# Efficient Multiview Depth Coding Optimization Based on Allowable Depth Distortion in View Synthesis

Yun Zhang, *Member, IEEE*, Sam Kwong, *Fellow, IEEE*, Sudeng Hu, and Chung-Chieh Jay Kuo, *Fellow, IEEE*

***Abstract*— Depth video is used as the geometrical information of 3D world scenes in 3D view synthesis. Due to the mismatch between the number of depth levels and disparity levels in the view synthesis, the relationship between depth distortion and rendering position error can be modeled as a many-to-one mapping function, in which different depth distortion values might be projected to the same geometrical distortion in the synthesized virtual view image. Based on this property, we present an allowable depth distortion (ADD) model for 3D depth map coding. Then, an ADD-based rate-distortion model is proposed for mode decision and motion/disparity estimation modules aiming at minimizing view synthesis distortion at a given bit rate constraint. In addition, an ADD-based depth bit reduction algorithm is proposed to further reduce the depth bit rate while maintaining the qualities of the synthesized images. Experimental results in intra depth coding show that the proposed overall algorithm achieves Bjontegaard delta peak signal-to-noise ratio gains of 1.58 and 2.68 dB on average for half and integer-pixel rendering precisions, respectively. In addition, the proposed algorithms are also highly efficient for inter depth coding when evaluated with different metrics.**

***Index Terms*— 3D video, depth coding, view synthesis, depth no-synthesis-error, rate-distortion optimization, allowable depth distortion.**

## I. Introduction

THREE Dimensional Video (3DV) has been attracting more and more attention recently since it is able to provide immersive vision, real 3D depth perception and new visual enjoyments for new type of multimedia applications, such as Three Dimensional TeleVision (3DTV) and Free-viewpoint TeleVision (FTV). Multiview depth video is one of the most important data of 3DV [1], which provides the geometrical information of the 3D world scene and allows arbitrary view rendering with high quality and low complexity [2]. To lower the complexity of the video clients, the depth videos are generated, encoded at server and transmitted to the clients instead of being generated with the multiview video at the clients [1]. However, the depth videos possess large data volume and their amounts increase with the number of views. So in addition to efficient color encoder, high efficiency depth encoding algorithm is also highly desired to reduce the requirements on the storage space and transmission bandwidth.

To tackle this problem, Multiview Video Coding (MVC) [3] and its coding optimizations [4], [5] can be extensively used to encode the depth video while regarding the depth video as illumination component of color video. However, the depth videos have different correlations and properties from the illumination component of the color videos. For example, the depth maps are generally smooth; they may have noise and temporal inconsistency raised by depth estimation. On the other hand, the depth videos are used as the geometrical information in view rendering for 3D video system, thus, depth coding distortion will be projected to be the geometrical errors inside or among video objects. For example, ring artifacts of depth video may lead to edge corruptions, while block artifacts may introduce false contours. Due to the differences between color and depth videos, traditional MVC algorithms and tools for color video coding are not effective enough to be applied directly to depth map coding for achieving good quality of synthesized videos.

Currently, the Joint Collaborative Team on 3D video coding extension development (JCT-3V), has been established to develop more advanced 3DV coding technology [6]. In addition, many researchers have devoted their efforts to the depth coding and a number of techniques were proposed. Since the qualities of the virtual view images are sensitive to the coding distortions in the boundaries of depth video, boundary reconstruction filter [7] and trilateral filter [8] were utilized to preserve the sharp depth boundaries. Nguyen *et al.* [9] presented weighted mode filtering in order to reduce the coding artifacts at the edges in reduced resolution depth coding. Since some depth maps are generated by the depth estimation algorithms, they may have noise and temporal inconsistency,

Manuscript received November 28, 2013; revised April 28, 2014; accepted August 20, 2014. Date of publication September 8, 2014; date of current version October 2, 2014. This work was supported in part by the National Natural Science Foundation of China under Grant 61102088, Grant 61272289, and Grant 61471348, in part by the Shenzhen Emerging Industries through the Strategic Basic Research Project under Grant JCYJ20120617151719115, and in part by the Natural Science Foundation of Guangdong Province under Grant S2012010008457. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Béatrice Pesquet-Popescu.

Y. Zhang is with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, and also with the Department of Computer Science, City University of Hong Kong, Hong Kong (e-mail: yun.zhang@siat.ac.cn).

S. Kwong is with the Department of Computer Science, City University of Hong Kong, Hong Kong, and also with the City University of Hong Kong Shenzhen Research Institute, Shenzhen 5180057, China (e-mail: cssamk@cityu.edu.hk).

S. Hu and C.-C. J. Kuo are with the Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: sudenghu@usc.edu; cckuo@sipi.usc.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2014.2355715

which leads to low depth coding efficiency. Thus, spatial and temporal smoothing filters [10] were proposed to reduce high frequency prediction residues and improve the compression efficiency. Zhu *et al.* [11] proposed a filtering scheme for the depth encoder aiming at compensating the view synthesis distortion by additionally transmitting the filter coefficients. In addition, based on the depth visual sensitivity in human 3D visual system, Silva *et al.* [12] proposed depth adaptive preprocessing filter so as to suppress the depth details within Just Notice Difference in Depth (JNDD) [13] and thus improved the depth compression ratio. Considering the depth videos usually contain less texture, down and up sampling algorithms [14] were developed for reduced resolution depth coding. As the depth maps are used for view synthesis, Zhao *et al*. [15] proposed a Depth NO-Synthesis-Error (D-NOSE) model to smooth the depth map, and thereby improved INTRA depth coding efficiency. However, the D-NOSE has not been included in the coding process and spatial redundancies were not fully exploited. On the other hand, it could hardly guarantee both filtering and quantization errors are within the D-NOSE range, especially in the case of using large Quantization Parameter (QP). These algorithms can be considered as pre-/post- refinement for the depth video.

In addition, many depth coding algorithms [16]–[20] have also been extensively studied. Pan *et al*. [16] exploited the mode correlation between depth and its corresponding color video to reduce the mode candidates for fast depth coding. In [17], motion information, including motion vector, mode, and reference frame index, were shared between the color and depth videos to reduce the depth coding bits. In [18], view synthesis prediction was adopted in depth coding in order to improve the depth inter-view prediction accuracy. Since the depth maps are used for the view synthesis in 3D video system, some depth coding algorithms [19], [20] were proposed with the target of maximizing the virtual image quality. Lee *et al.* [19] proposed a depth coding scheme which directly copied collocated block pixels and transmitted a flag instead when the corresponding color differences were sufficiently small. In [20], Kang *et al.* presented an INTRA prediction algorithm which refined INTRA prediction modes at depth edges and thereby preserved boundary for higher view synthesis quality.

As view synthesis image quality is finally adopted in measuring the depth map quality, the distortion term of the Rate-Distortion (RD) cost function in the depth coding shall consider the view synthesis distortion. To estimate the view synthesis distortion, linear model [21], [22] and power model [23] were adopted to model the relation between depth distortion and view synthesis distortion. Chung *et al.* [24] estimated the view synthesis distortion from the frequency domain. In addition, structural similarity was introduced [25] to measure the synthesized view distortion for better perceptual quality. These schemes can be regarded as the global estimation schemes which were applied to the entire depth image. In [26], each depth map was divided into two kinds of regions according to their characteristics and effects to the view synthesis. Then, linear model with different slopes were derived to guide the regional selective depth coding. As the statistical dependencies between the depth error and the distortion of synthesized virtual view changes along frames, Yuan *et al.* [27] derived a polynomial model and proposed a depth map coding by modifying the Lagrangian multiplier at frame level in order to maximize the quality of the synthesized images. For low complexity purpose, these models attempt to estimate the view synthesis distortion by not involving the view synthesis process. In [27], partial re-rendering was performed to obtain more accurate view synthesis distortion.

Since the quality of the synthesized virtual view images is not only affected by the quality of depth images, but also affected by the quality of color images, depth and color coding processes could be jointly optimized under the total bit rate constraint. In [29], joint bit allocation algorithms among depth and color channels were presented in order to maximize the quality of rendered virtual view images under the total bit constraints. Hu *et al*. [30] presented a joint rate control scheme with the target of maximizing the summative qualities of both the original color videos and the rendered images. In [31], it was revealed that the depth distortion and color distortion were additive to the view synthesis distortion in global aspect, while Oh *et al.* [32] found the local depth distortion had a joint effect with the corresponding color spatial texture while mapping to the view synthesis distortion.

In our previous work [26], we have exploited the regional selective properties of a depth map and divided it into two kinds of regions, named Color Texture Area corresponding Depth (CTAD) and Color Smooth Area corresponding Depth (CSAD). Since the depth distortion in CSAD has less impact on the quality of synthesized image than that in CTAD region, optimal QPs and Lagrangian multipliers were specifically adjusted to give higher priority to CTAD and lower priority to CSAD in allocating depth bits and using the coding techniques. Therefore, [26] is a regional selective coding algorithm which encodes the CSAD and CTAD differently. However, in this paper, considering that the number of depth levels is usually larger than the number of disparity levels in the view synthesis, several different depth values are projected to one disparity, i.e. a many-to-one mapping function. It indicates some small distortions in depth map will not lead to any rendering position error, i.e. no-synthesis-error [15]. In addition, it also allows multiple different depth distortions projecting to one non-zero rendering position error. These depth redundancies are called Allowable Depth Distortion (ADD) in view synthesis and they exist at each depth value and pixel. In this work, we exploit these ADD redundancies by designing a new depth distortion criterion with piecewise function and optimize the RD cost function for mode decision and Motion Estimation/Disparity Estimation (ME/DE) at macroblock (MB) level. Furthermore, the depth coding bits are reduced by exploiting the ADD.

The paper is organized as follows, in Section II, a many-to-one function for the ADD in view synthesis is derived to model the relationship between the view synthesis distortion and the depth distortion. In Section III, the ADD model is presented for the coding optimization. Then, two depth coding techniques, the Rate Distortion Optimization (RDO) and Depth Bit Reduction (DBR) algorithms, are proposed

based on the ADD model. The performances of the proposed algorithms with different settings are comparatively evaluated and analyzed in Section IV. Finally, Section V draws the conclusions.

## II. Allowable Depth Distortion (ADD) in Depth Image Based Rendering

In [15], Zhao *et al*. proposed the D-NOSE model to exploit the allowable distortion in the depth for the case that no rendering position error in view synthesis would be caused, i.e. no-synthesis-error case. However, another allowable distortion in depth that makes multiple different depth distortions project to one non-zero rendering position error was not considered in [15]. In this section, we analyze the ADD redundancies in view synthesis, which includes the both above two cases. In Depth Image Based Rendering (DIBR), the pixels of virtual view image can be rendered from the pixels of its view-neighboring reference images with the depth and camera parameters by [2]

$$p_2 = z_1 \mathbf{A}_2 \mathbf{R}_2 \mathbf{R}_1^{-1} \mathbf{A}_1^{-1} p_1 - \mathbf{A}_2 \mathbf{R}_2 \mathbf{R}_1^{-1} \mathbf{t}_1 + \mathbf{A}_2 \mathbf{t}_2, \tag{1}$$

where $p_2 = [a, b, c]^T$ and $p_1 = [x, y, 1]^T$ are the two corresponding pixels in rendered and real view images, respectively. $z_1$ is the depth for $p_1$; $\mathbf{A}_1$ and $\mathbf{A}_2$ are two $3 \times 3$ matrices indicating camera intrinsic parameters for the virtual and real cameras. $[\mathbf{R}_1, \mathbf{t}_1]$ and $[\mathbf{R}_2, \mathbf{t}_2]$ are the extrinsic parameters for the two cameras, where $\mathbf{R}_1$ and $\mathbf{R}_2$ are the rotation matrices, $\mathbf{t}_1$ and $\mathbf{t}_2$ are the translation factors. The disparity offsets between $p_1$ and $p_2$ in horizontal and vertical directions, $(d_x, d_y)$, can be calculated as

$$\begin{cases} d_x = \Phi\left(x - \dfrac{a}{c}\right) \\ d_y = \Phi\left(y - \dfrac{b}{c}\right), \end{cases} \tag{2}$$

where $\Phi()$ is a rounding operation. This rounding operation function relies on the disparity accuracy, i.e. the rendering precision of the view synthesis. This function can be mathematically expressed as

$$\Phi(x) = \frac{\lfloor x \cdot 2^m + k_f \rfloor}{2^m}, \tag{3}$$

where "$\lfloor\ \rfloor$" indicates a floor operation; $k_f$ is the compensation factor to round down and up decimal fractions, which is $2^{m-1}$; $m$ is the rendering precision, which is 0, 1 or 2 for integer, half or quarter-pixel precision, respectively.

