ZERO-SYNTHESIS VIEW DIFFERENCE AWARE VIEW SYNTHESIS OPTIMIZATION FOR HEVC BASED 3D VIDEO COMPRESSION

Siwei Ma, Shiqi Wang, Wen Gao
Institute of Digital Media, Peking University, Beijing, 100871, China
{swma, sqwang, wgao}@pku.edu.cn

ABSTRACT

In this paper, we explore a new scheme for view synthesis optimization (VSO) in the upcoming high efficiency video coding (HEVC) based 3D video compression standard. With the dynamic range analysis of depth value, we present a novel zero-synthesis view difference (ZSVD) model. This model accounts for the compound impact of depth-disparity mapping, texture masking and occlusion information. We further incorporate the ZSVD profile into rate distortion optimization process via the ZSVD adaptive VSO, which is developed by pruning the conventional view rendering algorithm. Experimental results exhibit that the proposed scheme can remarkably reduce the computational complexity with negligible performance loss.

***Index Terms*—**Depth compression, view synthesis optimization, computational complexity, zero-synthesis view difference.

1. INTRODUCTION

The test model of the upcoming high efficiency video coding (HEVC) based 3D video coding standard [1] employs the multiview video plus depth format, which can provide large view-angle scene with fewer amounts of data by using the depth image-based rendering (DBIR) technique [2]. To fulfil the requirement of 3D representation and offer the audience totally immersive entertainment experience, at the encoder each view associate with its corresponding depth map are compressed and at the decoder the intermediate virtual views can be rendered by the neighboring texture and depth.

Depth map is a grey scale image/video which plays a key role in obtaining high quality virtual synthesis view. Since it has different characteristics compared with the texture video, many advanced coding tools are employed in the current test model to improve the performance of depth compression [1], such as Wedgelet and Contour modes for depth modeling, motion parameter inheritance and synthesis view inspired rate distortion optimization (RDO). The basic principle behind the synthesis view inspired RDO lies in that it is not the depth map but the synthesized view to be displayed in the 3DV system. Though it is generally true that better depth leads to better quality of synthesis video [3], the quality of depth map still cannot be linear mapped to the quality of the synthesis view. Therefore, the encoding techniques need to be optimized in accordance with the quality of the virtual synthesis view instead of the depth map.

Recently, in view of this room to further improve the performance of depth compression, many algorithms are proposed in the synthesis view inspired depth RDO. These algorithms can be mainly classified into two categories: rendering and non-rendering based. In the first category, the view rendering process is actually performed in the encoder for RDO. In [1], the view synthesis optimization (VSO) scheme is proposed by employing a measure called synthesis view distortion change (SVDC). To compute the rate distortion (RD) cost, the view rendering is actually performed iteratively in the encoding process. Though this scheme achieves high compression efficiency, it imposes a heavy computation burden to the encoder. In [4][5], simplified RDO schemes are proposed based on estimated distortion models. In these distortion models, the synthesized view by the original texture and depth need to be computed before coding the current depth map. However, they are based on the integer position grid in virtual view, which may not be accurate in general. The non-rendering RDO originates from the observation that the sensitivity of depth distortion is highly correlated with the properties of texture image [6]-[9]. Therefore, some simplified distortion models based on the texture information and depth distortion are proposed. However, because of the absence of actual view rendering process, these distortion models may not accurately capture the true synthesis distortion.

Typically, significant number of distorted depth pixels will not cause synthesis view distortion, even in low bit-rate cases. Inspired by this observation, we propose an early termination strategy of zero-synthesis view difference (ZSVD) for VSO under the test model described in [1]. Specifically, in this scheme, three properties of depth map are employed to derive the conditions of ZSVD. Firstly, it is generally recognized that depth distortions may not always cause geometry changes in the synthesized view [10], as the disparity range is usually much less than the quantized depth levels. Therefore, several depth levels may correspond to the same disparity in the DIBR-based 3DV system. Secondly, texture masking is a major consideration in ZSVD as homogeneous texture region will mask the depth distortion.

As such, distorted depth which lies in the smooth texture regions may not cause view synthesis distortion. Thirdly, in general, depth pixel which lies in the occlusion region is not as important as other pixels. Following the ZSVD estimation approach, we propose a ZSVD aware VSO scheme based on the determination of ZSVD for all the pixels in the row. Specifically, if all pixels in the current to be synthesized row satisfy the ZSVD condition, computations of view rendering of this row can be completely skipped.

