3] in the receiver side.

In current 3D video system, a large amount of video and the corresponding depth maps should be compressed and transmitted to the receiver side. Thus efficient coding methods for video and depth maps are desirable. Conventional coding standards, such as H.264/AVC, MVC, used to compress the video information. However, depth maps have different characteristics than video signal. Firstly, the value in depth map represents the distance between an object in a scene and the camera and is stored in 8-bit gray scale format. Furthermore, depth maps are composed of many smooth regions and those smooth regions are separated by sharp edge, and the edge information is sensitive to render virtual views. Moreover, depth maps are not directly displayed but used in view rendering. Because of the characteristics of depth maps, conventional coding techniques are not suitable to code depth maps any more. To compress the depth maps efficiently, many techniques have been proposed. They can be classified by two categories.

The first category directly uses the characteristics of depth maps. In [4], an edge-aware intra prediction was proposed to reduce the prediction error energy in blocks with arbitrary edge shapes. In [5], a platelet-based depth coding method that separates continuous regions by straight lines along their boundaries by modeling them with piecewise-constant or piecewise-linear functions was proposed. In [6], shape adaptive wavelet based coding method was proposed. Another method that utilizes the structure similarity between depth and corresponding video was proposed in [7]. As we can see, all of these methods make great efforts to preserve the depth boundaries since it is sensitive to view rendering, but their coding efficiency is not very high since they encode the depth maps independently.

The second category is MVC-based or H.264-based methods. [8] presented a method for efficient compression of multi-view depth data based on multi-view video coding approach for color data. A fast and efficient multi-view depth image coding method using the texture images was proposed in [9]. In [10], a new H.264-based coding algorithm for the depth map sequence using motion information of the corresponding texture video was proposed. As we can see, all of these methods are efficient to compress depth maps since they use the video information to some extent. However, these methods don't fully consider the special characteristics of depth maps.

As we know, depth maps are not directly displayed to viewers, they are used to render virtual views. Hence, how to compress depth maps to ensure a high quality of

This work was supported in part by the Major State Basic Research Development Program of China (973 Program 2009CB320905), the Program for New Century Excellent Talents in University (NCET) of China (NCET-11-0797), the National Science Foundation of China (NSFC) under grants 61100095 and the Fundamental Research Funds for the Central Universities (Grant No. HIT.BRETIII.201221).

rendering views is what we should consider more rather than only consider the quality of depth maps. In [11] a new distortion model that takes into consideration camera parameters and global video characteristics in order to quantify the effect of coding of depth map on synthesized view quality was proposed. This method was extended in [12], in which the local characteristics of the videos were considered. However, these methods didn't consider the disparity rounding problem in view rendering. In this paper, we analyze the relationship between the depth coding distortion and the synthesized view distortion and propose a more accurate distortion model for mode decision in depth coding. We first propose a more accurate distortion model to measure the distortion of disparity by considering the rounding effects in view rendering. Then we investigate the relation between the distortion of the depth map and the distortion of the rendered virtual view, and propose a new linear relation model with low complexity but enough accuracy. Finally, we propose to use the modified Lagrangian functions [12] for RD optimization, considering that the view synthesis uses two reference frames and two depth maps.

The rest of this paper is organized as follows. Section 2 introduces the view synthesis techniques: Depth-image-based rendering (DIBR) and the effect of depth map distortion on the synthesized view will also be described. The proposed method is presented in Section 3. Experiments are performed using various test sequences and results are discussed in Section 4. Conclusions are given in Section 5.

2. THE EFFECT OF DEPTH MAP DISTORTION ON THE SYNTHESIZED VIEW

Depth-image-based rendering (DIBR) is used to synthesize virtual views in the current view synthesis reference software [14] with the reference video and corresponding depth maps. DIBR projects the video pixel value from reference video to the virtual view with the help of depth value. We assume that a pixel position in reference view is $\mathrm{r}(\mathrm{x_r}, \mathrm{y_r})$ and the corresponding position in virtual view is $v(x_v, y_v)$. A view-warping from r to v based on geometric camera parameters can be divided into two steps.

