Regional Bit Allocation and Rate Distortion Optimization for Multiview Depth Video Coding With View Synthesis Distortion Model

Yun Zhang, *Member, IEEE*, Sam Kwong, *Senior Member, IEEE*, Long Xu, Sudeng Hu, Gangyi Jiang, *Member, IEEE*, and C.-C. Jay Kuo, *Fellow, IEEE*

***Abstract*— In this paper, we propose a view synthesis distortion model (VSDM) that establishes the relationship between depth distortion and view synthesis distortion for the regions with different characteristics: color texture area corresponding depth (CTAD) region and color smooth area corresponding depth (CSAD), respectively. With this VSDM, we propose regional bit allocation (RBA) and rate distortion optimization (RDO) algorithms for multiview depth video coding (MDVC) by allocating more bits on CTAD for rendering quality and fewer bits on CSAD for compression efficiency. Experimental results show that the proposed VSDM based RBA and RDO can improve the coding efficiency significantly for the test sequences. In addition, for the proposed overall MDVC algorithm that integrates VSDM based RBA and RDO, it achieves 9.99% and 14.51% bit rate reduction on average for the high and low bit rate, respectively. It can improve virtual view image quality 0.22 and 0.24 dB on average at the high and low bit rate, respectively, when compared with the original joint multiview video coding model. The RD performance comparisons using five different metrics also validate the effectiveness of the proposed overall algorithm. In addition, the proposed algorithms can be applied to both INTRA and INTER frames.**

***Index Terms*— Multiview depth video, view synthesis distortion model, rate-distortion optimization, bit allocation.**

I. Introduction

RECENTLY, there is an increasing demand for Three-Dimensional video because of its capability of providing real depth perception, immersive vision and novel visual enjoyment for multimedia applications. Multiview Depth Video (MDV) [1] has become one of the most important parts of 3D video since it enables high quality representation of genuine 3D world video content and facilitates virtual view generation for interactive 3D video. To compress these large volumes of MDV efficiently, Multiview Depth Video Coding (MDVC) has recently attracted much attention. Before MDVC, Multiview Video Coding (MVC) [2] has been developed based on the state-of-the-art H.264/AVC standard with joint efforts by the Joint Video Team (JVT) of Motion Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG). Several advanced techniques have been utilized in MVC for encoding Mulitview Color Video (MCV) data, including illumination compensation and view synthesis/view interpolation prediction. MDV can be treated as the illumination component of the color video and the encoding mechanism for MDV is similar to MCV. Thus, MDVC is developed based on H.264/MVC [2] and the traditional MVC algorithms could be extended to depth encoding. In fact, MDV has a quite different view-temporal-frequency correlation from MCV. Also, the MDV is used as geometrical information for virtual view rendering instead of being watched directly, which has totally different properties compared with that of the color video. In this case, traditional MVC algorithms originally designed for MCV are not efficient enough when they are directly applied to MDV encoding. To achieve high coding efficiency for MDVC, a number of techniques were proposed for MDVC optimization in the literatures. They could be broadly classified into two categories as, depth enhancement in pre-/post-processing and depth video coding optimization.

Due to the fact that the original MDV may be noisy and have large number of residues after INTER/INTRA frame prediction, some pre-/post-processing [3]–[11] techniques were proposed for high efficient MDVC. Since the object boundaries of depth severely affect the rendering quality, Oh *et al.* [3] proposed a boundary reconstruction filter to preserve the sharp boundary of the reconstructed depth video. Also, this filter was applied to preserve boundary for down/up sampling related depth video coding [4]. As recognizing that the depth edges are more important for the rendered Virtual View Image (VVI) quality, Ekmekcioglu *et al.* [5] proposed an edge adaptive upsampling method for the depth edges. Additionally, spatial fusion filter [6] and temporal smoothing filters [7] were used to reduce high frequency prediction residues caused by depth noise and temporal inconsistency. Depth refinement techniques

Manuscript received October 5, 2012; revised March 13, 2013 and May 15, 2013; accepted May 22, 2013. Date of publication June 3, 2013; date of current version August 5, 2013. This work was supported in part by the Hong Kong RGC General Research Fund under Project 9041495 (CityU 115109), the Natural Science Foundation of China under Grants 61102088, 61272289 and 61202242, Shenzhen Emerging Industries of the Strategic Basic Research Project under Grant JCYJ20120617151719115, and the Guangdong Nature Science Foundation under Grant S2012010008457. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Karsten Mueller.