Suppose the virtual and real cameras are parallel to each other, well calibrated and have the same intrinsic parameters, i.e. $\mathbf{A}_1 = \mathbf{A}_2$, $\mathbf{R}_1 = \mathbf{R}_2$, $\mathbf{t}_1 - \mathbf{t}_2 = [L, 0, 0]^T$, where $L$ is the interval of the camera array in the baseline. The vertical rendering disparity $d_y$ is zero and the horizontal rendering disparity $d_x$ is

$$d_x = \Phi\left(\frac{f_x L}{Z}\right), \tag{4}$$

where $f_x$ is the horizontal focal length and $Z$ is physical depth. For the depth map in MPEG-3DV, a non-linear quantization scheme is adopted to convert the physical depth $Z$ into

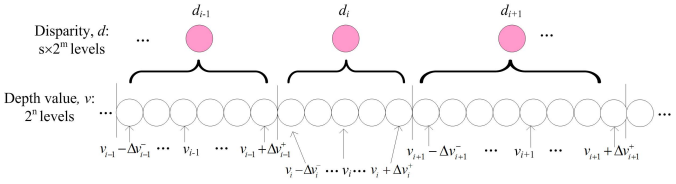


Fig. 1. Example of mapping depth value to disparity.

$n$-bit depth value ranges from 0 to $2^n$-1 [1], where $n$ is the bit width representing the depth value. The inverse quantization from depth value $v$ to depth Z is [1]

$$Z = Q^{-1}(v) = \frac{1}{\frac{v}{2^n}\left(\frac{1}{Z_{near}} - \frac{1}{Z_{far}}\right) + \frac{1}{Z_{far}}}, \tag{5}$$

where $Z_{near}$ and $Z_{far}$ are the distance from the camera to the nearest and furthest depth planes of a video scene, respectively. Applying (5) into (4), we can have

$$d_x = \Phi(L f_x (C_1 v + C_2)), \tag{6}$$

where $C_1 = \frac{1}{2^n}\left(\frac{1}{Z_{near}} - \frac{1}{Z_{far}}\right)$, $C_2 = \frac{1}{Z_{far}}$. In a 3DV system, depth video will be encoded and its bit stream is transmitted to the clients for arbitrary virtual view rendering.

In the lossy depth coding process, coding distortion will be introduced by the quantization. Suppose a depth coding distortion $\Delta v$ is introduced to the depth value $v$ by the encoder, and the distorted depth value $v + \Delta v$ obtained from decoding is then used in view synthesis at client's machine. Comparing with the view rendering of using the original depth $v$, the disparity difference $\Delta d_x$ can be calculated by

$$\begin{aligned} \Delta d_x = g(v, \Delta v) = {} & \Phi(L f_x (C_1 v + C_2)) \\ & - \Phi(L f_x (C_1 (v + \Delta v) + C_2)) \end{aligned} \tag{7}$$

Due to the rounding operation $\Phi()$, $\Delta d_x$ may not be changed sometimes even when $\Delta v$ changes, i.e. several different $\Delta v$s are mapped to one $\Delta d_x$. It indicates the relation between $\Delta d_x$ and $\Delta v$ is a many-to-one mapping.

Let $s$ be the dynamic disparity levels (number of levels from the maximum to the minimum disparity) between the rendered image and the reference image. Therefore, the number of disparity levels at a given rendering precision $m$ is $s \times 2^m$. The number of the depth levels (i.e. the number of levels of $v$) is $2^n$. We define a mapping coefficient $C_{\mathrm{MM}}$ as

$$C_{MM} = \frac{2^n}{s \times 2^m}, \tag{8}$$

where $C_{\mathrm{MM}}$ indicates the average number of depth levels corresponding to the same disparity level. According to the requirement of the 3D video, $n$ is usually 8 in the latest 3D depth representation [1]. The number of disparity levels ($s$) is usually less than 30, and most of them are smaller than 10. In this case, when we use the 1/2 pixel rendering process (i.e. $m = 1$), $C_{\mathrm{MM}}$ ranges from 4 to 12, which implies 4 to 12 different $\Delta v$s are mapped to one disparity $\Delta d_x$. As shown in Fig.1, instead of one single $v_i$, multiple points $v_i + \Delta v_i$, $\Delta v_i \in \left[-\Delta v_i^-, \Delta v_i^+\right]$, are mapped to the same disparity $d_i$. Here, $\Delta v_i$ is the $\Delta v$ for depth value $v_i$, $-\Delta v_i^-$ and $\Delta v_i^+$ are defined as the lower and upper bounds for $\Delta v_i$.

Consequently, two important features can be derived from this many-to-one mapping process. First, if both the original depth $v_i$ and distorted depth value $v_i + \Delta v_i$ are within the ADD range $\left[v_i - \Delta v_i^-, v_i + \Delta v_i^+\right]$, i.e. $\Delta v_i \in \left[-\Delta v_i^-, \Delta v_i^+\right]$, the disparity difference $\Delta d_x$ in (7) will equal to zero, i.e. no synthesis error. Second, if the depth distortion $\Delta v_k$ causes a non-zero disparity difference $\Delta d_x$, there is an allowable distortion range around the $\Delta v_k$, $\Delta v_i \in \left[\Delta v_k - \Delta v_i^-, \Delta v_k + \Delta v_i^+\right]$, leading to the same $\Delta d_x$. The depth distortion that meets the above requirements will not change the $\Delta d_x$ and thus it is allowable in view synthesis. Thereby, we call it allowable depth distortion in view synthesis, and denote it as 'ADD'.

For the upper bound $\Delta v_i^+$, it is the rightmost point of $\Delta v_i$ that is mapped to $d_i$ and its right neighboring point $\Delta v_i^+ + 1$ is mapped to $d_{i+1}$. Therefore, given a depth value $v_i$, its upper bound value $\Delta v_i^+$ can be obtained when it satisfies

$$\begin{cases} g\left(v_i, \Delta v_i^+ + 1\right) = \dfrac{1}{2^m} \\ g\left(v_i, \Delta v_i^+\right) = 0 \\ v_i + \Delta v_i^+ \in \left[0, 2^n - 1\right]. \end{cases}$$

Similarly, the lower bound value $\Delta v_i^-$ is obtained when it satisfies

$$\begin{cases} g\left(v_i, -\Delta v_i^- - 1\right) = -\dfrac{1}{2^m} \\ g\left(v_i, -\Delta v_i^-\right) = 0 \\ v_i - \Delta v_i^- \in \left[0, 2^n - 1\right]. \end{cases}$$

Actually, the lower bound value of $v_i (v_i - \Delta v_i^-)$ is equal to the upper bound of $v_{i-1} (v_{i+1} + \Delta v_{i-1}^+)$ plus 1, i.e. $v_i - \Delta v_i^- = v_{i-1} + \Delta v_{i-1}^+ + 1$. The number of $\Delta v_i$s mapping to one $\Delta d_x$ is defined as ADD depth interval $W_{DI}$, which equals to $\Delta v_i^+ + \Delta v_i^- + 1$. For the parallel camera setting, $W_{DI}$ is the $\Delta v_i$ when $L f_x C_1 \Delta v_i$ is approaching $\frac{1}{2^m}$ [15], which is

$$W_{DI} = \Delta v_i : L f_x C_1 \Delta v_i \to \frac{1}{2^m} \tag{9}$$

where "$x \to x_0$" indicates that the variable $x$ is ultimately approaching a constant $x_0$. Therefore, $W_{DI}$ is an integer $\Delta v_i$ that is generally independent to $v_i$ and it can be presented as

$$W_{DI} = \left\lfloor \frac{1}{2^m L f_x C_1} - \zeta \right\rfloor + 1, \tag{10}$$

where $\zeta$ is a positive constant approaching to 0, and it may make $W_{DI}$ slightly different for different depth values ($v_i$). Equation (10) indicates that the depth interval $W_{DI}$ is dependent on camera positions, and it decreases as the view synthesis precision ($m$), focal length ($f_x$) and camera baseline ($L$) increase. Additionally, it also changes as the nearest or farthest depth planes change. Fig.2 shows the depth interval $W_{\mathrm{DI}}$ for 3DV sequences when integer-pixel rendering precision ($m = 0$) is used in the view synthesis. The test sequences' information can be referred in Section IV. We observe that the depth interval $W_{DI}$ is sequence dependent. It ranges from 6 to 25 and most of them are larger than 10, which imply redundancies exist in the depth.

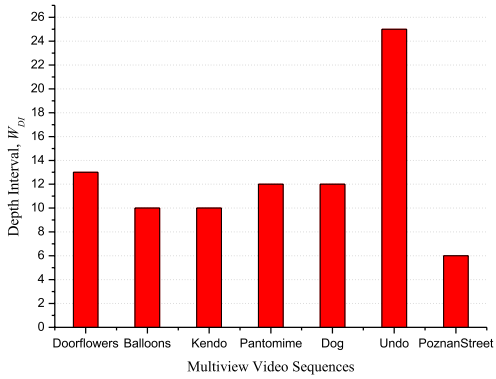


Fig. 2. ADD for different 3D video sequences.

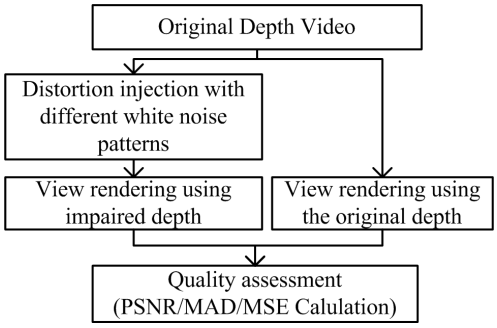


Fig. 3. Flowchart of analyzing depth effect in view synthesis.

## III. ADD Model and Depth Coding Optimizations

### A. Proposed ADD Model for Depth Coding

Since the quantization error in video coding resulting from uniform scalar quantization can be modeled by White Noise (WN) model [33], zero mean WN noise is added to the depth map to analyze the ADD and its impact on the view synthesis. The flowchart of analyzing depth effect in the view synthesis is shown in Fig.3. In this paper, we assume the color videos and depth videos are separately encoded, and optimize the depth coding while the color video is either original or already encoded. The color video can be regarded as unchanged before and after the depth coding/processing. Thus, the WN is only added into the depth videos and the color videos are the same in the two rendering processes. There are three different kinds of WN injection patterns. The first type is global WN injection where white noise with different variances is added to the entire depth map; the second type is global WN injection with ADD control, where the magnitudes of WN are clipped within the ADD range for each pixel, i.e. $\Delta v_i \in \left[-\Delta v_i^-, \Delta v_i^+\right]$; the third type is that the above two independent WNs are sequentially injected.