The rest of this paper is organized as follows. In Section 2, the SVDC based view synthesis optimization approach is briefly reviewed. In Section 3, we analyze the dynamic range of depth value and propose three conditions to determine ZSVD. Based on the proposed ZSVD model, we develop the ZSVD aware VSO scheme in Section 4. Experimental results are given in Section 5 to verify the proposed scheme. Finally, we conclude our paper in Section 6.

2. SVDC BASED VIEW SYNTHESIS OPTIMIZATION

The SVDC based VSO technique is adopted for depth compression in the current test model [1]. The motivation behind this technique is that lossy coding of depth image causes the geometry distortion in the synthesized intermediate views; hence, the distortion measure for depth RDO should be the synthesis view distortion. In this section, we briefly review the SVDC based VSO technique from two aspects: distortion model and RDO process for depth compression.

2.1. Distortion model

Since there doesn't exist a bijective mapping between distortion of depth map and that of virtual synthesis view, to obtain an accurate distortion measure for RDO, the actual view synthesis is carried out at the encoder. The distortion model of VSO is then defined to be the SVDC as follows [1]:

$$\begin{aligned}\Delta D &= \tilde{D} - D \\ &= \sum_{(x,y)\in I}[\tilde{s}_T(x,y) - s'_{T,R}(x,y)]^2 - \\ &\quad \sum_{(x,y)\in I}[s'_T(x,y) - s'_{T,R}(x,y)]^2\end{aligned} \tag{1}$$

where $s'_{T,R}(x,y)$ denotes the virtual view rendered by the original texture and original depth. Both $\tilde{s}_T(x,y)$ and $s'_T(x,y)$ are synthesized by the reconstructed texture. Generally, in the encoding process, the depth map is composed of three components: encoded depth, current tested depth block and the other depth blocks. $s'_T(x,y)$ denotes the virtual view rendered by the reconstructed depth for already encoded blocks, original depth for the current test block and original depth for others. By contrast, $\tilde{s}_T(x,y)$ differs from $s'_T(x,y)$ in that $\tilde{s}_T(x,y)$ is rendered by the distorted depth for the current test block.

2.2 RDO process for depth compression

The RDO process with VSO in depth coding can be expressed by minimizing the synthesis view distortion D with the number of bits R subjected to a constraint R_c, which can be converted to an unconstrained problem by:

$$\min\{J\} \quad where\ J = D + \lambda \cdot R \tag{2}$$

where J is called the Rate Distortion (RD) cost and λ is known as the Lagrange multiplier. D is the distortion measure which is defined to be SVDC in the VSO.

Due to the quadtree coding structure of HEVC, for coding each largest coding unit (LCU), block based depth view synthesis will be recursively performed to compute SVDC. To feasibly apply the view synthesis in the RDO process, the current test model adopted a view synthesis model which combines the common functions of view rendering, such as warping, hole filling and blending, into one single algorithm [1]. Without loss of generality, assume two neighboring pixels to be x_p, x_c and the render is from right to left. That is to say, all the rendering steps for x_p should be finished before rendering x_c.

1) Interpolation

Interpolation is performed when the current pixel is not occluded or disoccluded. In the current test model, DCT based interpolation filter is employed to generate the quarter-accuracy version of the input texture. Assume the wrapped position of x_p, x_c to be x_p', x_c' and the full sample between x_p' and x_c' is x_{FP}. The position in the up-sampled texture which corresponds to x_{FP} is derived as follows:

$$\hat{x} = 4 \cdot \left(\frac{x_{FP} - x_p'}{x_c' - x_p'} + x_p\right) \tag{3}$$

2) Occlusion

Occlusion is detected by checking whether the internal boundary is reversed, for example, $x_p' > x_c'$. Moreover, the foreground edge is stored to determine whether next pixels are still occluded. Therefore, if occlusion is detected, no matter how great the depth pixel is distorted, provided that the pixel is still rendered into the occluded interval from the distorted depth data, the view synthesis process won't be affected.

3) Disocclusion

Disocclusion is detected by the width between x_p' and x_c' and if x_c'- x_p' is greater than two times of full-pixel sample intervals, the current interval is set to be disocclusion and thus hole filling is carried out. In general, if the current view is interpolated by left and right views, hole won't cause big distortions in the synthesis view as the other view will handle the occluded regions [7].