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, Kowloon 999077, Hong Kong (e-mail: yunzhang@cityu.edu.hk).

S. Kwong is with the Department of Computer Science, City University of Hong Kong, Kowloon 999077, Hong Kong (e-mail: cssamk@cityu.edu.hk).

L. Xu is with the School of Automation and Electrical Engineering, University of Science and Technology, Beijing 100876, China (e-mail: lxu@jdl.ac.cn).

S. Hu and C.-C. J. Kuo are with the Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089-2564 USA (e-mail: sudenghu@usc.edu; cckuo@sipi.usc.edu).

G. Jiang is with Department of Information Science and Engineering, Ningbo University, Ningbo 315211, China (e-mail: jianggangyi@nbu.edu.cn).

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.2013.2265883

[8]–[10] were proposed to reduce the noise in depth video by using textural information in color video. Zhao *et al*. [11] presented a depth no-synthesis-error model which conceals the depth quantization and filtering error in view synthesis. These schemes [3]–[11] mainly attempted to refine the depth video in pre-/post-processing and to reduce coding residue while maintaining high rendering quality. However, the depth map's regional selective impacts to the virtual view synthesis were not considered for these schemes. Moreover, the correlation between depth and texture video had not been utilized.

As for the depth coding optimization, Silva *et al*. [12], [13] presented the Just Noticeable Depth Differences (JNDD) and analyzed the device dependent visual sensitivity of depth perception. The compression distortion was controlled within the JNDD range so that the depth distortion was imperceptible. However, the depth distortion effect on objective VVI quality was not considered. Since depth video is not perceived directly by viewer but mainly utilized for virtual view rendering, a number of efforts had been devoted to improve the coding efficiency. Na *et al*. [14] improved inter-view depth prediction accuracy by using rendered depth as a reference. Since depth video is smooth and contains less texture, the depth video may not be necessary to maintain high fidelity and resolution. Therefore, reduced resolution depth video coding [15] was used.

Due to high correlation between MDV and MCV, a number of joint 3D video coding algorithms were developed aiming at improving the VVI quality with bit rate constraints. In [16], joint two-pass bit allocation and its improved schemes were presented for MDV and MCV channels to maximize rendering quality by using a linear D-Q model. Lee *et al*. [17] proposed a depth video coding scheme that helped to encode blocks with SKIP mode based on its correlation with the previously encoded texture images. In addition, the motion similarity [19], [20] and structure similarity [21] among the texture and depth video were exploited to facilitate depth video coding aiming at improving coding efficiency and lowering complexity. To further lower the complexity, several specific MDVC fast algorithms, including fast INTER [22] and INTRA mode decision [23] were developed. These algorithms mainly focus on the coding optimization and are based on information theory and statistics method. To explore the depth redundancies, Yuan *et al.* [24] analyzed the depth/color effects to VVI quality and presented that VVI distortion was linear to depth and color distortion, where depth and color distortion were additive. Oh *et al*. [20] found depth distortion is multiplicative to color difference while affecting the VVI quality. Also, linear model [25] and powered model [26] were proposed for 3D depth video coding. However, the regional properties of the depth video have not been analyzed and fully exploited for the depth coding.

In this paper, we analyze the regional relationship between the depth and synthesized image and present a View Synthesis Distortion Model (VSDM). Based on this VSDM model, Regional Bit Allocation (RBA) and Rate Distortion Optimization (RDO) algorithm are then presented to improve the compression efficiency. The paper is organized as follows: motivations and analyses are presented in Section II.