Firstly, $r(x_r, y_r)$ is projected into world coordinate $P(u, v, w)$ with depth of reference view by

$$[u,v,w]^T = \mathbf{R}_r \cdot \mathbf{A}_r^{-1} \cdot [x_r, y_r, 1]^T \cdot Z_r(x_r, y_r) + \mathbf{T}_r \quad (1)$$

where **A** is an intrinsic parameter matrix, **R** is a rotation matrix and **T** is a translation vector. Z_r represents the real depth value in position r. The subscript r indicates the reference view. The relationship of real depth value Z_r and the value $D_r(x,y)$ in depth map is described by the following equation:

$$\frac{1}{Z_r} = \frac{D_r(x,y)}{255}\left(\frac{1}{Z_{\text{near}}} - \frac{1}{Z_{\text{far}}}\right) + \frac{1}{Z_{\text{far}}}, \quad (2)$$

where Z_{near} and Z_{far} represent the nearest and the farthest depth values in the scene, which correspond to value 255 and 0 in the depth map **D**.

Then the point (u, v, w) in world coordinate is mapped to the point (x', y', z') in the virtual view by

$$[x', y', z']^T = \mathbf{A}_v \cdot \mathbf{R}_v^{-1} \cdot \left\{[u, v, w]^T - \mathbf{T}_v\right\}, \quad (3)$$

where the subscript v indicates the virtual view. By combining (1) and (3), we can derive that

$$[x', y', z']^T = \mathbf{A}_v \cdot \mathbf{R}_v^{-1} \cdot \left\{\mathbf{R}_r \cdot \mathbf{A}_r^{-1} \cdot [x, y, 1]^T \cdot Z_r(x_r, y_r) + \mathbf{T}_r - \mathbf{T}_v\right\}. \quad (4)$$

The corresponding pixel location in the rendered image is $(x_v, y_v) = \left(x'/z', y'/z'\right)$. For 3D video systems with a 1D parallel camera arrangement, we can get

$$(x_v, y_v) = \left(\frac{xZ_r(x,y) + a \cdot \delta x}{Z_r(x,y)}, y\right) = \left(x + \frac{a \cdot \delta x}{Z_r(x,y)}, y\right), \quad (5)$$

where a is the focal length of the camera in the horizontal direction with the unit of pixel, δx is the distance between two cameras (horizontal). The term $d = a \cdot \delta x / Z_r(x,y)$ in equation (5) is the disparity in horizontal direction.

When the depth value of pixel r is encoded and the reconstructed real depth value is

$$\hat{Z}_r(x,y) = Z_r(x,y) + \Delta Z_r(x,y). \quad (6)$$

With the reconstructed real depth value, the pixel location in the rendered image is $v'(x_v', y_v')$

$$\left(x_v', y_v'\right) = \left(x + \frac{a \cdot \delta x}{\hat{Z}_r(x,y)}, y\right). \quad (7)$$

We set $\hat{d} = a \cdot \delta x / \hat{Z}_r(x,y)$ which is the disparity value for the reconstructed depth value.

The rendering position error (geometry distortion) [11] [12] is

$$\begin{aligned}\Delta x_v &= x_v' - x_v = \frac{a \cdot \delta x}{\hat{Z}_r(x,y)} - \frac{a \cdot \delta x}{Z_r(x,y)} \\ &= \hat{d} - d = a \cdot \delta x \cdot \frac{\Delta D_r(x,y)}{255}\left(\frac{1}{Z_{\text{near}}} - \frac{1}{Z_{\text{far}}}\right), \end{aligned} \quad (8)$$

where $\Delta D_r(x,y)$ is the depth coding error in position r.

3. PROPOSED METHOD FOR DEPTH CODING

From the analysis in the previous section, the geometry distortion in the rendered view could be evaluated by equation (8). However, the geometry distortion metric is not accurate because it doesn't consider the disparity rounding problem. In this section, we first derive a more accurate relationship between coding errors in the depth map and geometry errors in view rendering, then propose a new distortion model that considers synthesis distortions caused by depth map compression and use this distortion model in rate-distortion optimization of depth compression.