After the view synthesis process by interpolation, occlusion or disocclusion, the synthesis pixels from the left view and right view are blended together to generate the final virtual view. The whole process of the VSO is carried out by the following modules.

M1. Before coding the current depth map, the view synthesis is performed by the original texture and original depth, as well as the distorted texture and original depth.

M2. In each RD cost calculation, the encoder will re-render part of the synthesis virtual view with the distorted depth block to obtain $\tilde{s}_T(x,y)$ for SVDC calculation.

M3. When the encoding of a depth block is finished, the final reconstructed depth block will be utilized to update the virtual view $s_T^{'}(x,y)$.

In the encoding process, M2 and M3 are recursively performed to obtain accurate SVDC for VSO.

3. ZSVD FOR DEPTH

In general, the distortion of the synthesis view can be written as [7]:

$$\begin{aligned} &SSD_v \\ &= \sum_{(x',y')} |s'_{T,R}(x',y') - s'_{T,D}(x',y')|^2 \\ &= \sum_{(x,y)} | f(T(x,y), D(x,y)) - f(\tilde{T}(x,y), \tilde{D}(x,y))|^2 \end{aligned} \quad (4)$$

where $T(x,y)$ represents the original texture video and $D(x,y)$ represents the original depth map, respectively. f indicates the warping function which synthesize the virtual view $s'_{T,R}(x',y')$ by $T(x,y)$ and $D(x,y)$. Similarly, $\tilde{T}(x,y)$, $\tilde{D}(x,y)$ and $s'_{T,D}(x',y')$ indicate the reconstructed texture video, depth, and the synthesized view rendered by $\tilde{T}(x,y)$ and $\tilde{D}(x,y)$, respectively. From Equ. (4), the distortion can be represented as:

$$\begin{aligned} &SSD_v \\ &= \sum | f(T,D) - f(\tilde{T},\tilde{D})|^2 \\ &= \sum | f(\tilde{T} + (T-\tilde{T}), D) - f(\tilde{T},\tilde{D})|^2 \\ &\approx \sum | f(\tilde{T},D) - f(\tilde{T},\tilde{D})|^2 + e(T) \end{aligned} \quad (5)$$

where $e(T)$ denotes the texture coding distortion. It indicates that the distortion of the synthesis view caused by distortion of texture and depth can be approximated to be independent. For more details of this proof, please refer to [11], [12]. This conclusion also implies that if the distorted depth data has no influence on the view synthesis process, the SVDC calculation can be avoided as $\Delta D = 0$.

Assume the original and distorted depth value to be v and v'. The synthesized views with v and v' are denoted to be $s(v)$ and $s(v')$, respectively. Then the sufficient condition for ZSVD can be formulated as:

$$C_{ZSVD} = \{v' \mid s(v) = s(v')\} \quad (6)$$

Basically, there are three properties of depth map that account for this ZSVD condition, namely depth-disparity mapping, texture masking and occlusion. In this section, we will analyze these properties and deduce three corresponding conditions to derive the dynamic range of ZSVD for depth. It is also noted that in this work, we assume that the virtual view is placed on the horizontal axis of the original view, which is commonly known as the 1-D parallel arrangement.

3.1 Depth disparity mapping

Assume the true depth is z and the disparity is d, the relationship between z and d can be expressed as

$$d = \frac{f \cdot l}{z} \quad (7)$$

where f indicates the focal length of the camera and l represents the baseline between two views.

In practical, depth is represented by L-bit quantized values and the quantization process is given by

$$v = Q(z) \quad (8)$$

Thus, the relationship between d and v is derived as

$$d = g(v) = \frac{f \cdot l}{Q^{-1}(v)} \quad (9)$$

Generally, the disparity is rounded to $1/N$ sub-pixel position, as shown in Fig. 1. Let $R_N(d)$ denote the rounding process and (x_p, y_p) denote the position of the current pixel in the texture picture. d_{op} and d_{sp} represent the original and the distorted disparity at (x_p, y_p), respectively. Then the disparity distortion at p can be defined as

$$D_N(d_{op}, d_{sp}) = | R_N(d_{op}) - R_N(d_{sp}) | \quad (10)$$

Since the virtual view is synthesized using the rounded disparity, the first condition to guarantee ZSVD is derived as

$$C_1 = \{v' \mid D_N(g(v), g(v')) = 0\} \quad (11)$$

In the current implementation, as the interpolation is performed with quarter-accuracy, N is set to be 4.