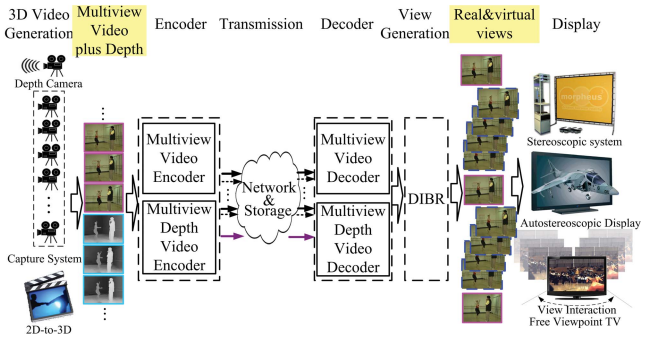


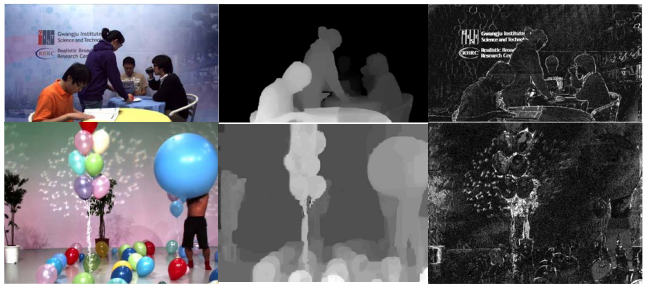
Fig. 1. Framework of 3D video system.

Section III presents the proposed VSDM, where the relationship between the depth distortion and VVI quality is analyzed in detail. Then in Section IV, regional RBA and RDO algorithms for MDVC are presented based on this new distortion model. Experimental results and analyses are presented in Section V. Finally, conclusions are drawn in Section VI.

II. Motivations and Analyses

Multiview video plus depth [1], which consists of MCV and corresponding MDV, has been the mainstream data format for 3D video owning to its high rendering quality, low complexity and flexible arbitrary view rendering. Fig. 1 shows a general framework of the 3D video system. In the system, MCV is obtained by multiple cameras capturing a scene simultaneously from slightly different positions/angles. MDV generated by stereo matching based algorithms or captured by depth cameras provides the depth information for the corresponding MCV. These multiview color videos plus depth maps are encoded at server and their bitstreams are transmitted to the client for decoding. Then, at the client, the decoded multiview color and depth videos are used to synthesize the intermediate VVIs, which are used for interactivity or depth perception of 3D applications, such as 3DTV and Free-viewpoint TV. Since the rendered VVIs are finally perceived by the viewers, they are required to maintain high quality with the constraint of total bit rate. Since VVIs are synthesized by referencing the multiview color and depth video, the image quality of VVIs depends on the quality of color and depth video as well as the synthesis algorithms/tools. In this work, we focus on analyzing VVI distortion caused by the depth distortion and perform the depth video coding optimization. Thus, we encode the MCV and MDV separately.

In Depth Image Based Rendering (DIBR) [27], the distortions in the depth map will cause geometrical displacement for the pixels of VVIs, which consequently degrade the quality of VVIs. However, the magnitudes of those image quality degradations are different over different regions even with the same distortion level in the depth map. They are correlated with the corresponding color texture. Fig. 2 shows the difference between VVIs that are rendered by original depth video and by white noise distorted depth video, where the white noise is evenly distributed. The gray images in Fig. 2(c) illustrate the errors caused by distorted depth map. The dark areas indicate small errors while bright areas indicate large errors. From the figures, it is found that the bright pixels not only



(a) original image (b)depth map (c)VVI distortion

Fig. 2. Difference between VVIs that rendered by original depth map and distorted depth map. Up: Café, Bottom: Balloons. (a) Original image. (b) Depth map. (c) VVI distortion.

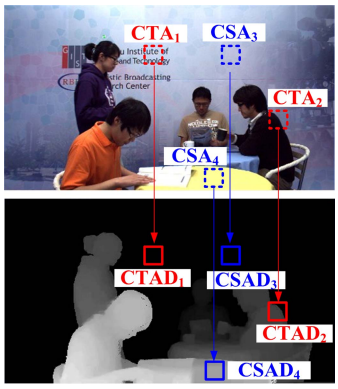


Fig. 3. Graphic explanation of CTA/CSA and CTAD/CSAD in 3D video.

locate at the depth boundaries, but also at the background color boundaries where the depth is smooth, e.g. the background words (Cafe) and texture background (Balloons). In other words, it is the depth distortion in the texture boundaries that has more important impacts on the VVI quality, instead of depth boundaries only. Therefore, in depth video coding, the depth regions corresponding to texture boundaries shall be given high priority and well preserved. Here arise two major problems need to be solved: First is the regional mathematical relationship between the depth distortions and quality of VVIs; Second is the MDVC optimization using the regional properties.