Fig.4 shows the relationship between depth distortion ($D_d$) and view synthesis distortion ($D_{VS}$) with and/or without ADD control. The $x$-axis is the average Mean Squared Error (MSE) of the depth images and the $y$-axis is the MSE of synthesized virtual view images. The "NoADD" indicates $D_{VS}$&$D_d$ relation when zero mean WN with different variances are injected without control. For "ADD_0" to "ADD_15", zero mean WN

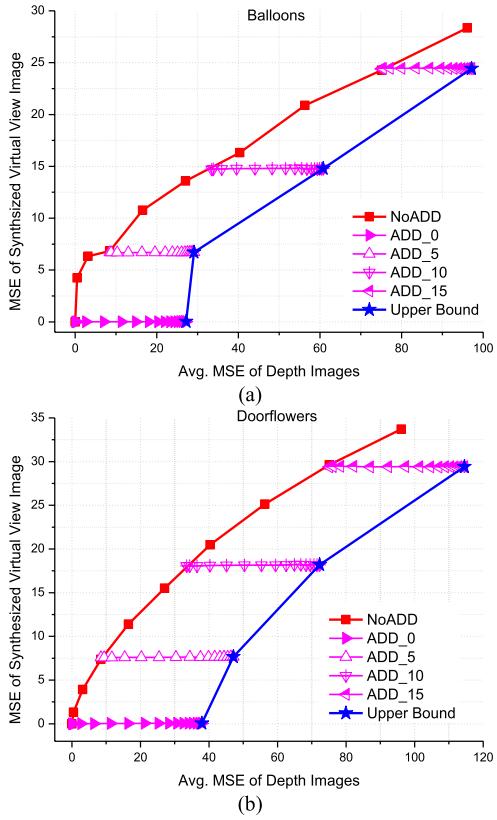


Fig. 4. Relationship between $D_{\mathrm{d}}$ and $D_{\mathrm{VS}}$ with and/or without ADD control. (a) Balloons, (b) Doorflowers.

with variances 0, 5, 10 or 15 are initially added to the depth to simulate the depth coding distortion caused by quantization. Then, another independent WN with different variances are added under ADD control, where all the distortion is clipped within $\left[-\Delta v_i^-, \Delta v_i^+\right]$. We connect the right most points of ADD_0 to ADD_15, and formulate the blue curve with start symbol. This blue line is the upper bound since no more distortion ($x$-axis) can be further added in the depth while maintaining the same virtual view image quality ($y$-axis).

It can be observed from Fig.4 that 1). $D_{VS}$ is monotonically increasing with $D_d$ for NoADD curves. It is generally linear or logarithmic between $D_{VS}$ and $D_d$. The increasing slopes are different over sequences and they usually increase as the texture of the videos getting more complex. 2). The blue upper bound curve is almost parallel with the red NoADD curve. 3). Curves "ADD_0" to "ADD_15" are almost horizontal and the $D_{VS}$ is almost the same as $D_d$ changes, which means the added depth distortion has little effect on $D_{VS}$ while under the ADD control. Though the depth distortion with ADD control does not cause error for the 3D warping, the depth dependent hole filling and merging after the rendering process will cause some differences in the synthesized virtual view [15]. However, this effect is small in the simulation and the slopes of ADD_0 to ADD_15 approximately equals to zero. 4). ADD_0 to ADD_15 are the four parallel lines between the NoADD and the upper bound. Actually, when we seamlessly change the variance for the first round WN injection, $D_{VS}\&D_d$ points can cover the whole region in the

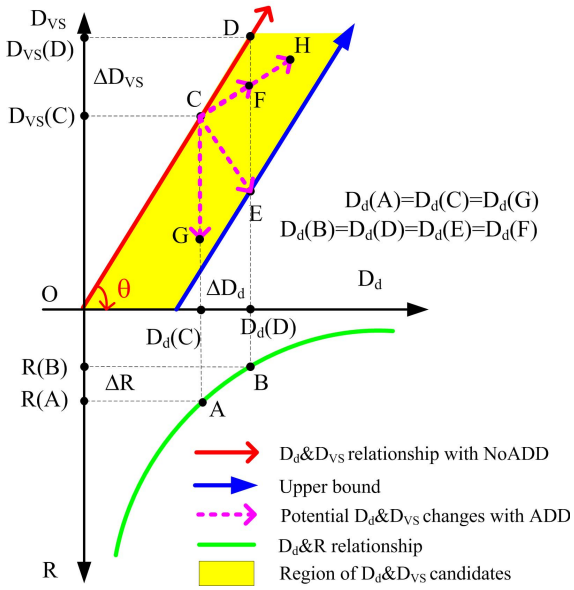


Fig. 5. RD analysis on the ADD based depth coding.

area between the NoADD and the upper bound. Similar results can also be found when the distortion is measured with Mean Absolute Difference (MAD).

Based on the above observations of $D_{VS}\&D_d$ relation, we derive a new model for depth bit rate ($R$) and view synthesis distortion ($D_{VS}$), i.e. $R\&D_{VS}$ model, by combining $D_{VS}\&D_d$ and $R\&D_d$ relations. Fig.5 shows a sketch map for the ADD based RD model, where the $x$-axis is the depth distortion ($D_d$), the positive $y$-axis is the view synthesis distortion ($D_{VS}$), and the negative $y$-axis is the depth bit rate ($R$). The first quadrant is the $D_{VS}\&D_d$ relationship and the fourth (bottom) quadrant shows the $R\&D_d$ relationship. In the first quadrant, the red line indicates the linear $D_{VS}\&D_d$ relationship without ADD control, which is deduced from the 'NoADD' curve in Fig.4 by linear approximation. The red line is also regarded as the lower bound for the $D_{VS}\&D_d$ relation. If the $D_{VS}\&D_d$ relation is located on the red curve, it indicates that the performance is almost the same as the original encoder. If the $D_{VS}\&D_d$ is located in the left region of the red line, the depth coding performance may degrade. The angle between the red line and $x$-axis is denoted by $\theta$, and the slope value of the red line is $\tan\theta$. The blue line is parallel to the red line and it is the upper bound of $D_{VS}\&D_d$ with ADD control, which is derived from the blue upper bound curve in Fig.4 by linear approximation. Compared with the red line, the $D_{VS}\&D_d$ point on the blue line has smaller view synthesis distortion while maintaining the same bit rate or depth distortion. Or, it has smaller bit cost while maintaining the same view synthesis distortion. The yellow region between the red and blue lines is the region of $D_{VS}\&D_d$ candidates, which could be achieved by a certain coding or processing scheme.

For better understanding, we define set {$R$(K), $D_d$(K), $D_{VS}$(K)} to measure the RD performance of an algorithm at point K $\in$ {A, B, C, D, E, F, G} in Fig.5, where $R$(K) is the depth coding bit rate, $D_d$(K) is the depth distortion and element $D_{VS}$(K) is the view synthesis distortion. When the

depth video is encoded with a traditional depth coding algorithm, such as the original Joint Multiview Video Coding (JMVC), we suppose it has depth bit rate $R$(A) and the corresponding depth distortion is $D_d$(A), here $D_d$(A) = $D_d$(C). Then the depth distortion $D_d$(C) will be mapped to $D_{VS}$(C) with a linear $D_{VS}$&$D_d$ relationship while ADD control is disabled. In other words, we have a set {$R$(A), $D_d$(C), $D_{VS}$(C)} to indicate the RD performance of the traditional depth coding algorithm. Similarly, when the depth video is encoded with the same depth encoder but with different coding parameters, e.g. larger $QP$s, we have another set {$R$(B), $D_d$(D), $D_{VS}$(D)}, where $D_d$(D) = $D_d$(B). Therefore, from the RD set {$R$(A), $D_d$(C), $D_{VS}$(C)} to the set {$R$(B), $D_d$(D), $D_{VS}$(D)}, the bit rate saving $\Delta R = R$(A)-$R$(B) is achieved at the cost of depth quality degradation, $\Delta D_d = D_d$(D)-$D_d$(C), and view synthesis image quality degradation, $\Delta D_{VS} = D_{VS}$(D)-$D_{VS}$(C). Basically, the coding performances of sets {$R$(A), $D_d$(C), $D_{VS}$(C)} and {$R$(B), $D_d$(D), $D_{VS}$(D)} are for one depth encoder.

Depth video is used to synthesize virtual view images. Thus, the view synthesis distortion will be considered in the depth coding optimization. To improve the depth coding efficiency with the ADD model, we can properly allocate the depth distortion based on the ADD information, and move the distortion from C to the inner yellow region, such as the point F. We can see that the view synthesis distortion difference $D_{VS}$(F)−$D_{VS}$(C) is smaller than $D_{VS}$(D)−$D_{VS}$(C), which means view synthesis distortion is reduced while maintaining the same bit rate reduction $\Delta R = R$(A)−$R$(B). It means F has better RD performance than C and D. In fact, the point C may change to E, F, G or H due to different optimization techniques, and E, F, G or H are all better than C in terms of $R$&$D_{VS}$ performance. If the RD moves from C to G, it means the ADD based optimization can reduce the view synthesis distortion from $D_{VS}$(C) to $D_{VS}$(G) with the same bit rate $R$(A). If RD moves from C to E, it means both the view synthesis distortion and bit rate can be reduced. We also find that the smaller the distance is between the end point (G, E, F or H) and the blue line, the better the RD performance is. The blue line is the upper bound of the RD performance for ADD based optimization techniques.

The maximum potential Peak Signal-to-Noise-Ratio (PSNR) gain at the same bit rate can be calculated when the $D_{VS}$&$D_d$ point locates at the upper bound, which is

$$\Delta PSNR_{VS,\max} = 10\log_{10}\frac{D_{VS}+\Delta D_{VS,\max}}{D_{VS}}, \quad (11)$$

where $D_{VS}$ is the view synthesis distortion of the original coding scheme, and $\Delta D_{VS,max}$ is the additional view synthesis distortion reduction achieved by a new scheme. Based on the $D_{VS}$&$D_d$ relationship of NoADD, we get the maximum view synthesis distortion difference as $\Delta D_{VS,max} = \Delta D_{d,max}\tan\theta$, where $\Delta D_{d,max}$ is MSE of $\Delta v_{ij}$, and $\Delta v_{ij}$ is an additional quantization error at position ($i$, $j$), $\Delta v_{ij} \in \left[-\Delta v_i^-, \Delta v_i^+\right]$. In the new scheme, the maximum PSNR is achieved when $\Delta v_{ij}$ equals to its the maximum or minimum values, $\Delta v_i^+$ and $-\Delta v_i^-$. Suppose $\Delta v_i^- = \Delta v_i^+$, they approximate to half of the depth interval, i.e. $W_{DI}/2$. Then, $\Delta D_{d,max}$ is the MSE of $\Delta v_{ij}$ when $\Delta v_{ij}$ is ± $W_{DI}/2$, thus, $\Delta D_{d,max}$ is obtained as $W_{DI}^2/4$. Consequently, applying it to (11), the maximum potential PSNR gain is

$$\Delta PSNR_{VS,\max} = 10\log_{10}\frac{D_{VS}+\left(W_{DI}^2/4\right)\tan\theta}{D_{VS}}. \quad (12)$$

Taking the Doorflowers sequence as an example, the $W_{DI}$ is 12; from the "NoADD" in Fig.4b, $\tan\theta$ is approximately equal to 0.35 in terms of the ratio of its $y$-axis dynamic range to its $x$-axis dynamic range; $D_{VS}$ is 6.5 when the original PSNR of synthesized image is 40dB. Thereby, we get the potential gain $\Delta PSNR_{VS,max}$ is up to 4.68dB, which is a large coding gain.

In color video coding, the objective is minimizing the distortion under the bit constraint. However, in depth coding, the depth is used for virtual view synthesis. The objective is minimizing view synthesis distortion $D_{VS}$ under depth bit rate constraint, or, minimizing the depth coding bit rate while maintaining the same view synthesis image quality. In other words, the optimization RD model for depth coding is not $R$&$D_d$ model but $R$&$D_{VS}$ model. In the following subsections, we firstly build the mathematical relation between $D_{VS}$ and $D_d$ with ADD redundancies. Then, we present a new ADD based RD model to minimize $D_{VS}$ in the variable block size mode decision, multi-reference frame selection as well as ME/DE. Finally, we present an ADD based DBR scheme, which further reduces the depth bit rate while maintaining $D_{VS}$.