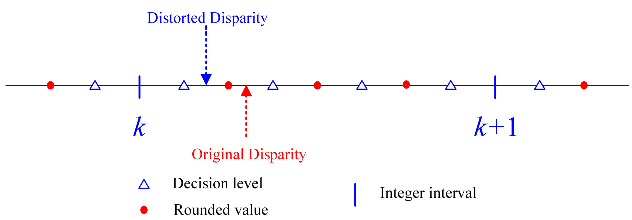

Fig. 1. Illustration of the quarter-accuracy rounding process of disparity.

3.2 Texture masking

In general, the same depth distortions in complex and homogeneous region will cause different synthesis distortion. Fig. 2 presents the synthesis view with the original and distorted depth, respectively. It is shown that the synthesis view distortion can be approximated to be the difference between the current pixel value and the value with position-shift d_{op}-d_{sp}. Therefore, the distortion in the synthesis view is approximated to be [13]:

$$\begin{aligned} D_V(p) &= (\tilde{T}(x_p, y_p) - \tilde{T}(x_p + d_{op} - d_{sp}, y_p))^2 \\ &\approx (\tilde{T}(x_p, y_p) - (\tilde{T}(x_p, y_p) + \nabla\tilde{T}(x_p, y_p) \cdot (d_{op} - d_{sp})))^2 \\ &= (\nabla\tilde{T}(x_p, y_p) \cdot (d_{op} - d_{sp}))^2 \\ s.t. \quad & |d_{sp} - d_{op}| < \sigma, \sigma \to 0 \end{aligned} \tag{12}$$

From this equation, it is observed that the distortion of the synthesis view can be affected by the texture gradient. The smooth texture has nearly no distortion while the high gradient region has significant synthesis distortion. Therefore, the second condition to determine ZSVD for the current depth is derived as

$$C_2 = \{v' \mid D_1(g(v), g(v')) = 0 \ and \ G(x_p, y_p) = 0\} \tag{13}$$

where D_1 is defined in Eq. (10) with N=1. $G(x_p, y_p)$ indicates the horizontal gradient at position (x_p, y_p) and is defined as follows:

$$G(x_p, y_p) = \sum_{i=-1,1} |\tilde{T}(x_p, y_p) - \tilde{T}(x_p + i, y_p)| \tag{14}$$

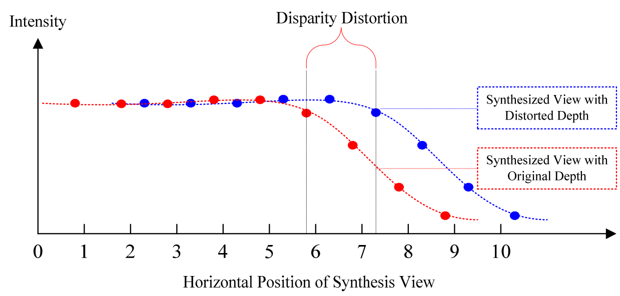

Fig. 2. Synthesized views with the original depth and the distorted depth.

3.3 Occlusion information

Generally, a pixel will be occluded if its neighbouring pixel has much larger disparity value, as shown in Fig. 3. The occluded pixel does not affect the rendering process as long as the distorted depth still locates within the occlusion interval. Thus, the occluded pixels are not as important as other pixels. Inspired by this observation, we give the third condition to determine ZSVD as

$$C_3 = \{v' \mid D_1(g(v), g(v')) = 0 \ and \ v \in O\} \tag{15}$$

Similarly, D_1 is defined in Eq. (10) with N=1. O denotes the set of depth pixels which corresponds to the occlusion regions.

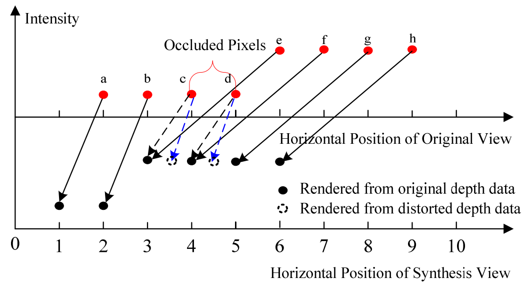

Fig. 3. Illustration of the occluded pixels during warping process.