III. Proposed Regional VSDM

Usually, the VVI quality degradation caused by depth distortion in textural area will be more severe than those in smooth areas. To analyze this phenomenon, we divided the MCV and corresponding MDV into two kinds of regions, Color Texture Area (CTA) and Color Smooth Area (CSA), which correspond to texture and smooth areas in color video, respectively. The CTA and CSA in the corresponding depth video, i.e. collocated CTA and CSA in the depth video, are denoted by CTAD and CSAD, respectively, Fig. 3 shows an example of CTA/CSA and CTAD/CSAD. The color image is divided into CTA and CSA. Correspondingly, the depth image can be divided into CTAD and CSAD. It is noted that the CTAD includes not only depth edges but also smooth depth area.

A. Theoretical Analyses on Depth Distortion Effects to VVI

In this section, the depth distortion effects on the synthesized image quality will be analyzed in detail. In addition, the regional properties of the depth distortion effect in the CTAD and CSAD are presented as well.

Fig. 4 shows pixel mapping with different depth videos in the rendering process. O_L and O_R are the optical center for the left and right view, respectively. O_V is the optical center for the virtual view. Suppose $I_V(i, j)$ is a pixel value at (i, j) of VVI generated by view synthesis with original depth video. Initially, the pixel $I_L(i_{L1}, j_{L1})$ in left image and pixel $I_R(i_{R1}, j_{R1})$ in right image are warped to generate virtual view pixel $I_V(i, j)$. Generally, there are mainly two kinds of fusion techniques for VVI generation [27]. One is a weighted fusion in which each virtual view pixel is weighted added

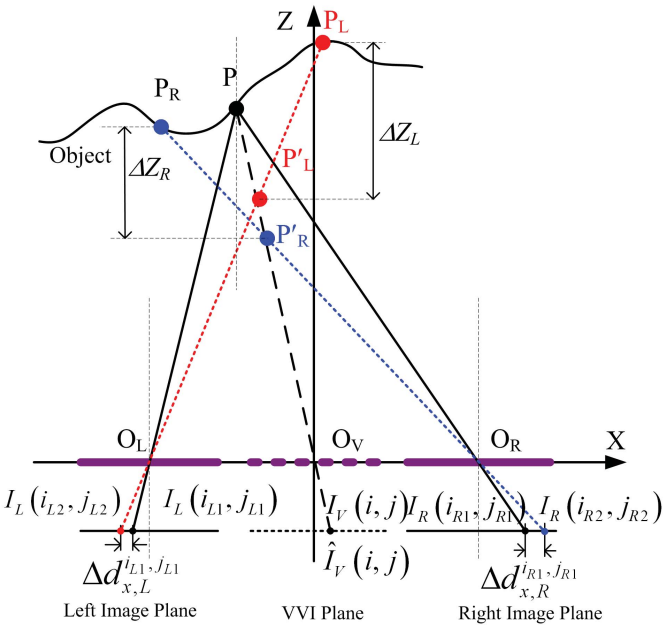


Fig. 4. Pixel mapping with different depth videos.

from visible pixels of the neighboring views. The other is to select one view as the chief reference, and to pick the pixels of the other view to fill the hole and occlusion area. This fusion method can be regarded as a special case of the weighted fusion. Thus, $I_V(i, j)$ can be calculated as

$$I_V(i, j) = \omega_L I_L(i_{L1}, j_{L1}) + \omega_R I_R(i_{R1}, j_{R1}), \quad (1)$$

where ω_L and ω_R are weighting coefficients depending on the view rendering algorithms, virtual view positions and occlusions, and they satisfy $\omega_L + \omega_R = 1$; $I_L(i_{L1}, j_{L1})$ and $I_R(i_{R1}, j_{R1})$ are pixels in left and right views at position (i_{L1}, j_{L1}) and (i_{R1}, j_{R1}), respectively. For the second view synthesis case, ω_L and ω_R are either 1 or 0.