3.1. More accurate geometry distortion

As we know, the disparity $a \cdot \delta x / Z_r(x,y)$ is non-integer in most cases and it needs to be rounded for determining the warped pixel position. Therefore, directly doing the subtraction between the disparities is not appropriate in equation (8). It can't reflect the real geometry distortion in view rendering. In this paper, we consider λ-rounding with $1/M$-precision method [13]. If a pixel is allowed to be mapped to sub-pixel position in the virtual view based on the disparity function, the pixel position after projection is rounded to a neighbor position $1/M$ in the sub-pixel sample. We often set λ=0.5/M. The disparity λ-rounding function $Round(d)$ can be written as

$$Round(d) = \frac{\lceil (d-\lambda)\cdot M \rceil}{M}. \quad (9)$$

The rounding of the disparity d in (5) determines the exact position of a pixel after the pixel mapping. The geometry distortion in the rendered view could be derived as follow:

$$|\Delta p(n)| = \left| Round(d) - Round(\hat{d}) \right|. \quad (10)$$

This equation reflects the accurate geometry distortion in the rendered view caused by the distortion of depth map coding.

3.2. Proposed model for estimating distortion in rendered view

In a depth image based rendering (DIBR) system, a virtual view can be generated by some reference video frames and their corresponding depth maps. When compressing a depth map, the distortion in rendered view can be approximated using the corresponding video frame. Assume **X** is the block in the video frame collocated with the depth map block to be coded, $X(n)$ is the pixel in the block, $n = 0, 1, \ldots, N.N$ is the number of pixel in current block. The distortion $D_v(n)$ in rendered view caused by $D(n)$ in depth map could be calculated approximately as

$$\begin{aligned} D_v(n) &= E\left\{ \left(X(n) - X(n+\Delta p(n)) \right)^2 \right\} \\ &= EX^2(n) - 2E\{X(n)X(n+\Delta p(n))\} + EX^2(n) \\ &= 2\delta^2_{X(n)}(1-\rho^{|\Delta p(n)|}) \end{aligned} \quad (11)$$

where $\delta^2_{X(n)}$ is calculated by the variance of the video block. ρ represents the video block correlation when translated by one pixel. $|\Delta p(n)|$ is the geometry distortion caused by depth distortion which is calculated by equation (10). As we known, ρ is less than 1, and $|\Delta p(n)|$ is not very large. So the equation (11) can be approximated as

$$D_v(n) = 2\delta^2_{X(n)}(1-\rho)|\Delta p(n)| \quad (12)$$

The distortion in rendered view caused by the distortion of current depth block can be expressed as

$$\begin{aligned} D_v &= \sum_{n=1}^{N} 2\delta^2_{X(n)}(1-\rho)|\Delta p(n)| \\ &= 2(1-\rho)\delta^2_{X(n)} \sum_{n=1}^{N} |\Delta p(n)|. \end{aligned} \quad (13)$$

The estimation of distortion in rendered view in [12] is computed as

$$D_v' = 2(N-1)\left(1 - \frac{1}{N}\sum_{n=1}^{N} \rho^{|\Delta p(n)|}\delta^2_{X(n)}\right) \quad (14)$$

Both of the equations (13) and (14) are estimation of distortion in rendered view. They use the same parameters to compute the distortion but they are computed in different ways. Moreover, we use the proposed geometry distortion $|\Delta p(n)|$ in section 3.1 which is more accuracy. It can be seen that our proposed D_v can be computed faster than [12] because we don't have exponential term which is time-consuming. We will do some experiments in section 4 to verify that our proposed estimation model (13) is more accuracy than (14).