Overall, the condition used for determining ZSVD is given by union of C_1, C_2 and C_3 as:

$$C = C_1 \cup C_2 \cup C_3 \tag{16}$$

It is noted that the sufficient derivations of C_2 and C_3 are based on the following strong assumptions:

1) When two integer pixels are identical, the sub-pixels lying in the interval between them are also equal to them.

2) If the warping position with the distorted depth lies in the same integer interval with that of the original depth, and the original pixel is occluded, then the distorted pixel is still occluded.

In practical, these two assumptions are fairly reasonable to derive the sufficient condition, as can be verified in Section 5.2. Moreover, it is generally believed that more sufficient and tighter conditions require more complex computational models, which will lead to more additional computations. Therefore, to achieve a good balance between the computational complexity and accuracy, we employ the proposed model to simplify the VSO process.

4. ZSVD AWARE VSO

In this section, we propose a ZSVD aware VSO approach as an attempt to achieve fast depth compression. The ZSVD model is incorporated into M2 and M3 of the VSO process, as described in Section 2. The purpose of M2 and M3 is to update the current synthesized view with the changed depth block by actually view rendering. In this work, we aim to prune the view rendering process with the help of the ZSVD estimator. Based on the assumption of the 1-D parallel arrangement, we can draw the conclusion that the view rendering for each row is independent. By contrast, within each row, the distorted depth pixel may affect the synthesis result of the undistorted depth pixel. Therefore, the basic unit for the ZSVD aware VSO is defined to be a row in the current depth block.

The whole process of the proposed scheme is summarized in Fig. 4. For a to be rendered MxN depth block in M2 or M3, we check each row by the derived condition C. Assume the pixel set of the j-th row in current block is S_j. If all pixels in the current row satisfy condition C, view rendering of this row can be completely skipped.

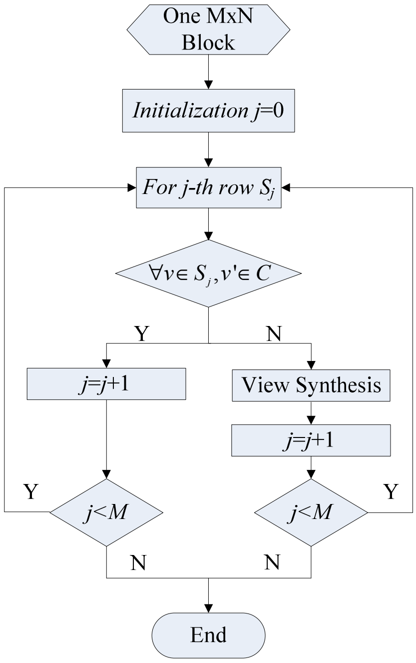

Fig. 4. Flowchart of the proposed ZSVD aware VSO.

5. VALIDATIONS

To validate the accuracy and efficiency of the proposed ZSVD aware VSO, we integrate our scheme into the 3D-HTM reference software (version 0.4). Specifically, the quantization parameters (QP) for texture are from 25 to 40, and the corresponding QPs for depth are from 34 to 45.

5.1 Performance Comparison

Firstly, we evaluate the performance of the scheme by the RD performance and the coding complexity. Specifically, the RD performance is evaluated by the BD-rate and the time reduction is evaluated as follows:

$$\Delta T = \frac{T_{pro} - T_{org}}{T_{org}} \times 100\% \tag{17}$$

where T_{org} and T_{pro} indicate the total coding time of both texture and depth with the original 3D-HTM and the proposed scheme, respectively.

Table 1. R-D performance of the proposed scheme.

Sequence	3 View			2 View		
	ΔR_v	ΔR_s	$\Delta R_{v,s}$	ΔR_v	ΔR_s	$\Delta R_{v,s}$
Balloons	0.0%	-0.2%	-0.1%	0.0%	-0.1%	-0.1%
Kendo	0.0%	-0.1%	0.0%	0.0%	-0.1%	-0.1%
Newspaper	0.0%	0.1%	0.1%	0.0%	-0.4%	-0.2%
GhostTownFly	0.0%	-0.1%	-0.1%	0.0%	0.0%	0.0%
PoznanHall2	0.0%	0.3%	0.2%	0.0%	0.3%	0.2%
PoznanStreet	0.0%	0.0%	0.0%	0.0%	0.1%	0.0%
UndoDancer	0.0%	0.1%	0.1%	0.0%	0.1%	0.1%
Average	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%