To analyze the depth distortion effect to the pixel mapping, we fixed the color view images and changed the corresponding depth images with distorted depth images, and rendered the VVI $\hat{I}_V$. While using the decoded depth video with compression distortion, virtual view pixel $\hat{I}_V(i, j)$ was rendered from the pixels $I_L(i_{L2}, j_{L2})$ and $I_R(i_{R2}, j_{R2})$ in left and right reference images. The value of virtual view pixel at (i, j),

$\hat{I}_V(i, j)$, is

$$\hat{I}_V(i, j) = \omega_L I_L(i_{L2}, j_{L2}) + \omega_R I_R(i_{R2}, j_{R2}). \quad (2)$$

The depth distortions may introduce the geometrical rendering position errors while mapping pixels of reference images to the VVI. As shown in Fig. 4, when the color images are fixed, $I_L(i_{L2}, j_{L2})$ and $I_R(i_{R2}, j_{R2})$ are the corresponding neighboring pixels of $I_L(i_{L1}, j_{L1})$ and $I_R(i_{R1}, j_{R1})$. The geometrical relationship between pixels $I_\varphi(i_{\varphi 1}, j_{\varphi 1})$ and $I_\varphi(i_{\varphi 2}, j_{\varphi 2})$, $\varphi \in \{L, R\}$, can be presented as

$$I_\varphi\left(i_{\varphi 1} + \Delta d_{x,\varphi}^{i_{\varphi 1}, j_{\varphi 1}}, j_{\varphi 1} + \Delta d_{y,\varphi}^{i_{\varphi 1}, j_{\varphi 1}}\right) = I_\varphi\left(i_{\varphi 2}, j_{\varphi 2}\right), \quad (3)$$

where $\Delta d_{x,\varphi}^{i_{\varphi 1}, j_{\varphi 1}}$ and $\Delta d_{y,\varphi}^{i_{\varphi 1}, j_{\varphi 1}}$ are the horizontal and vertical rendering position errors caused by depth distortion at position $(i_{\varphi 1}, j_{\varphi 1})$ in view φ, respectively. They are caused by the depth distortion in view φ and different from pixel to pixel.

Based on the above two rendering processes, the view synthesis distortion caused by depth distortion can be generally measured by the sum of the differences between the VVI pixels $I_V(i, j)$ and $\hat{I}_V(i, j)$. The Mean Squared Error (MSE) of virtual image caused by depth distortion, MSE_{VS}, can be calculated as

$$MSE_{VS} = \frac{1}{MN}\sum\sum\left[I_V(i, j) - \hat{I}_V(i, j)\right]^2, \quad (4)$$

where M and N are the height and width of the rendered image. Appling (1), (2) and (3) to (4), we can obtain

$$\begin{aligned} MSE_{VS} &= \frac{1}{MN}\sum\sum\left[I_V(i, j) - \hat{I}_V(i, j)\right]^2 \\ &= \frac{1}{MN}\sum_{\phi\in\{L,R\}}\sum\sum\omega_\phi^2 \\ &\quad \times\left[I_\phi(i, j) - I_\phi\left(i + \Delta d_{x,\phi}^{i,j}, j + \Delta d_{y,\phi}^{i,j}\right)\right]^2 \\ &\quad + C_{LR}, \end{aligned} \quad (5)$$

where $C_{LR} = \frac{2}{MN}\sum\sum\prod_{\varphi\in\{L,R\}}\omega_\varphi[I_\varphi(i, j) - \hat{I}_\varphi(i, j)]$, $\varphi \in \{L, R\}$. For the first image fusion case where ω_L and ω_R are non-zero and $\omega_L + \omega_R = 1$, the item C_{LR} approximates to zero according to the Law of Large Number (LLN) [24]. For the second image fusion case, C_{LR} is equal to zero because either ω_L or ω_R is zero. Therefore, the average synthesis distortion, MSE_{VS}, can be formulated as

$$MSE_{VS} \approx \sum_{\varphi\in\{L,R\}}\omega_\varphi^2 MSE_\varphi\left(\Delta d_{x,\varphi}, \Delta d_{y,\varphi}\right), \quad (6)$$

where

$$\begin{aligned} &MSE_\varphi\left(\Delta d_{x,\varphi}, \Delta d_{y,\varphi}\right) \\ &= \frac{1}{MN}\sum\sum\left[I_\varphi(i, j) - I_\varphi\left(i + \Delta d_{x,\varphi}^{i,j}, j + \Delta d_{y,\varphi}^{i,j}\right)\right]^2, \\ &\quad \varphi \in \{L, R\}. \end{aligned}$$