In (13), $\delta^2_{X(n)}$ and ρ are derived before computing D_v. Both of the variables reflect the local characteristics of the video. Intuitively, the effect of geometry error on the quality of rendered view will depend on local characteristics of the video. The proposed distortion model also confirms this. In regions of a video frame with complex textures and objects, corresponding to small ρ and large $\delta^2_{X(n)}$, the distortion of rendered view caused by the geometry error will be significant, since different positions should have quite different pixel values. On the other hand, in regions with simple textures or homogeneous regions, corresponding to large ρ and small $\delta^2_{X(n)}$, the distortion due to geometry error will be small since pixels at different positions are similar.

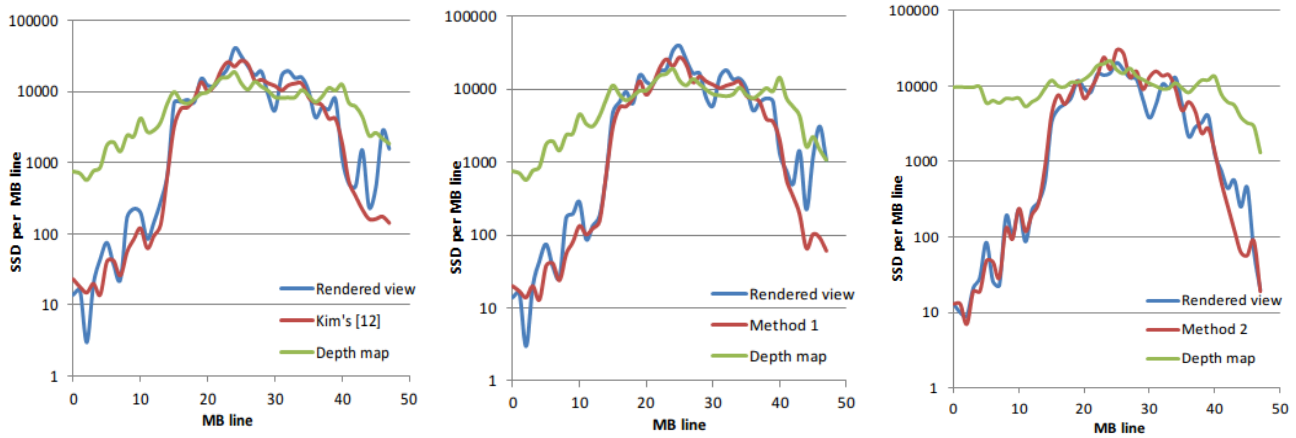
3.3. Modified Lagrangian optimization in the rate distortion optimized mode selection

As depth maps are only used for view rendering and are not directly displayed to end users, the quality of depth map is not so important for us. On the contrary, the quality of rendered view is very important. So we use the estimated D_v to replace the distortion of depth map in Lagrange cost formula, then the formula can be

$$J = D_v + \lambda R_d, \quad (15)$$

where D_v is the estimated distortion in rendered view which can be calculated by (13), λ is the Lagrange multiplier, R_d is the bitrate of encoding the depth maps.

Currently, the view synthesis reference software of MPEG uses two reference video frames and corresponding depth maps as input to synthesize a virtual view. If the two depth maps are encoded together, the distortion is calculated by a weighted sum of the distortion caused by each depth map and the Lagrange cost can be expressed as



(a) Kim's [12] (b) Method 1 (c) Method 2

Fig.1. Estimated distortion using (a) Kim's [12]. (b) Method 1. (c) Method 2. x-axis: MB row index; y-axis: total SSD for all MBs in a row (in log scale).

$$J = (1-w)D_v^{left} + wD_v^{right} + \lambda(R_{depth}^{left} + R_{depth}^{right}) \quad (16)$$

where D_v^{left} and D_v^{right} represent the virtual view synthesis distortion from left camera and right camera depth data, R_{depth}^{left} and R_{depth}^{right} represent the bitrate of the encoding depth map from left camera and right camera, the weight w is defined as

$$w = \frac{|\mathbf{T}_V - \mathbf{T}_L|}{|\mathbf{T}_V - \mathbf{T}_L| + |\mathbf{T}_V - \mathbf{T}_R|} \quad (17)$$