### *B. ADD Based Distortion Model*

The depth distortion ($D_d$) leads to the rendering position error ($D_r$) in the view synthesis, and then this $D_r$ leads to the view synthesis distortion ($D_{VS}$). Therefore, to analyze the $D_{VS}$&$D_d$ relation and the ADD in view synthesis, we divide the analyses into two sub-steps, which are analyzing $D_{VS}$&$D_r$ and $D_r$&$D_d$ relationship.

When the uncompressed depth maps are used in view synthesis, the virtual view image $\mathbf{I}_{V,Dorg}$ is projected from the pixels of reference color image $\mathbf{I}_T$ with disparity $\mathbf{d}$, i.e. $\mathbf{I}_{V,Dorg} = \mathbf{I}_T(\mathbf{d})$, where $\mathbf{d}$ is a 2D disparity map in terms of $\mathbf{I}_T$ and $\mathbf{I}_{V,Dorg}$. The disparity map $\mathbf{d}$ can be expressed as {$(d_x(i, j), d_y(i, j))|i \in [0, M); j \in [0, N)$}, where $d_x(i, j)$ and $d_y(i, j)$ are horizontal and vertical disparity at ($i$, $j$), $M$ and $N$ are the width and height of the image $\mathbf{I}_T$, respectively. However, when distorted depth maps are used in view synthesis, the virtual view image $\mathbf{I}_{V,Drec}$ is also projected from $\mathbf{I}_T$ but with different disparity $\mathbf{d}+\Delta\mathbf{r}$. This $\Delta\mathbf{r} = \{(\Delta r_{ij}^x, \Delta r_{ij}^y)|i \in [0, M); j \in [0, N)\}$is a 2D rendering position error caused by the depth distortions, where $\Delta r_{ij}^x$ and $\Delta r_{ij}^y$ are the horizontal and vertical rendering position errors at position ($i$, $j$). Thus, the new virtual view image is $\mathbf{I}_{V,Drec} = \mathbf{I}_T(\mathbf{d} + \Delta\mathbf{r})$. Consequently, the difference map between synthesized virtual view images $\mathbf{D}_V$ can be calculated as

$$\mathbf{D}_V = \mathbf{I}_{V,Dorg} - \mathbf{I}_{V,Drec} = \mathbf{I}_T(\mathbf{d}) - \mathbf{I}_T(\mathbf{d}+\Delta\mathbf{r}). \quad (13)$$

It means the synthesized image difference $\mathbf{D}_V$ caused by depth distortion can be presented as the difference among neighboring pixels in the reference color image $\mathbf{I}_T$ [31]. Therefore, the

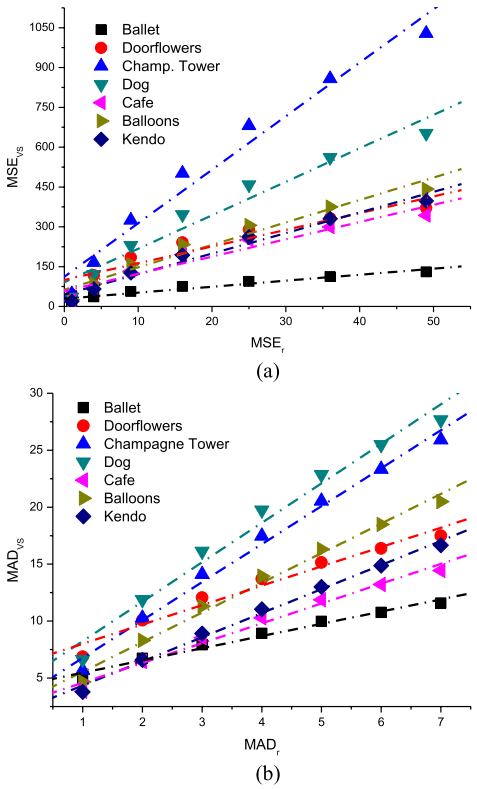


Fig. 6. Statistical $D_{\mathrm{VS}}$&$D_{\mathrm{r}}$ relationship in terms of MSE and MAD. (a) MSE. (b) MAD.

average view synthesis distortion $D_{VS}$ is computed by

$$D_{VS} = \frac{1}{MN}\sum_{i,j}\left|I_T(i,j) - I_T\left(i+\Delta r_{ij}^{x}, j+\Delta r_{ij}^{y}\right)\right|^{\beta}, \quad (14)$$

where $I_T(i, j)$ is the pixel value in image $\mathbf{I}_T$ with position $(i, j)$, $\beta$ is 1 for MAD and 2 for MSE.

For parallel camera settings, the disparity is mainly either horizontal or vertical, i.e. one of $\Delta r_{ij}^{x}$ and $\Delta r_{ij}^{y}$ element is approximately equal to zero. Thus, to analyze this relation between $D_{VS}$ and rendering position error in (14), each $\left(\Delta r_{ij}^{x}, \Delta r_{ij}^{y}\right)$ was randomly set as one of the four sets, $\{(\Delta r_{ij}, 0), (-\Delta r_{ij}, 0), (0, \Delta r_{ij}), (0,-\Delta r_{ij})\}$, to calculate the $D_{VS}$, where $\Delta r_{ij} \in \{1, 2, 3, 4, 5, 6, 7\}$. Seven different 3DV sequences were tested. Fig.6 plots the relationship between $D_{VS}$&$D_r$, where the $x$-axis is MSE or MAD of $\Delta r_{ij}$, i.e. $D_r$, the $y$-axis is view synthesis distortion measured with MSE or MAD, i.e. $D_{VS}$. The points with different symbols are real collected data and dash dot lines are the linear fitting results of the collected data. We used the correlation coefficient to indicate the goodness of fitting and the fitting is better when it is closer to 1. In Fig.6, the average correlation coefficients are 0.973 and 0.992 for the linear fittings of the MSE and MAD values, respectively, which indicates real data and fitting results are highly correlated. Thus, we can conclude that it has a linear relationship between $D_{VS}$ and $D_r$. Therefore, the $D_{VS}$ can be modeled as [26]

$$D_{VS} = K_1 D_r + K_2, \quad (15)$$

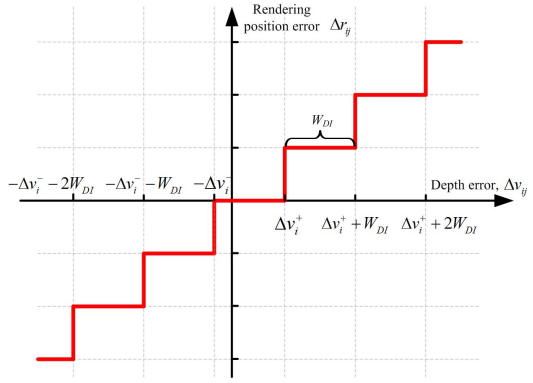


Fig. 7. Piecewise relation between depth distortion and rendering position error.

where $D_r = \frac{1}{MN}\sum\sum\left|\Delta r_{ij}\right|^{\beta}$ is MSE or MAD of the rendering position error $\Delta r_{ij}$. $K_1$ and $K_2$ are constants, $K_1$ is correlated with color texture and usually increases as the texture gets complex. Derived from (15), this linear relationship is also true when $D_{VS}$ and $D_r$ are measured with either Sum of Absolute Difference (SAD) or Sum of Squared Difference (SSD).

Based on the ADD analyses in Section II and subsection III.A, an example of relationship between the rendering position error and depth distortion is illustrated in Fig.7. The $x$-axis is depth distortion at position $(i, j)$ $(\Delta v_{ij})$ and $y$-axis is rendering position error at position $(i, j)$ in the rendered images $(\Delta r_{ij})$. It is a many-to-one mapping while projecting depth distortion $\Delta v_{ij}$ to the rendering position error $\Delta r_{ij}$. This $\Delta r_{ij}$ and $\Delta v_{ij}$ relation map might not be symmetric with the origin of the coordinate. Mathematically, $\Delta r_{ij}$ can be presented as

$$\Delta r_{ij} = \begin{cases} \left\lfloor \frac{\Delta v_{ij} - \Delta v_{ij}^{+}}{W_{DI}} \right\rfloor + 1 & \Delta v_{ij} > \Delta v_{ij}^{+} \\ 0 & -\Delta v_{ij}^{-} \le \Delta v_{ij} \le \Delta v_{ij}^{+} \\ \left\lceil \frac{\Delta v_{ij} + \Delta v_{ij}^{-}}{W_{DI}} \right\rceil - 1 & \Delta v_{ij} < -\Delta v_{ij}^{-}, \end{cases} \quad (16)$$

where "$\lceil\ \rceil$" is a ceil operation, $\Delta v_{ij}$ is the difference between the original depth value $v_{ij}$ and the reconstructed depth value $\tilde{v}_{ij}$ at position $(i, j)$. If $\Delta v_{ij}^{+}$ and $\Delta v_{ij}^{-}$ are zero, $W_{DI}$ equals to 1 and $\Delta r_{\mathrm{ij}}$ equals to $\Delta v_{\mathrm{ij}}$. In this case, the ADD based distortion metric is just the same as the traditional distortion metric. According to the definitions, the depth distortion $D_d$ and rendering position error $D_r$ are the MAD or MSE of $\Delta v_{ij}$ and $\Delta r_{ij}$, Therefore, the $D_{VS}$&$D_d$ relationship can implicitly be revealed by combining (15) and (16).

### C. ADD Based RD Model for Mode Decision and ME/DE

The RD model in H.264/AVC based video codec can be presented as [26], [34]

$$R(D) = k\ln\left(\sigma^2/D\right), \quad (17)$$

where $D$ is output distortion and $\sigma^2$ is variance of an input picture, $k$ is a constant. Taking the derivative of $R(D)$ with

respect to $D$ and setting its value to $-1/\lambda_{MODE}$ yields [26]

$$dR(D)/dD \equiv -1/\lambda. \tag{18}$$

Substituting (17) into (18), we get the optimal Lagrangian multiplier as [26]

$$\lambda = D/k. \tag{19}$$

As the depth video can be treated as Y component of color and coded by traditional hybrid H.264/AVC based coding standard, this model in (17) is also applicable to the depth coding. Thus the RD model for depth coding is

$$R_d(D_d) = k_d \ln\left(\sigma_d^2/D_d\right), \tag{20}$$

where $D_d$ is output distortion; $\sigma_d^2$ is the variance of an input depth; $k_d$ is a constant. However, since the reconstructed depth video is used for virtual view rendering, the view synthesis distortion ($D_{VS}$) will be taken into account in the new RD model. On the other hand, compressed depth bit rate $R_d$ is actually transmitted. To calculate the new Lagrangian factor ($\lambda^{VS}$) for view synthesis oriented video coding, we take the derivative of $R_d$ with respect to $D_{VS}$ and set its value to $-1/\lambda^{VS}$. Thus,

$$\frac{dR_d}{dD_{VS}} = \frac{dR_d(D_d)/dD_d}{dD_{VS}/dD_d} \equiv -\frac{1}{\lambda^{VS}}. \tag{21}$$