It indicates that MSE_{VS} is approximated to the weighted sum of MSE_φ in left and right views. Because video content in left and right views are highly similar, the distortion effects on the left view are similar to the right view. Thus, we can analyze the distortion effects on one view and obtain the effects on the other view by similar deduction. In the following sections, let Δd_φ be variable taking values $\Delta d_{x,\varphi}^{i,j}$ and $\Delta d_{y,\varphi}^{i,j}$ for simplicity. According to the view synthesis process, we know that the rendering position offset Δd_φ is caused by depth distortion Δv_φ, and the MSE_{VS} is derived from Δd_φ. Therefore, to analyze the relationship between Δv_φ and MSE_{VS}, we divide the analysis into two sub-steps: establish 1) the relationship between Δv_φ and Δd_φ, and 2) the relationship between MSE_{VS} and Δd_φ.

B. Relationship Between Rendering Position Offset Δd_φ and Depth Distortion Δv_φ

According to the 3D warping process in the virtual view generation, the virtual view pixel can be calculated as [27]

$$p_2 = z_1\mathbf{A}_2\mathbf{R}_2\mathbf{R}_1^{-1}\mathbf{A}_1^{-1}p_1 - \mathbf{A}_2\mathbf{R}_2\mathbf{R}_1^{-1}\mathbf{t}_1 + \mathbf{A}_2\mathbf{t}_2, \quad (7)$$

where $p_2 = [l, m, n]^{\mathrm{T}}$ and $p_1 = [x, y, 1]^{\mathrm{T}}$ are the corresponding pixels in rendered virtual view image and real image, respectively. z_1 is the depth for p_1. $\mathbf{A}_1$ and $\mathbf{A}_2$ are two 3×3 matrices indicating camera intrinsic parameters for the virtual view camera and real camera, respectively. $[\mathbf{R}_1, \mathbf{t}_1]$ and $[\mathbf{R}_2, \mathbf{t}_2]$ are extrinsic parameters for the two cameras; $\mathbf{R}_1$ and $\mathbf{R}_2$ are the rotation matrices; $\mathbf{t}_1 = [t_{10}, t_{11}, t_{12}]^{\mathrm{T}}$ and $\mathbf{t}_2 = [t_{20}, t_{21}, t_{22}]^{\mathrm{T}}$ are translation factors. They are the positions of the origin of the world coordinate system expressed in the coordinates of the camera centered coordinate system. Suppose the real camera and virtual camera have the same intrinsic parameters and have the same rotation angle, we can get

$$\mathbf{A}_1 = \mathbf{A}_2 = \begin{pmatrix} f_x & \lambda & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{R}_1 = \mathbf{R}_2, \quad (8)$$

where f_x and f_y are the focal length in horizontal and vertical direction, respectively. u_0 and v_0 represent the principal points of the centre of the image. λ represents the skew coefficient between x and y axis, and it equals to zero when the cameras are well calibrated. Applying (8) into (7), we obtain

$$\begin{pmatrix} l \\ m \\ n \end{pmatrix} = z_1\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} + \begin{pmatrix} f_x & \lambda & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} t_{20} - t_{10} \\ t_{21} - t_{11} \\ t_{22} - t_{12} \end{pmatrix}. \quad (9)$$

Therefore, solving (9), the disparities in horizontal and vertical directions, d_x and d_y, can be calculated as

$$\begin{cases} d_x = x - \frac{z_1 x + f_x(t_{20} - t_{10}) + \lambda(t_{21} - t_{11}) + u_0(t_{22} - t_{12})}{z_1 + (t_{22} - t_{12})} \\ d_y = y - \frac{z_1 y + f_y(t_{21} - t_{11}) + v_0(t_{22} - t_{12})}{z_1 + (t_{22} - t_{12})} \end{cases}. \quad (10)$$