where $\mathbf{T}_L$, $\mathbf{T}_R$, $\mathbf{T}_V$ are the translation vectors of the left camera, the right camera, and the virtual camera, respectively. If the two depth maps are encoded independently from other depth map, the Lagrange cost for encoding the left depth map can be expressed as

$$J = (1-w)D_v^{left} + \lambda R_{depth}^{left} , \quad (18)$$

and the Lagrange cost for encoding the right depth map can be expressed as

$$J = wD_v^{right} + \lambda R_{depth}^{right} . \quad (19)$$

The new Lagrange cost formula is aimed to focus on the distortion of synthesized view to encode depth maps, so the quality of the synthesized view will be improved. The formula can be integrated into current video standards, such as H.264/AVC, MVC, or HEVC to encode depth map efficiently. In the following section, we integrate the proposed methods into H.264/AVC for experiments.

4. EXPERIMENTAL RESULTS

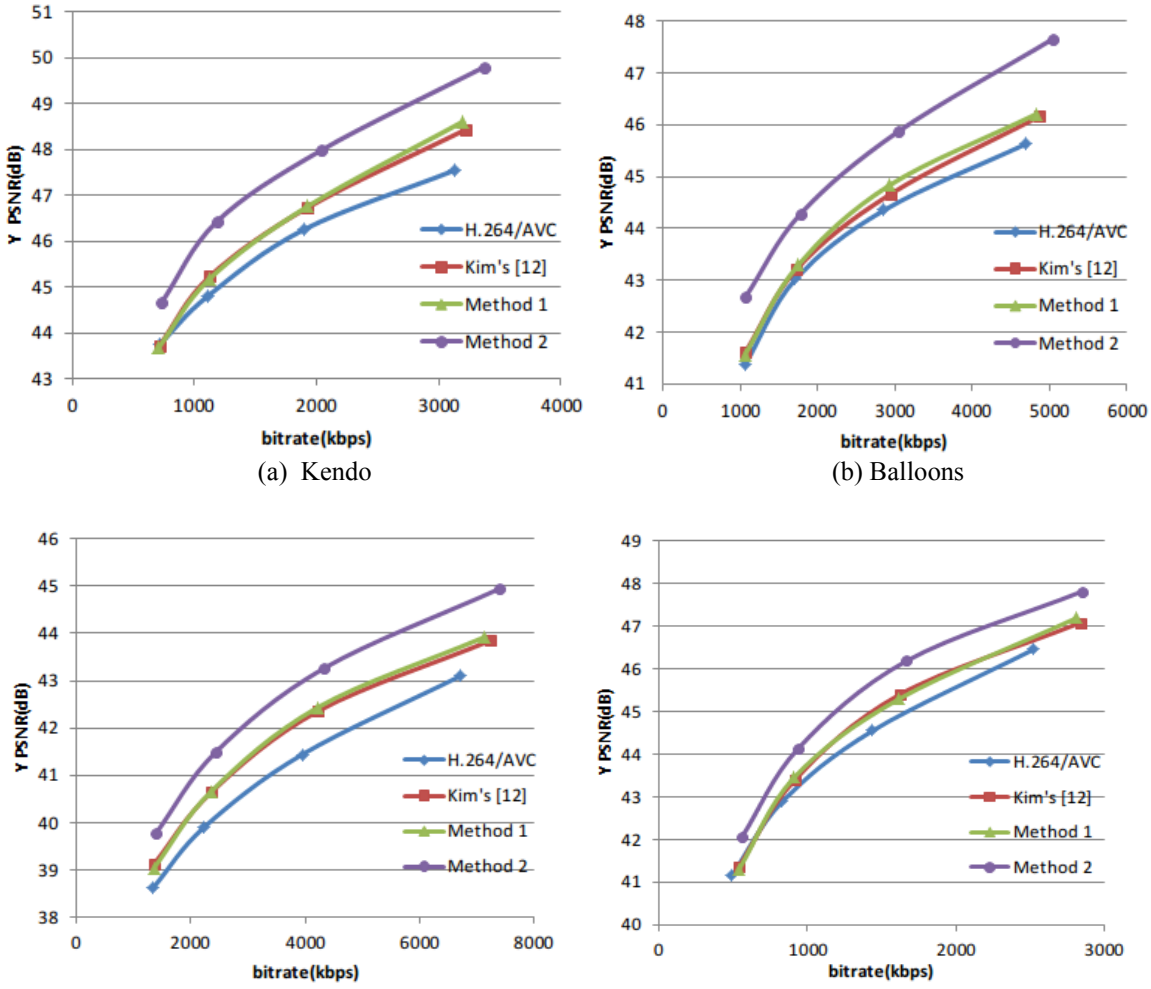
We integrate the proposed distortion model and RD optimized mode selection scheme into H.264/AVC reference software JM version 15.1. In disparity rounding part, we set $M = 1$, $\lambda = 0.5$ in this experiment, so the geometry distortion is