According to (16), the rendering position error $\Delta r_{ij}$ can be rewritten as

$$\Delta r_{ij} = \frac{1}{W_{DI}}\Delta v_{ij} + \varepsilon_{ij}, \tag{22}$$

where $\varepsilon_{ij}$ is a zero mean uniform distributed rounding error. As indicated by the Law of Large Number (LLN), the average value of all the samples approximates to their mathematical expectation when the number of samples is large. When the distortion is measured with MSE, the $D_r$ can be presented as

$$\begin{aligned} D_r \approx E\left(\Delta r^2\right) &= \left(\frac{1}{W_{DI}}\right)^2 E\left(\Delta v^2\right) \\ &+2\frac{1}{W_{DI}}E(\Delta v\varepsilon) + E\left(\varepsilon^2\right) \end{aligned} \tag{23}$$

where $E()$ is the mathematical expectation function. The depth distortion $\Delta v$ and the rounding error $\varepsilon$ can be regarded as independent variables in the coding process. Thereby, $E(\Delta v\varepsilon)$ is equal to $E(\Delta v)E(\varepsilon)$. The quantization error $\Delta v_{ij}$ and rounding error $\varepsilon_{ij}$ can be regarded as zero mean distributed [31], thus, $E(\Delta v\varepsilon) = 0$ since $E(\Delta v) = 0$ and $E(\varepsilon) = 0$. Therefore, (23) can be expressed as

$$D_r = \frac{1}{W_{DI}^2}D_d + E\left(\varepsilon^2\right), \tag{24}$$

where $D_d$ and $D_r$ are measured with MSE. Since $E(\varepsilon^2)$ is independent to $D_d$, its derivative $\partial E(\varepsilon^2)/\partial D_d = 0$. Therefore, for the mode decision, we apply (15) to (21) and (21) can be rewritten as

$$\frac{-k_d/D_d}{d(K_1D_r + K_2)/dD_d} = -\frac{1}{\lambda_{MODE}^{VS}}. \tag{25}$$

where $\lambda_{MODE}^{VS}$ is Lagrangian multiplier for the mode decision. Hence, applying (24) into (25), we obtain

$$\lambda_{MODE}^{VS} = K_1D_d/\left(k_dW_{DI}^2\right). \tag{26}$$

When the depth video is coded by the H.264/AVC based video codec, the $D$ and $k$ in (19) equal to $D_d$ and $k_d$. Then, applying (19) to (26), the Lagrangian multiplier for the mode decision is

$$\lambda_{MODE}^{VS} = K_1\lambda_{MODE}/W_{DI}^2. \tag{27}$$

The objective of depth coding is to minimize the view synthesis distortion at a given depth bit rate, i.e. minimizing $R\&D_{VS}$ cost function. Therefore, we need to minimize the ADD based Lagrangian cost function for mode decision as

$$\begin{aligned} \min J_{MODE}^{VS}, \quad J_{MODE}^{VS} &= SSD_{VS} + \lambda_{MODE}^{VS}R_{d,MODE} \\ &= K_1SSD_r + \left(K_1\lambda_{MODE}/W_{DI}^2\right) \\ &\quad \times R_{d,MODE} + MNK_2, \end{aligned} \tag{28}$$

where $K_1$ and $K_2$ are constants, $R_{d,MODE}$ is the bits of encoding mode and residue. Since $MNK_2$ and $K_1$ are constants, the optimization objective in (28) can be rewritten as

$$\min J_{MODE}^{VS}, \quad J_{MODE}^{VS} = W_{DI}^2SSD_r + \lambda_{MODE}R_{d,MODE} \tag{29}$$

where $SSD_r = \sum\sum\left|\Delta r_{ij}\right|^2$ and $\Delta r_{ij}$ is a piecewise function in (16). Equation (29) implies that if we apply ADD based mode decision to the traditionally video codec and encode the depth video with it, we need to replace the original distortion term with $W_{DI}^2SSD_r$, where $W_{DI}$ is calculated from camera parameters and baselines, etc., according to (10). On the other hand, the new distortion term is irrelevant to coefficients $K_1$ and $K_2$.

As for the ME/DE process, the second order distortion metric, MSE/SSD, is replaced by the first order distortion metric, such as MAD/SAD, for low complexity purpose. Therefore, we can deduce the new RD model for ME/DE in a similar way. Derived from (16), the absolute $\Delta r_{ij}$, i.e. $|\Delta r_{ij}|$, can be calculated as

$$\left|\Delta r_{ij}\right| = \frac{1}{W_{DI}}\left|\Delta v_{ij}\right| + \zeta_{ij}, \tag{30}$$

where $\zeta_{ij}$ is the rounding error satisfying uniform distribution with $W_{DI}/2$ mean. Similarly, when the depth distortion is measured with MAD, $D_r$ can be presented as the mathematical expectation of $|\Delta r_{ij}|$ according to the LLN, which is

$$D_r \approx E(|\Delta r|) = \frac{1}{W_{DI}}E(|\Delta v|) + E(\zeta), \tag{31}$$

where $E(|\Delta v|)$ is the MAD of distorted depth image, i.e. $D_d$, $E(\zeta) = W_{DI}/2$ is a constant and independent to the distortion $D_d$. Applying (31) to (21), we get

$$-\frac{kW_{DI}}{K_1D_d} = -\frac{1}{\lambda_{MOTION}^{VS}}. \tag{32}$$

By applying (19) to (32) as $D$ and $k$ are replaced by $D_d$ and $k_d$, the Lagrangian multiplier for ME/DE is

$$\lambda_{MOTION}^{VS} = K_1\lambda_{MOTION}/W_{DI}. \tag{33}$$

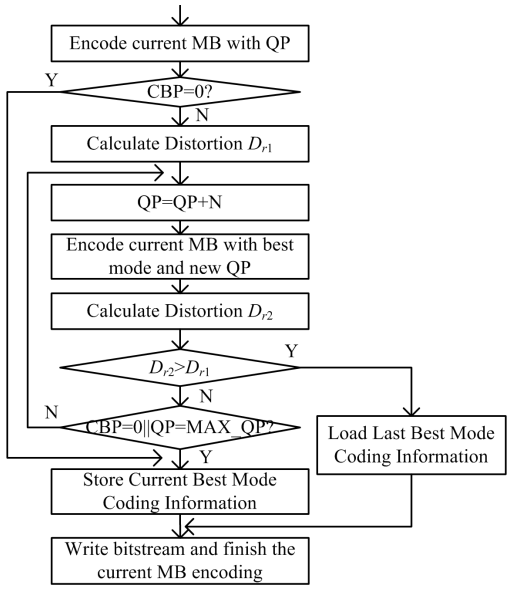


Fig. 8. Flowchart of the proposed ADD based DBR.

The new Lagrangian cost function for ME/DE and reference frame selection

$$\begin{aligned}\min J^{VS}_{MOTION},\ J^{VS}_{MOTION} &= SAD_{VS} + \lambda^{VS}_{MOTION} R_{d,MOTION} = K_1 SAD_r \\ &\quad + (K_1 \lambda_{MOTION}/W_{DI})\, R_{d,MOTION} + MNK_2 \end{aligned} \quad (34)$$

where $R_{d,MOTION}$ indicates the coding bits of motion/disparity vectors and reference frame indices. Therefore, the optimization target can be rewritten as

$$\begin{aligned}\min J^{VS}_{MOTION},\ J^{VS}_{MOTION} &= W_{DI} SAD_r + \lambda_{MOTION} R_{d,MOTION} \end{aligned} \quad (35)$$

where $SAD_r = \sum\sum |\Delta r_{ij}|$ and $\Delta r_{ij}$ can be referred from (16). It implies that we can replace the original distortion term with $W_{DI} SAD_r$ in the calculation of RD cost in the processes of the ME/DE and reference frame selection, etc.

### *D. ADD Based Depth Bit Reduction (DBR)*

In the above section, a new ADD model is utilized to minimize the view synthesis distortion for mode decision, reference frame selection, and motion/disparity estimation. Though the view synthesis distortion is minimized in these processes, the bit rate could also be reduced to further improve the RD performance. In depth coding, depth residues will be encoded and transmitted to compensate the differences between the predicted and original signals. These residues cost the depth bit rate. However, the depth video is used for virtual view rendering in 3D video system and the view synthesis distortion is finally measured. Some encoded residues reduce $D_d$ but not necessary reduce $D_{VS}$ due to the ADD in view synthesis. These residues cost depth bit rate but do not contribute to reducing $D_{VS}$. Thus, they should not be encoded and these coding bits could be saved consequently.

The flowchart of the proposed ADD based DBR is illustrated in Fig.8. Firstly, the current MB is encoded with initial $QP$ of the current slice. If the Coded Block Pattern (CBP) of the best mode equals to zero, no residual bits are encoded in the current MB. It means no more bits could be saved and the coding process for the current MB can be early terminated. Otherwise, we calculate the rendering position error (i.e. MSE of the $\Delta r$, $D_{r1}$) which is caused by the depth distortion of the current MB using (16). Then, increase the $QP$ by a step size of $N$, i.e. $QP = QP + N$, and re-encode the current MB with the new $QP$. We need to re-calculate the rendering position error $D_{r2}$ for the current MB. If the $D_{r2}$ is larger than $D_{r1}$, it implies that the view synthesis distortion increases when using the new $QP$. The previous $QP$ is the best one and we will load the previous best coding information and end this coding process. Otherwise, we will further increase $QP$ and re-encode the current MB until the residual coefficient are all-zero (i.e. CBP is zero) or $QP$ reaches the pre-defined maximum value, MAX_QP.

The ADD based DBR algorithm is a multiple pass coding which maximizes the $R\&D_{VS}$ performance for the non-zero coefficient MB. Meanwhile, the optimal MB mode and ME/DE vectors will be selected with different $QP$s. The incremental step $N$ indicates the fidelity of $QP$. As the $QP$ step size $N$ increases, it reaches the termination conditions more quickly, and thus, the coding complexity can be reduced. However, the coding efficiency may decrease as $N$ increases since the encoder might miss the optimal $QP$ when uses coarse fidelity. In this paper, $N$ is set as the minimum value 1 and fixed for all test sequences to maximize the $R\&D_{VS}$ performance. For the MB whose CBP is zero, its coding complexity is identically the same as the original JMVC. Based on the statistical analyses on the five different 3D depth sequences and different $QP$s, we found that the CBPs of 71% MBs on average in INTRA frames and 96% MBs on average in INTER frames are zero. It means only 29% INTRA MBs and 4% INTER MBs on average need the multiple pass coding in the proposed ADD based DBR scheme.

TABLE I
PROPERTIES OF THE TEST 3D VIDEO SEQUENCES

| Multiview video | Resolution | Frame rate(fps)/ Spacing (cm) | Coded views | Rendered views | Depth |
|---|---|---|---|---|---|
| Kendo | 1024×768 | 29.4/5 | 1-3-5 | 2-4 | A |
| Balloons | 1024×768 | 29.4/5 | 1-3-5 | 2-4 | A |
| Doorflowers | 1024×768 | 16.7/6.5 | 7-9-11 | 8-10 | N/A |
| Champ.Tower | 1280×960 | 30/5 | 36-38-40 | 37-39 | N/A |
| Dog | 1280×960 | 30/5 | 36-38-40 | 37-39 | N/A |
| Pantomime | 1280×960 | 30/5 | 36-38-40 | 37-39 | N/A |
| PoznanStreet | 1920×1088 | 25/13.75 | 3-4-5 | 3.5-4.5 | A |
| UndoDancer | 1920×1088 | 25/- | 1-2-3 | 1.5-2.5 | A |

## IV. EXPERIMENTAL RESULTS AND ANALYSES

To evaluate the coding efficiency of the proposed algorithms, MVC reference software JMVC 8.3 was used. Eight 3D video sequences, including Kendo, Balloons, Champ.Tower, Pantomime, Dog [35], Doorflowers [36], PoznanStreet [37] and UndoDancer [38], with various motion properties, resolutions and camera baselines were used. Their properties are shown in Table I. Three depth views