If the cameras are positioned in parallel and well geometrically calibrated, we have $t_{12} = t_{22}$, $\lambda = 0$. Therefore, (10) can be simplified as

$$d_\phi = f_\phi l_\phi / z_1, \quad \phi \in \{x, y\}, \quad (11)$$

where $l_x = t_{10} - t_{20}$ is the horizontal baseline, and $l_y = t_{11} - t_{21}$ is the vertical displacement of the two cameras. In MPEG 3D video, a non-linear quantization process [28] is adopted

to transform depth Z to depth value v in the range from 0 to 255, which is

$$v = Q(Z) = \left\lfloor 255 \frac{Z_{near}}{Z} \frac{Z_{far} - Z}{Z_{far} - Z_{near}} + 0.5 \right\rfloor, \quad (12)$$

where Z_{near} and Z_{far} are the nearest and farthest depth planes of the video scene, which correspond to depth value 0 and 255, respectively; v is the quantized depth value with 8-bit depth, "$\lfloor\ \rfloor$" is floor operation. Therefore, the relationship between depth distortion Δv and rendering position offset Δd_ϕ (i.e. rendering position error) can be expressed as

$$\Delta d_\phi = f_\phi l_\phi C_1 \Delta v, \quad (13)$$

where $C_1 = \left(1/Z_{near} - 1/Z_{far}\right)/255$. It indicates that in parallel camera setting, the rendering error offset Δd_ϕ has linear relationship with the depth distortion Δv, focal length and the camera baseline distance with given texture video, which was also proved in [26]. For other non-parallel camera system, the rendering error offset Δd_ϕ is a monotonic function of Δv [29]. In this paper, we focus on the first case where cameras are aligned parallel. Since (13) can be applied to both the left view and right view in the rendering process, it can be rewritten as

$$\Delta d_{\phi,\varphi} = k_{\phi,\varphi} \Delta v_\varphi, \quad (14)$$

where $k_{\phi,\varphi}$ is a coefficient, $k_{\phi,\varphi} = f_{\phi,\varphi} l_{\phi,\varphi} C_1$, $\phi \in \{x, y\}$, $\varphi \in \{L, R\}$.

C. Relationship Between Pixel-Wise Position Offset Δd_φ and View Synthesis Distortion MSE_{VS}

Because of the depth distortion Δv, view synthesis distortion MSE_{VS} will be caused by mapping neighboring pixels with rendering position offset Δd_φ to the current rendered pixel, as represented by (6). Since MSE_{VS} is the weighted sum of $MSE_\varphi(\Delta d_{x,\varphi}, \Delta d_{y,\varphi})$, the relationship between Δd_φ and MSE_{VS} can be easily deduced from the relationship between Δd_φ and $MSE_\varphi(\Delta d_{x,\varphi}, \Delta d_{y,\varphi})$. According to LLN, $MSE_\varphi(\Delta d_{x,\varphi}, \Delta d_{y,\varphi})$ can be approximated to be

$$MSE_\varphi\left(\Delta d_{x,\varphi}, \Delta d_{y,\varphi}\right) \approx f_\varphi\left(\Delta d_{x,\varphi}, 0\right) + f_\varphi\left(0, \Delta d_{y,\varphi}\right), \quad (15)$$

where

$$f_\varphi(x, y) = \frac{1}{MN} \sum_{i=1}^{N} \sum_{j=1}^{M} \left[I_\varphi(i, j) - I_\varphi\left(i + x^{i,j}, j + y^{i,j}\right) \right]^2,$$

$x^{i,j}$ and $y^{i,j}$ are the values of x and y at position (i, j), M and N are the width and height of the image.