Table 1 tabulates the RD performance of the proposed scheme for 2 view and 3 view cases, where the 2 view test case is extracted from the 3 view data. ΔR_v, ΔR_s and $\Delta R_{v,s}$ indicate the bit rate reduction in terms of the original video, the synthesis view, and both the original and synthesis view, respectively. The experimental results show that on average the proposed scheme does not bring any performance loss. The RD performance comparisons of the proposed scheme to the original VSO for sequence *PoznanHall2* are shown in Fig. 5, and it is seen that, the performance variation with the proposed scheme over the full range of QP is ignorable.

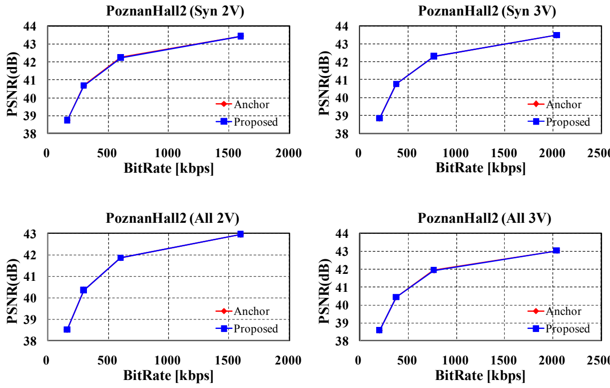

Fig. 5. Coding performance comparisons of the proposed scheme to the original VSO.

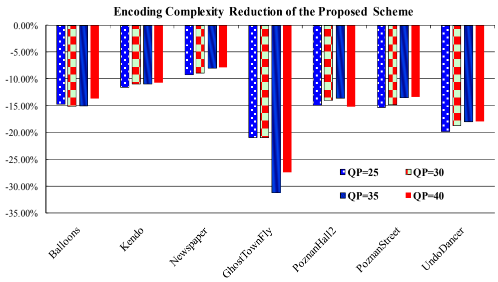

Fig. 6. Encoding complexity reduction of the proposed scheme.

To verify the robustness and efficiency of the proposed scheme, the encoding complexity reduction computed by Equ. (17) is shown in Fig. 6. From this figure, it is observed that the proposed algorithm can achieve 8% to 30% of total encoding time reduction depending on texture QP value ranging from 25 to 40, and on average about 15% of complexity reduction is achieved. It indicates that our scheme can efficiently reduce the encoding complexity in terms of the view synthesis computations.

5. 2 Analysis of hit rate and false alarm

To further verify the effective of the proposed ZSVD aware VSO, we measured the hit rate and false alarm by comparing the forecast result of the proposed ZSVD model with the actually view rendering approach. The first 100 frames of each sequence are tested. The hit rate which measures the fraction of true ZSVD that are correctly forecasted by the proposed model is shown in Table 2. It is observed that on average 69% of the true ZSVD can be forecasted. The missed forecasts mainly originate from the texture and occlusion masking. For example, the wider the current smooth region is, the more distortion of depth can be allowed without generating synthesis view artifact. However, exactly calculating the range of depth variation for ZSVD needs more overhead computations. The false alarm of the proposed scheme is tabulated in Table 3, which indicates that the fraction of the forecast ZSVD that were not true is no more than 1%. This proves that the proposed model can provide an accurate determination of ZSVD for the VSO.

Table 2. Hit rate of the proposed ZSVD model to the actual view rendering.

Sequence	QP=25	QP=30	QP=35	QP=40
Balloons	77.41%	74.50%	72.69%	70.65%
Kendo	72.50%	67.32%	66.38%	63.04%
Newspaper	70.17%	63.71%	59.52%	58.99%
GhostTownFly	76.28%	70.81%	70.99%	69.06%
PoznanHall2	69.71%	61.04%	59.12%	56.85%
PoznanStreet	73.98%	69.24%	66.44%	64.11%
UndoDancer	84.35%	79.73%	75.28%	66.10%
Average	74.91%	69.48%	67.2%	64.11%

Table 3. False alarm of the proposed ZSVD model to the actual view rendering.