TABLE II
BDBR AND BDPSNR COMPARISONS FOR INTRA DEPTH CODING, VIRTUAL VIEW IMAGES WERE RENDERED WITH THE ORIGINAL COLOR IMAGES. (UNIT: %/DB)

| Precision | Test 3D Seq. | BDBR (%) | | | | | BDPSNR (dB) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ZhaoTIP [15] | ZhangTIP [26] | ADD_ DBR | ADD_ RDO | ADD_ RDO+DBR | ZhaoTIP [15] | ZhangTIP [26] | ADD_ DBR | ADD_ RDO | ADD_ RDO+DBR |
| Integer pixel | Balloons | -15.91 | -17.22 | -29.14 | -86.17 | -84.61 | 0.54 | 0.39 | 0.72 | 3.11 | 3.65 |
| | Kendo | -20.67 | -13.69 | -23.63 | NA | NA | 0.90 | 0.33 | 0.82 | 3.01 | 3.82 |
| | Doorflowers | -18.76 | -34.90 | -25.52 | NA | NA | 0.63 | 1.33 | 0.88 | 2.54 | 3.01 |
| | Dog | -18.01 | -14.45 | -18.33 | NA | NA | 0.56 | 0.50 | 0.56 | 1.35 | 1.73 |
| | Champ.tower | NA | -10.45 | -2.88 | -1.70 | -1.65 | -0.89 | 0.86 | 0.27 | 0.13 | 0.21 |
| | Pantomime | -27.92 | -15.46 | -23.21 | NA | NA | 0.56 | 0.54 | 0.46 | 1.28 | 1.87 |
| | PoznanStreet | 6.68 | -12.76 | -14.74 | -41.02 | -45.02 | -0.08 | 0.26 | 0.31 | 1.23 | 1.39 |
| | UndoDancer | 39.81 | -9.17 | -12.47 | -63.87 | -73.89 | -1.34 | 0.65 | 0.99 | 4.89 | 5.76 |
| | **Average** | **-7.83** | **-16.01** | **-18.74** | **-48.19** | **-51.29** | **0.11** | **0.61** | **0.63** | **2.19** | **2.68** |
| Half pixel | Balloons | 80.37 | -8.88 | -3.36 | -5.43 | -8.67 | -1.54 | 0.75 | 0.29 | 0.53 | 0.85 |
| | Kendo | NA | -1.47 | -3.45 | -5.37 | -6.06 | -1.97 | 0.14 | 0.45 | 0.71 | 0.82 |
| | Doorflowers | -3.56 | -24.31 | -9.15 | -9.86 | -14.27 | -0.25 | 1.71 | 0.52 | 0.64 | 0.99 |
| | Dog | -13.19 | -6.52 | -11.54 | -22.10 | -27.72 | 0.40 | 0.32 | 0.34 | 0.93 | 1.15 |
| | Champ.tower | 15.44 | NA | 2.36 | -5.65 | -12.70 | 0.29 | -0.05 | 0.70 | 1.08 | 1.39 |
| | Pantomime | -28.69 | -23.25 | -18.67 | -40.20 | -45.81 | 0.72 | 0.67 | 0.41 | 1.35 | 1.56 |
| | PoznanStreet | -2.38 | -5.22 | -8.43 | -15.84 | -18.74 | 0.04 | 0.12 | 0.20 | 0.52 | 0.63 |
| | UndoDancer | NA | -5.74 | -10.09 | -47.47 | -57.09 | -2.11 | 0.48 | 0.85 | 4.21 | 5.27 |
| | **Average** | **8.00** | **-10.77** | **-7.79** | **-18.99** | **-23.88** | **-0.55** | **0.52** | **0.47** | **1.25** | **1.58** |

were encoded in the three-view configuration of the JMVC codec. The two intermediate views synthesized by the original color and depth videos were used as a reference for the view synthesis quality comparison [32]. For example, the 1st, 3rd and 5th depth view were encoded as the 0, 1 and 2 view in the three-view configuration codec. The 2nd and 4th view are synthesized as virtual views for image quality evaluation. In these test sequences, depth videos of Kendo, Balloons, PoznanStreet and UndoDancer are available, the rest of depth sequences were generated by Depth Estimation Reference Software (DERS) [39]. View Synthesis Reference Software (VSRS) [40] was used for the view synthesis, where both integer and half-pixel rendering precision were tested. Averaging process was used for the view merging and hole-filling in the view synthesis which is the same as the settings in [15]. Basis *QP*s were set as 16, 20, 24 and 28. Six different coding schemes, the original JMVC, Zhao's scheme [15] denoted as 'ZhaoTIP', our previous scheme [26] denoted as 'ZhangTIP', the proposed ADD based RDO scheme (denoted by 'ADD_RDO'), proposed DBR scheme (denoted by 'ADD_DBR'), the proposed overall scheme which integrates ADD_RDO and ADD_DBR (denoted by 'ADD_RDO+DBR') were implemented for comparison.

Average PSNR of synthesized images was used for the depth image quality evaluation. The PSNR of synthesized image is calculated as

$$PSNR_{VS,\chi} = 10\log_{10}\frac{255^2}{MSE_{VS,\chi}} \tag{36}$$

$$MSE_{VS,\chi} = \frac{1}{MN}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\left|V_\chi(i,j) - V_O(i,j)\right|^2 \tag{37}$$

where $V_O(i,\ j)$ is the rendered image pixel value at $(i,\ j)$ generated by the original color video and the original depth video. $V_\chi(i,\ j)$ is the rendered image pixel value at $(i,\ j)$ generated by the original color video and reconstructed depth video, $\chi$ indicates depth coding scheme, $\chi \in$ {JMVC, ZhaoTIP [15], ZhangTIP [26], ADD_RDO, ADD_DBR, ADD_RDO+DBR}, $M$ and $N$ are the width and height of the rendered images, respectively.

When the integer-pixel precision setting was used in the view synthesis, ZhaoTIP scheme improves the coding efficiency for most sequences compared with the original JMVC. It has Bjontegaard Delta PSNR (BDPSNR) [41] gain of 0.11dB, which means 0.11dB improvement in PSNR under the same amount of bit consumption or 7.83% Bjontegaard Delta Bit Rate (BDBR) [41] which means reducing 7.83% of bit consumption with the same quality in PSNR, as shown in Table II. Note that in Tables II and III, some BDBR values are not available because the fitting algorithm is not applicable or the gap is too large. Thus, they are labeled as 'NA'. These unavailable data were not in the calculation of the average value. For high definition video sequences (e.g. PoznanStreet and UndoDancer), ZhaoTIP scheme is inferior to JMVC, which reveals the coding gain of ZhaoTIP is not stable and sequence content dependent. The main reason is that ZhaoTIP scheme is a pre-processing for depth video, which smoothes the depth video to reduce INTRA prediction residues and thus improves coding efficiency. Though the smoothing distortion can be controlled within ADD, it can hardly guarantee the total distortion (i.e. the quantization distortion plus the smoothing distortion) is still within the ADD depth interval $W_{\mathrm{DI}}$, especially in the case of using larger *QP*s. For ZhangTIP scheme, it exploits the regional selectivity of the depth redundancies. It improves the RD performance for all sequences and achieves 0.61 dB BDPSNR gain on average for integer-pixel accuracy setting. In terms of BDBR, it achieves 16.01% bit rate saving on average.

As for the proposed algorithms, the proposed ADD_DBR scheme not only reduces size of unnecessary depth coding

TABLE III
BDBR AND BDPSNR COMPARISONS FOR INTRA DEPTH CODING, WHERE VIRTUAL VIEW IMAGES WERE RENDERED WITH THE CODED COLOR IMAGES. (UNIT: %/DB)

| Precision | Test 3D Seq. | BDBR (%) | | | | | BDPSNR (dB) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ZhaoTIP [15] | ZhangTIP [26] | ADD_ DBR | ADD_ RDO | ADD_ RDO+DBR | ZhaoTIP [15] | ZhangTIP [26] | ADD_ DBR | ADD_ RDO | ADD_ RDO+DBR |
| Integer pixel | Balloons | -9.38 | -9.49 | -16.86 | -48.54 | -52.98 | 0.21 | 0.22 | 0.42 | 1.47 | 1.78 |
| | Kendo | -11.35 | -12.73 | -14.88 | -31.52 | -34.76 | 0.32 | 0.34 | 0.41 | 1.24 | 1.50 |
| | Doorflowers | -6.89 | -23.51 | -12.49 | -16.73 | -22.05 | 0.17 | 0.59 | 0.31 | 0.48 | 0.67 |
| | Dog | -2.28 | -3.25 | -4.42 | -12.84 | -14.88 | 0.06 | 0.08 | 0.11 | 0.35 | 0.40 |
| | Champ.tower | NA | -9.87 | -2.21 | -6.44 | -6.27 | -1.06 | 0.45 | 0.10 | 0.31 | 0.32 |
| | Pantomime | -7.28 | -7.12 | -6.50 | -21.78 | -26.30 | 0.17 | 0.17 | 0.15 | 0.54 | 0.71 |
| | PoznanStreet | 4.81 | -5.32 | -5.31 | -19.38 | -21.64 | -0.10 | 0.13 | 0.13 | 0.55 | 0.64 |
| | UndoDancer | 11.10 | -2.04 | -2.34 | -10.19 | -11.65 | -0.44 | 0.09 | 0.11 | 0.48 | 0.56 |
| | **Average** | **-3.04** | **-9.17** | **-8.13** | **-20.93** | **-23.82** | **-0.08** | **0.26** | **0.22** | **0.68** | **0.82** |
| Half pixel | Balloons | 47.50 | -4.29 | -1.60 | -3.28 | -4.40 | -0.60 | 0.19 | 0.07 | 0.15 | 0.21 |
| | Kendo | 20.91 | -7.10 | -2.00 | -2.47 | -2.91 | -0.41 | 0.26 | 0.06 | 0.09 | 0.10 |
| | Doorflowers | 4.56 | -15.65 | -4.72 | -2.28 | -5.00 | -0.05 | 0.46 | 0.14 | 0.06 | 0.14 |
| | Dog | -2.20 | -1.27 | -1.87 | -3.22 | -4.64 | 0.06 | 0.02 | 0.05 | 0.09 | 0.12 |
| | Champ.tower | 21.23 | NA | -3.65 | -19.08 | -25.23 | -0.11 | 0.05 | 0.57 | 1.21 | 1.47 |
| | Pantomime | -7.32 | -5.95 | -4.74 | -10.84 | -13.74 | 0.19 | 0.16 | 0.12 | 0.27 | 0.36 |
| | PoznanStreet | 0.82 | -1.33 | -2.98 | -8.19 | -9.74 | -0.03 | 0.03 | 0.08 | 0.23 | 0.28 |
| | UndoDancer | 16.95 | -1.35 | -2.49 | -10.01 | -11.27 | -0.57 | 0.06 | 0.12 | 0.49 | 0.56 |
| | **Average** | **12.81** | **-5.28** | **-3.01** | **-7.42** | **-9.62** | **-0.19** | **0.16** | **0.15** | **0.32** | **0.40** |

*Note that label 'NA' indicates the BDBR value is not available and is not in the calculation of the average value.

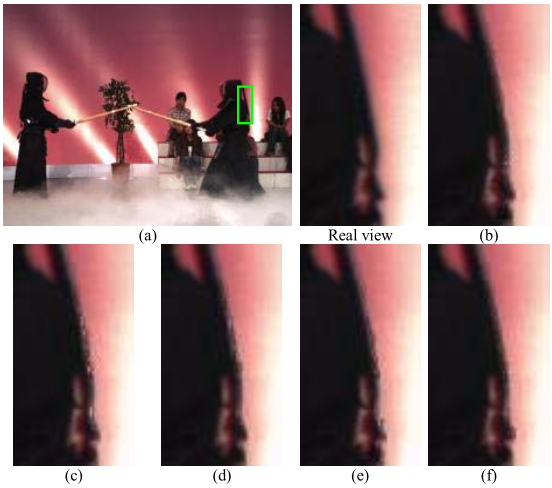


Fig. 9. Visual comparisons among the rendered images, 2nd view of Kendo sequence. (a) Rendered image, (b) enlarged virtual image rendered by using the original color and depth maps, (c) to (f) are enlarged images rendered from the depth maps coded by the JMVC, ZhaoTIP, ZhangTIP and proposed ADD_RDO+DBR scheme, respectively.


Fig. 10. Visual comparisons among the rendered images, 3.5 view of PoznanStreet sequence. (a) Rendered image, (b) enlarged virtual image rendered by using the original color and depth maps, (c) to (f) are enlarged images rendered from the depth maps coded by the JMVC, ZhaoTIP, ZhangTIP and proposed ADD_RDO+DBR scheme, respectively.

bits, but also improves the rendered image quality by optimal mode selection with different *QP*s. The BDPSNR gain of the proposed ADD_DBR scheme is achieved from 0.27 dB to 0.99 dB and 0.63 dB on average for the integer pixel precision setting, as shown in Table II. The proposed ADD_RDO algorithm improves BDPSNR from 0.13 dB to 4.89 dB and 2.19 dB on average when compared with the original JMVC, which is a significant gain. By combining the ADD_DBR and ADD_RDO algorithm together, the proposed overall algorithm (ADD_RDO+DBR) improves BDPSNR from 0.21 dB to 5.76 dB and 2.68 dB on average. The RD gains of the proposed ADD_DBR and ADD_RDO are additive, which indicates the two algorithms improve the RD performance in two different aspects. In terms of the BDBR, the three proposed algorithms achieve 18.74%, 48.19% and 51.29% average bit rate saving, respectively. These significant coding gains indicate that the proposed algorithms are effective in exploiting the spatial redundancies in INTRA depth coding.