$$|\Delta p(n)| = \left| \lceil d - 0.5 \rceil - \lceil \hat{d} - 0.5 \rceil \right| . \quad (19)$$

We use the MPEG test sequences: *Kendo*, *Balloons*, *Newspaper*, and *Lovebird1* in our experiment. The method proposed in [12] is used to compare with our proposed methods. Depth maps were encoded in intra mode using CABAC entropy coding with QPs: 22, 27, 32, 37 and texture videos were not encoded. The reconstructed depth maps and texture videos (texture videos were not encoded) were used as input to view synthesis that was performed with view synthesis reference software (VSRS) version 3.5.

To verify that our proposed distortion model in (13) is more accuracy than (14), we first do some experiments use (13) without disparity rounding which we refer to as 'Method 1', namely, we use (8) to compute $|\Delta p(n)|$. Then, we use (13) with disparity rounding which we refer to as 'Method 2' to confirm the importance of disparity rounding and the efficiency of the whole proposed method for depth coding. We use the same analytical method as [12] to verify the accuracy of the proposed distortion estimation methods. We compute the real distortion in the rendered view, distortion in the compressed depth map, and the estimation distortion by 'Kim's' [12], 'Method 1' and 'Method 2' for all MBs in a row, respectively. The results for *Kendo* sequence are showed in Fig.1. Fig.1 shows all three methods closely follow the actual distortion of the rendered view than the depth map distortion, i.e., they provide more accuracy distortion estimation than using the distortion of the depth map, and the proposed 'Method 2' most closely follows the actual distortion occurred in the rendered view. Our proposed 'Method 1' gives almost the same results as [12].

We provide R-D curves for all four sequences in Fig. 2 in which 'H.264/AVC' represents an anchor, i.e., using original SSD distortion metric for depth coding, 'Kim's' represents the scheme of previous distortion metric [12], 'Method 1' represents our proposed method without disparity rounding, and 'Method 2' represents our proposed method with disparity rounding. In order to see the detailed



(a) Kendo (b) Balloons

(c) Newspaper (d) Lovebird1

Fig.2. Comparison of R-D curve for Kim's [12], the proposed method without disparity rounding (Method 1) and the proposed method with disparity rounding (Method 2).

performance of each method, we show its detailed PSNR improvement and bitrate saving values in Tables I and Table II. Here, we compute the BD-bitrate and BD-PSNR between the rendered view using compressed depth sequences and the rendered view using uncompressed depth sequences. As shown in our results, our proposed 'Method 1' is a little better than [12] which confirm that our proposed distortion model is more accuracy than [12]. Our proposed 'Method 2' makes the large gain when considering the disparity rounding. To sum up, the proposed scheme (Method 2) improves the coding performance up to 34.48% bit-rate saving or gain of 1.28 dB compared with conventional H.264/AVC in depth coding.

Need to explained, our experimental results of [12] is different from the results given in paper [12], because we use the synthesized view by the original color and depth reference views for PSNR computation, and the texture videos are not encoded in our experiments. This quality metric could reflect the effect of coding distortion of depth video on synthesized view more accuracy.