Sequence	QP=25	QP=30	QP=35	QP=40
Balloons	0.39%	0.36%	0.34%	0.31%
Kendo	0.45%	0.49%	0.43%	0.39%
Newspaper	0.61%	0.62%	0.61%	0.51%
GhostTownFly	0.14%	0.22%	0.28%	0.42%
PoznanHall2	0.17%	0.22%	0.26%	0.30%
PoznanStreet	0.16%	0.14%	0.16%	0.22%
UndoDancer	0.15%	0.18%	0.23%	0.33%
Average	0.30%	0.32%	0.33%	0.35%

6. CONCLUSION

In this paper, we propose a ZSVD aware VSO scheme for fast depth compression. The novelty of this scheme lies in defining a novel ZSVD model and incorporating the ZSVD model in the VSO process. Specifically, three properties of depth map are employed in the ZSVD modeling. The proposed model provides precise conditions for the early determination of ZSVD. We further present the application of ZSVD in the adaptive VSO process, in which the view rendering algorithm is pruned by the ZSVD estimator. The proposed scheme demonstrates superior performance as compared to state-of-the-art HEVC 3DV codec by offering significant complexity reduction, while keeping the same level of RD performance.

7. ACKNOWLEDGEMENT

This research is supported by the 973 program (2009CB 320903), the 863 program (2012AA011505) and the National Science Foundation (61121002, 61103088) of China, which are gratefully acknowledged.

8. REFERENCES

[1] H. Schwarz, K. Wegner, “Test Model under Consideration for HEVC based 3D video coding, ISO/IEC JTC1/SC29/WG11 MPEG, Doc. M12350, Nov. 2011, Geneva, Switzerland.

[2] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, “Multiview video plus depth representation and coding,” in Proc. IEEE Int. Conf. Image Process., 2007, pp. 201–204.

[3] P. Merkle, Y. Morvan, A. Smolic, and D. Farin, “The effects of multiview depth video compression on multiview rendering,” Signal Process.: Image Commun., vol. 24, no. 1-2, pp. 73–88, Jan. 2009.

[4] D. V. S. X. De Silva, W. A. C. Fernando and S. T. Worrall, “Intra Mode Selection Method for Depth Maps of 3D Video based on Rendering Distortion Modeling,” Consumer Electronics, IEEE Transactions on, vol. 56, Issue 4, pp. 2735-2740, 2010.

[5] J. Jung, E. Son, S. Yea, “3D-HEVC-CE10 results on modified depth distortion measure by LG”, ISO/IEC JTC1/SC29/WG11 MPEG, Doc. M23856, Feb. 2012.

[6] W. Kim, A. Ortega, P. Lai, D. Tian, C. Gomila, “Depth map distortion analysis for view rendering and depth coding”, in Proc. IEEE Int. Conf. Image Process., 2009, pp. 721–724.

[7] B. Oh, J. Lee and D. Park, “Depth Map Coding Based on Synthesized View Distortion Function”, IEEE journal of selected topics in signal processing, Vol. 5, No. 7, Nov. 2011.

[8] 3D-HEVC-CE10 results on Joint RDO for depth coding of 3D video by ZJU, ISO/IEC JTC1/SC29 WG11, MPEG, m23829, Feb. 2012, San Jose.

[9] B. Oh, J. Lee and D. Park, “3D-HEVC-CE10 results on Samsung’s view synthesis optimization using distortion in synthesized views”, ISO/IEC JTC1/SC29 WG11, MPEG, m23668, Feb. 2012, San Jose.

[10] Y. Zhao, C. Zhu, Z. Chen and L. Yu, “Depth No-Synthesis-Error Model for View Synthesis in 3-D Video”, IEEE Trans. on Image Processing Vol. 20, No. 8, Aug. 2011.

[11] H. Yuan, Y. Chang, F. Yang, and Z. Lu, “Model-based joint bit allocation between texture videos and depth maps for 3-D video coding,”IEEE Trans. Circuits Syst. Video Technol., vol. 21, no. 4, pp. 485–497, Apr. 2011.

[12] Q. Wang, X. Ji, Q. Dai, N. Zhang, “Free Viewpoint Video Coding with Rate-Distortion Analysis”, accepted, IEEE Trans. Circuits Syst. Video Technol.

[13] C. Chen, Y. Chen, F. Yang, W. Peng, “A Synthesis-Quality-Oriented Depth Refinement Scheme for MPEG Free Viewpoint Television (FTV)”, 2009, 11th IEEE International Symposium on Multimedia, pp. 171-178.