Additionally, the virtual view images rendered from different schemes are also visually compared. Fig. 9 and Fig. 10 show the comparisons for Kendo and PoznanStreet, where (a) is an example of rendered image where the green rectangle is enlarged, (b) is the enlarged region of the virtual image rendered by using the original color and depth maps. It is regarded as the ground truth according to [32]. Subfigures (c) to (f) are enlarged images that are rendered from

the depth maps coded by the JMVC, ZhaoTIP, ZhangTIP and the proposed ADD_RDO+ DBR scheme, respectively. As for the Kendo sequence, we can observe that the ground truth is most coincident with the real view. Meanwhile, comparing (c) to (f) with (b), we can observe that there are artifacts in the boundaries and this distortion is gradually reduced from (c) to (f). The rendered image quality from the proposed scheme is the best among the compared benchmarks. For the PoznanStreet sequence, there are some visible artifacts for the ground truth. This is because the noise may be included in the original depth maps. In this paper, we assume that the original depth map is noise free, and we regard the rendered image quality is better if it is closer to the ground truth image [32]. We can find the image by the proposed algorithm is closer to the ground truth compared with those from the benchmark schemes, which thereby proves the effectiveness of the proposed algorithm in the visual comparison.

When the half-pixel precision was used in the view synthesis, the ADD interval $W_{\mathrm{DI}}$ and the ADD range are reduced. The RD comparisons of using half-pixel precision setting are illustrated in bottom part of the Table II. In this setting, ZhaoTIP scheme is inferior to the original JMVC for most sequences because the quantization error plus the smooth error exceeds the ADD interval ($W_{\mathrm{DI}}$) and range more easily. For Dog and Pantomime, ZhaoTIP scheme achieves significant RD improvement mainly because much noise is in these depth sequences as they are generated by DERS. For ZhangTIP scheme, it achieves 0.52 dB BDPSNR gain or 10.77% bit reduction on average compared with the original JMVC. The proposed ADD_DBR, ADD_RDO and ADD_RDO+DBR achieve BDPSNR gains of 0.47 dB, 1.25dB and 1.58 dB on average, respectively. Their bit rate savings are 7.79%, 18.99% and 23.88%, respectively, which are almost half of the bit reduction of the integer pixel precision setting. Though the BDPSNR gains are smaller than those of the integer precision setting due to their smaller $W_{\mathrm{DI}}$, the proposed overall scheme still outperforms the JMVC, ZhaoTIP and ZhangTIP significantly.

Furthermore, the RD performances of the benchmarks and the proposed algorithms were also evaluated when the color images with compressed distortions were rendered in the view synthesis. Color and depth videos were separately compressed with $QP_{\mathrm{C}}$ and $QP_{\mathrm{D}}$, where $QP_{\mathrm{C}}, QP_{\mathrm{D}} \in \{16, 20, 24, 28\}$. The color video was encoded with the original JMVC and depth video was encoded with the tested coding schemes. Their reconstructed color and depth from decoding, $QP_{\mathrm{C}} = QP_{\mathrm{D}}$, were used to synthesize the virtual view images, and total color plus depth bits were counted in $x$-axis [26], [32] for RD evaluation. Table III shows the BDBR and BDPSNR of ZhaoTIP, ZhangTIP and the three proposed algorithms when compared with the original JMVC. We can observe that ZhaoTIP reduces 3.04% bit rate but degrades BDPSNR 0.08dB on average for the integer pixel precision setting. On the other hand, it increases bit rate 12.81% and degrades BDPSNR 0.19dB on average for half-pixel precision setting. Its RD performance is inferior to the original JMVC in average BDPSNR and BDBR. Another benchmark, ZhangTIP scheme, reduces the BDBR 9.17% and 5.28%, respectively, for integer and half pixel precision. Or, it achieves gains of 0.26 dB and 0.16 dB on average in terms of BDPSNR.

The proposed schemes, ADD_DBR, ADD_RDO and ADD_RDO+DBR, achieve 8.13%, 20.93% and 23.82% BDBR reduction on average for integer-pixel precision, respectively, when compared to the original JMVC. Meanwhile, they can achieve 3.01%, 7.42% and 9.62% BDBR reduction on average for half-pixel precision, respectively. In terms of BDPSNR, the proposed ADD_DBR, ADD_RDO and ADD_RDO+DBR can achieve average gains of 0.22 dB, 0.68dB and 0.82 dB, respectively, for the integer pixel precision. And they achieve gains of 0.15 dB, 0.32dB and 0.40dB on average, respectively, for the half-pixel rendering precision. We find that 1) compared with the half pixel precision, more gains are achieved for the integer pixel precision. 2) The proposed ADD_RDO and ADD_RDO+DBR schemes achieve much better coding performance than the comparative benchmarks, including the JMVC, ZhaoTIP and ZhangTIP. 3) The gains are smaller than in the cases rendered by the original color images. It is because the color encoder was not optimized, and meanwhile the color bits were counted in the bit rate which shares the gains.

In addition to the INTRA depth coding, the RD performance for the INTER frame depth coding is also evaluated. Full ME/DE was enabled and their search ranges are $\pm 64$, SAD metric was used for both full and sub-pixel ME/DE search. The number of bi-prediction iteration is 4 and search range for iteration is 8. The maximum number of reference frames is 2 for each memory list and Group-Of-Picture (GOP) length is 8. Three depth views were encoded with MVC structure using Hierarchical B prediction [1]. Six 3D video sequences with different characteristics and resolutions, Balloons, Kendo, Doorflowers, Dog, UndoDancer and PoznanStreet, were tested and their depth videos were encoded. The reconstructed depth images and the original/coded color images were used to render the virtual view images. The integer pixel precision was used in the view synthesis. The PSNR of the virtual view images were measured between the images rendered from the coded color/depth images and the images rendered from the original color/depth images. Five schemes, including JMVC, ZhangTIP, the proposed ADD_DBR, ADD_RDO and ADD_RDO+DBR, were implemented and compared.

From the upper part of the Table IV, we can observe that ZhangTIP scheme reduces the depth bit rate and meanwhile improves the view synthesis image quality. Compared with the original JMVC, it achieves 0.56 dB BDPSNR gain on average or 18.16% BDBR reduction. ADD_DBR achieves 0.34dB BDPSNR gain on average or 12.27% BDBR reduction, which is a little bit inferior to ZhangTIP. Moreover, the ADD_RDO scheme achieves BDPSNR gain of 3.71 dB on average or 58.34% bit reduction. While combining ADD_DBR and ADD_RDO schemes together, it improves BDPSNR more which is 4.07 dB gains on average. Coding gain is especially high for UndoDancer since it has relative larger $W_{\mathrm{DI}}$. Compared with RD performance of INTRA coding, the ADD_DBR algorithm achieves less BDPSNR gain because INTER frames usually contain less residue and thus have smaller room for the bit reduction optimization. Besides, the

TABLE IV
BDBR/BDPSNR COMPARISON FOR INTER AND INTRA CODING (UNIT: %/DB)

| Rendered from decoded depth and the original color images | | | | |
|---|---|---|---|---|
| Test 3D Seq. | ZhangTIP [26] | ADD_DBR | ADD_RDO | ADD_ RDO+DBR |
| Balloons | -28.68/0.55 | -24.65/0.54 | NA/3.46 | -81.25/3.73 |
| Kendo | -34.56/1.17 | -13.42/0.43 | -73.67/3.69 | -74.78/3.95 |
| Doorflowers | -25.11/1.01 | -17.00/0.62 | -28.75/2.51 | -31.53/2.87 |
| Dog | -15.02/0.37 | -9.86/0.22 | -49.80/2.25 | -53.34/2.41 |
| PoznanStreet | -6.40/0.11 | -6.77/0.13 | -69.02/1.70 | -69.58/1.75 |
| UndoDancer | -1.89/0.15 | -1.92/0.12 | -70.49/8.75 | -87.81/9.83 |
| **Average** | **-18.16/0.56** | **-12.27/0.34** | **-58.34/3.72** | **-66.38/4.09** |
| Rendered from decoded depth and color images | | | | |
| Test 3D Seq. | ZhangTIP [26] | ADD_DBR | ADD_RDO | ADD_ RDO+DBR |
| Balloons | -15.73/0.33 | -13.93/0.26 | -60.95/1.80 | -61.73/1.91 |
| Kendo | -22.12/0.59 | -7.94/0.18 | -48.24/1.58 | -49.70/1.68 |
| Doorflowers | -12.25/0.25 | -7.29/0.14 | -27.72/0.70 | -30.29/0.78 |
| Dog | -3.99/0.10 | -1.69/0.04 | -23.59/0.67 | -24.21/0.69 |
| PoznanStreet | -4.06/0.08 | -2.05/0.04 | -34.60/0.96 | -35.40/0.95 |
| UndoDancer | -1.66/0.07 | -0.29/0.02 | -12.87/0.60 | -13.30/0.62 |
| **Average** | **-9.97/0.24** | **-5.53/0.11** | **-34.66/1.05** | **-35.77/1.11** |

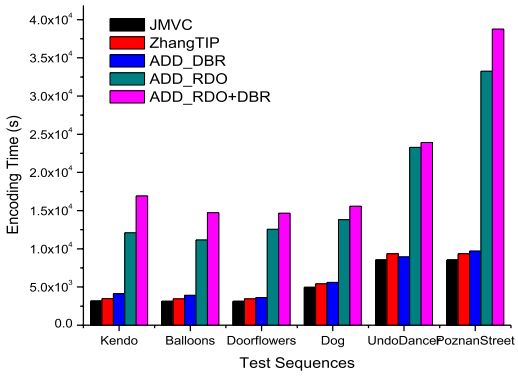


Fig. 11. Computational complexity comparison.

proposed ADD_RDO scheme achieves more BDPSNR gain compared with INTRA depth coding. This is because the new RDO model in the INTER depth coding enables the ME/DE to find most matching block with less view synthesis distortion. Generally, the RD gains achieved by ADD_DBR and ADD_RDO are additive.

Furthermore, the bottom part of the Table IV shows the BDBR and BDPSNR comparison among the four different schemes, where coded color images were used in view synthesis and total bits (i.e. color plus depth bits) were counted. It is observed that ZhangTIP achieves 9.97% BDBR reduction or 0.24dB BDPSNR gain on average when compared with the JMVC. Besides, the proposed ADD_DBR, ADD_RDO and ADD_RDO+DBR achieve 5.53%, 34.66% and 35.77% BDBR reduction or achieve average BDPSNR gains of 0.11dB, 1.05dB and 1.11dB, respectively. Generally, ADD_DBR is better than JMVC but inferior to ZhangTIP. Meanwhile, the proposed ADD_RDO and ADD_RDO+DBR are much better than ZhangTIP scheme. Usually, the RD gains are getting smaller when the color bits possess lager proportion of the total bits because the color video encoder was not optimized.

Fig.11 shows the average encoding time for each GOP among the tested depth coding schemes. Compared with the JMVC, ZhangTIP scheme increases the complexity from 9.5% to 10.1%, 9.8% on average. It is mainly caused by additional operations of image segmentation and pre-analysis on MAD for the depth video. The ADD_DBR increases the coding complexity from 4.9% to 30.8% and 17.0% on average, due to the multiple pass coding for non-zero coefficient blocks. On the other hand, ADD_RDO increases the complexity 247% on average. It is because the new ADD based distortion metric requires additional operations in software implementation, including one division, one subtraction, one if-else operation and several loading operations, when compared with the traditional SAD. It is time-consuming when integrated in the ME/DE and high frequently called by RD cost calculation. The overall algorithm ADD_RDO+DBR increases complexity about 320% due to the multiple pass coding plus the new distortion metric in RD cost calculation.