Table I. BD-PSNR results of Kim's [12] and the proposed methods comparing to H.264/AVC

Sequence	Kim's[12]	Method 1	Method 2
Kendo	0.41 dB	0.42 dB	1.47 dB
Balloons	0.23 dB	0.32 dB	1.31 dB
Newspaper	0.62 dB	0.67 dB	1.41 dB
Lovebird1	0.27 dB	0.27 dB	0.94 dB
average	0.38 dB	0.42 dB	1.28 dB

Table II. BD-bitrate results of Kim's [12] and the proposed methods comparing to H.264/AVC

Sequence	Kim's[12]	Method 1	Method 2
Kendo	12.81%	12.91%	39.72%
Balloons	7.32%	9.77%	35.41%
Newspaper	20.17%	20.83%	38.52%
Lovebird1	7.62%	7.57%	24.27%
average	11.98%	12.77%	34.48%

5. CONCLUSIONS

In this paper, we proposed a new distortion model which focuses on the distortion in rendered view and the disparity rounding problem for an optimized mode selection scheme in depth map coding. Compared to the previous work for rendering view distortion based depth maps coding, the proposed scheme greatly improves the coding efficiency. The experimental results show that our method can get the coding gain of 1.28 dB or 34.48% bitrate reduction on average compared with original H.264/AVC in depth coding.

6. REFERENCES

[1] Call for Proposals on 3D Video Coding Technology, Doc. N12036, ISO/IEC JTC1/SC29/WG11, Mar. 2011.

[2] Overview of 3D Video Coding, Doc.N9784, ISO/IEC JTC1/SC29/WG11, May. 2008.

[3] C. Fehn, "Depth-Image-Based Rendering (DIBR), Compression and Transmission for a New Approach on 3D-TV," *in Proc. of the SPIE Stereoscopic Displays and Virtual Reality Systems XI*, Jan. 2004.

[4] G. Shen, W. S. Kim, A. Ortega, and J. Lee, "Edge-aware intra prediction for depth-map coding," *in Proc. IEEE Int. Conf. Image Process*., pp. 3393–3396, 2010.

[5] P. Merkle, Y. Morvan, A. Smolic, K. Mûller, P. H. N. de With, and T. Wiegand, "The effect of depth compression on multiview rendering quality," *in Proc. 3DTV Conf.*, , pp. 245–248, May. 2008.

[6] M. Maitre and M. N. Do, "Joint Encoding of The Depth Image Based Representation Using Shape-Adaptive Wavelets," *in Proc. IEEE Int. Conf. Image Process.*, pp. 1768–1771, Oct. 2008.

[7] S. Liu, P. Lai, D. Tian, C. W. Chen, "New depth coding techniques with utilization of corresponding video," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp.551–561, 2011

[8] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Efficient compression of multi-view depth data based on MVC," *in Proc. IEEE 3DTV Conf.*, May 2007.

[9] J. Y. Lee, H. Wey, and D. S. Park, "A fast efficient multiview depth image coding method based on temporal and inter-view correlations of texture images," *IEEE Trans. Circuits Syst. for Video Technol.*, pp. 1–4, 2011.

[10] H. Oh and Y. S. Ho, "H.264-based depth map coding using motion information of corresponding texture video," *Adv. Image Video Technol.*, vol. 4319, 2006.

[11] W. S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map distortion analysis for view rendering and depth coding," *in Proc. IEEE Int. Conf. Image Process.*, 2009.

[12] W. S. Kim, A. Ortega, P. Lai, D. Tian, C. Gomila, "Depth Map Coding with Distortion Estimation of Rendered View," *in Proc. SPIE Vis. Inf. Process. Commun.*, 2010.

[13] Y. Zhao, C. Zhu, Z. Chen, L. Yu, "Depth no-synthesis-error model for view synthesis in 3D video," *IEEE Trans. Image Processing*, vol. 2, no. 8, pp.151–172, Aug. 2011.

[14] ISO/IEC JTC1/SC29/WG11, "View synthesis algorithm in view synthesis reference software 2.0 (VSRS2.0)," Doc. M16090, Feb. 2009.