## V. CONCLUSIONS

In this paper, we formulate view synthesis distortion and depth distortion as a many-to-one mapping relationship, and then present an Allowable Depth Distortion (ADD) model for depth video coding optimization. Based on this ADD model, we propose a new RD model for mode decision and motion/disparity estimation by minimizing the view synthesis distortion at given bit rate. In addition, ADD based depth bit reduction algorithm is also presented to extensively exploit the ADD redundancies and improve coding efficiency. Experimental results over different video sequences, parameters and metrics demonstrate the high efficiency of the proposed ADD based algorithms.

## REFERENCES

[1] K. Muller, P. Merkle, and T. Wiegand, "3D video representation using depth maps," *Proc. IEEE*, vol. 99, no. 4, pp. 643–656, Apr. 2011.
[2] C. Fehn, "Depth-image-based rendering (DIBR), compression and transmission for a new approach on 3D-TV," *Proc. SPIE*, vol. 5291, pp. 93–104, May 2004.
[3] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the stereo and multiview video coding extensions of the H.264/MPEG-4 AVC standard," *Proc. IEEE*, vol. 99, no. 4, pp. 626–642, Apr. 2011.
[4] Y. Zhang, S. Kwong, G. Jiang, X. Wang, and M. Yu, "Statistical early termination model for fast mode decision and reference frame selection in multiview video coding," *IEEE Trans. Broadcast.*, vol. 58, no. 1, pp. 10–23, Mar. 2012.
[5] Y. Zhang, S. Kwong, G. Jiang, and H. Wang, "Efficient multi-reference frame selection algorithm for hierarchical B pictures in multiview video coding," *IEEE Trans. Broadcast.*, vol. 57, no. 1, pp. 15–23, Mar. 2011.
[6] K. Muller and A. Vetro, *AHG Report on 3D Video Coding*, document JCT3V-A1001, ITU-T SG16 WP3&ISO/IEC JTC1/SC29/WG11, Stockholm, Sweden, Jul. 2012.
[7] K. J. Oh, A. Vetro, and Y. S. Ho, "Depth coding using a boundary reconstruction filter for 3D video systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 3, pp. 350–359, Apr. 2011.
[8] S. Liu, P. Lai, D. Tian, and C. Chen, "New depth coding techniques with utilization of corresponding video," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 551–561, Jun. 2011.
[9] V.-A. Nguyen, D. Min, and M. N. Do, "Efficient techniques for depth video compression using weighted mode filtering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 2, pp. 189–202, Feb. 2012.
[10] J. Choi, D. Min, D. Kim, and K. Sohn, "3D JBU based depth video filtering for temporal fluctuation reduction," in *Proc. 17th IEEE ICIP*, Sep. 2010, pp. 2777–2780.
[11] L. Zhu, Y. Zhang, X. Wang, and S. Kwong, "View synthesis distortion elimination filter for depth video coding in 3D video broadcasting," *Multimedia Tools Appl.*, Feb. 2014, doi: 10.1007/s11042-014-1898-1
[12] V. De Silva, W. Fernando, S. Worrall, H. K. Arachchi, and A. Kondoz, "Sensitivity analysis of the human visual system for depth cues in stereoscopic 3D displays," *IEEE Trans. Multimedia*, vol. 13, no. 3, pp. 498–506, Jun. 2011.

[13] D. V. S. X. De Silva, E. Ekmekcioglu, W. A. C. Fernando, and S. T. Worrall, "Display dependent preprocessing of depth maps based on just noticeable depth difference modeling," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 2, pp. 335–351, Apr. 2011.
[14] Q. Liu, Y. Yang, R. Ji, Y. Gao, and L. Yu, "Cross-view down/up-sampling method for multiview depth video coding," *IEEE Signal Process. Lett.*, vol. 19, no. 5, pp. 295–298, May 2012.
[15] Y. Zhao, C. Zhu, Z. Chen, and L. Yu, "Depth no-synthesis-error model for view synthesis in 3D Video," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2221–2228, Aug. 2011.
[16] Z. Pan, Y. Zhang, and S. Kwong, "Fast mode decision based on texture–depth correlation and motion prediction for multiview depth video coding," *J. Real-Time Image Process.*, Mar. 2013, doi: 10.1007/s11554-013-0328-3.
[17] J. Seo, D. Park, H.-C. Wey, S. Lee, and K. Sohn, "Motion information sharing mode for depth video coding," in *Proc. 3DTV-CON*, Tampere, Finland, Jun. 2010, pp. 1–4.
[18] S.-T. Na, K.-J. Oh, C. Lee, and Y.-S. Ho, "Multi-view depth video coding using depth view synthesis," in *Proc. IEEE ISCAS*, May 2008, pp. 1400–1403.
[19] J. Y. Lee, H.-C. Wey, and D.-S. Park, "A fast and efficient multi-view depth image coding method based on temporal and inter-view correlations of texture images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 12, pp. 1859–1868, Dec. 2011.
[20] M.-K. Kang and Y.-S. Ho, "Depth video coding using adaptive geometry based intra prediction for 3D video systems," *IEEE Trans. Multimedia*, vol. 14, no. 1, pp. 121–128, Feb. 2012.
[21] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map distortion analysis for view rendering and depth coding," in *Proc. 16th IEEE ICIP*, Nov. 2009, pp. 721–724.
[22] Q. Zhang, P. An, Y. Zhang, and Z. Zhang, "Efficient rendering distortion estimation for depth map compression," in *Proc. 18th IEEE ICIP*, Sep. 2011, pp. 1105–1108.
[23] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map coding with distortion estimation of rendered view," *Proc. SPIE*, vol. 7543, pp. 75430B-1–75430B-10, Jan. 2010.
[24] T.-Y. Chung, W.-D. Jang, and C.-S. Kim, "Efficient depth video coding based on view synthesis distortion estimation," in *Proc. IEEE VCIP*, Nov. 2012, pp. 1–4.
[25] H.-P. Deng, L. Yu, B. Feng, and Q. Liu, "Structural similarity-based synthesized view distortion estimation for depth map coding," *IEEE Trans. Consum. Electron.*, vol. 58, no. 4, pp. 1338–1344, Nov. 2012.
[26] Y. Zhang, S. Kwong, L. Xu, S. Hu, G. Jiang, and C.-C. J. Kuo, "Regional bit allocation and rate distortion optimization for multiview depth video coding with view synthesis distortion model," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3497–3512, Sep. 2013.
[27] H. Yuan, S. Kwong, J. Liu, and J. Sun, "A novel distortion model and Lagrangian multiplier for depth maps coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 443–551, Mar. 2014.
[28] G. Tech, H. Schwarz, K. Muller, and T. Wiegand, "3D video coding using the synthesized view distortion change," in *Proc. PCS*, May, 2012, pp. 25–28.
[29] 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," *Signal Process., Image Commun.*, vol. 24, no. 8 pp. 666–681, Sep. 2009.
[30] S. Hu, S. Kwong, Y. Zhang, and C.-C. J. Kuo, "Rate-distortion optimized rate control for depth map-based 3D video coding," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 585–594, Feb. 2013.
[31] H. Yuan, Y. Chang, J. Huo, F. Yang, and Z. Lu, "Model-based joint bit allocation between texture videos and depth maps for 3D video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 4, pp. 485–497, Apr. 2011.
[32] B. T. Oh, J. Lee, and D.-S. Park, "Depth map coding based on synthesized view distortion function," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 7, pp. 1344–1352, Nov. 2011.
[33] L. Xiao, M. Johansson, H. Hindi, S. Boyd, and A. Goldsmith, "Joint optimization of communication rates and linear systems," *IEEE Trans. Autom. Control*, vol. 48, no. 1, pp. 148–153, Jan. 2003.
[34] K. Takagi, Y. Takishima, and Y. Nakajima, "A study on rate distortion optimization scheme for JVT coder," *Proc. SPIE*, vol. 5150, pp. 914–923, Jun. 2003.
[35] M. Tanimoto, T. Fujii, and N. Fukushima, *1D Parallel Test Sequences for MPEG-FTV*, document M15378, ISO/IEC JTC1/SC29/WG11, Archamps, France, Apr. 2008.
[36] I. Feldmann *et al.*, *HHI Test Material for 3D Video*, document M15413, ISO/IEC JTC1/SC29/WG11, Archamps, France, Apr. 2008.
[37] M. Domañski *et al.*, *Poznañ Multiview Video Test Sequences and Camera Parameters*, document ISO/IEC JTC1/SC29/WG11 MPEG 2009/M17050, Xi'an, China, Oct. 2009.
[38] *Undo-Dancer Video Sequences*. [Online]. Available: http://mpeg3dv.research.nokia.com, accessed Dec. 2011.
[39] M. Tanimoto, T. Fujii, M. P. Tehrani, and M. Wildeboer, *Depth Estimation Reference Software (DERS) 5.0*, document M16923, ISO/IEC JTC1/SC29/WG11, Xi'an, China, Oct. 2009.
[40] M. Tanimoto, T. Fujii, and K. Suzuki, *View Synthesis Algorithm in View Synthesis Reference Software 3.0 (VSRS 3.0)*, document M16090, ISO/IEC JTC1/SC29/WG11, Feb. 2009.
[41] G. Bjontegaard, *Calculation of Average PSNR Differences Between RD-Curves*, document VCEG-M33, ITU-T Video Coding Experts Group (VCEG), Austin, TX, USA, 2001.

**Yun Zhang** (M'12) received the B.S. and M.S. degrees in electrical engineering from Ningbo University, Ningbo, China, in 2004 and 2007, respectively, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing, China, in 2010. From 2009 to 2014, he was a Post-Doctoral Research Associate and Visiting Scholar with the Department of Computer Science, City University of Hong Kong, Hong Kong. In 2010, he became an Assistant Professor with the Shenzhen Institute of Advanced Technology, CAS, where he has served as an Associate Professor since 2012. His research interests are video compression, 3D video processing, and visual perception.

**Sam Kwong** (M'93–SM'04–F'13) received the B.S. degree in electrical engineering from the State University of New York at Buffalo, Buffalo, NY, USA, in 1983, the M.S. degree in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 1985, and the Ph.D. degree from the University of Hagen, Hagen, Germany, in 1996. From 1985 to 1987, he was a Diagnostic Engineer with Control Data Canada, Montreal, QC, Canada. He joined Bell Northern Research, Ottawa, ON, Canada, as a member of the scientific staff. In 1990, he became a Lecturer with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, where he is currently a Professor with the Department of Computer Science. His research interests are video and image coding and evolutionary algorithms.

**Sudeng Hu** received the B.Eng. degree from Zhejiang University, Hangzhou, China, in 2007, and the M.Phil. degree from the Department of Computer Science, City University of Hong Kong, Hong Kong, in 2010. From 2010 to 2011, he was a Research Associate with the Department of Computer Science, City University of Hong Kong. In 2012, he took an internship with Mitsubishi Electric Research Laboratories, Cambridge, MA, USA. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA, USA. His research interests include image and video compression, rate control, scalable video coding, and 3-D video coding.

**Chung-Chieh Jay Kuo** (F'99) received the B.S. degree from National Taiwan University, Taipei, Taiwan, in 1980, and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1985 and 1987, respectively, all in electrical engineering. He is currently the Director of the Multimedia Communications Laboratory and a Professor of Electrical Engineering, Computer Science, and Mathematics with the Ming-Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA, USA. His research interests include digital image/video analysis and modeling, multimedia data compression, communication and networking, and biological signal/image processing. He has co-authored about 200 journal papers, 850 conference papers, and 10 books. He is a fellow of the American Association for the Advancement of Science and the International Society for Optical Engineers.