To analyze the relationship between $MSE_\varphi(\Delta d_{x,\varphi}, \Delta d_{y,\varphi})$ and $\Delta d_{x,\varphi}$, $\Delta d_{y,\varphi}$, we suppose the average vertical and horizontal offset are the same, i.e. $\Delta d_\varphi = \Delta d_{x,\varphi}^{i,j} = \Delta d_{y,\varphi}^{i,j}$ and different $\Delta d_\varphi \in \{1, 2, 3, 4, 5, 6, 7\}$ are tested. Fig. 5 illustrates the relationship between the rendering position offset Δd_φ^2 and the average MSE for CTAD/CSAD. The x-axis is the squared rendering position offset Δd_φ^2 and the y-axis is the $MSE_{CTAD,\varphi}(\Delta d_\varphi)$ or $MSE_{CSAD,\varphi}(\Delta d_\varphi)$, which are the $MSE_\varphi(\Delta d_{x,\varphi}, \Delta d_{y,\varphi})$ in CTAD or CSAD regions, respectively. From the figure, the relationship between Δd_φ^2 and

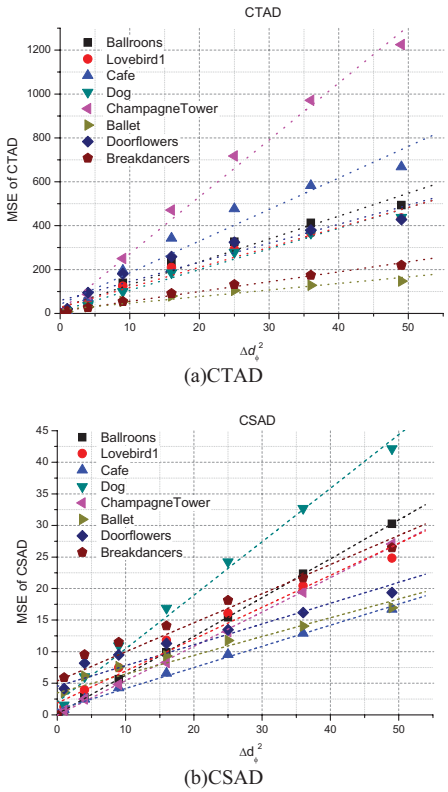


Fig. 5. The relationship between distortion and Δd_φ^2 for CTAD and CSAD regions. (a) CTAD. (b) CSAD.

MSE of CTAD can be approximated as a linear function for $\Delta d_\varphi \in \{1, 2, 3, 4, 5, 6, 7\}$. The average correlation coefficients over different test sequences by using linear fitting are more than 0.98 for CTAD. The correlation coefficient indicates the goodness of fitting and the fitting is better when it is closer to 1. Other fitting algorithms, such as the exponential function or power model, are also possible, especially for larger Δd_φ. However, we fit MSE_{CTAD} and Δd_φ^2 with linear function for simplicity. Thus, the relationship between MSE_{CTAD} and Δd_φ^2 in CTAD can be represented as

$$MSE_{CTAD,\varphi}\left(\Delta d_\varphi\right) = \beta_{CTAD,\varphi} \cdot \Delta d_\varphi^2 + \gamma_{CTAD,\varphi}. \quad (16)$$

Similarly, for the CSAD regions, the relationship between Δd_φ^2 and MSE_{CSAD} can also be approximated to be linear, where the average correlation coefficient of the linear fitting for CSAD is 0.99. Therefore,

$$MSE_{CSAD,\varphi}\left(\Delta d_\varphi\right) = \beta_{CSAD,\varphi} \Delta d_\varphi^2 + \gamma_{CSAD,\varphi}, \quad (17)$$

where $\beta_{CTAD,\varphi}$, $\beta_{CSAD,\varphi}$, $\gamma_{CTAD,\varphi}$ and $\gamma_{CSAD,\varphi}$ are coefficients. Moreover, it is obvious that the $\beta_{CTAD,\varphi}$ is much larger than $\beta_{CSAD,\varphi}$, which indicates that the distortion MSE_{VS} is more easily affected by the rendering position errors in CTAD than those in CSAD.

The view synthesis distortion MSE_{VS} of an image is the sum of MSE_{VS} in the CTAD and CSAD regions, thus

$$MSE_{VS} = MSE_{VS,CTAD} + MSE_{VS,CSAD}, \quad (18)$$

$$MSE_{VS,\psi} = \sum_{\varphi \in \{L,R\}} \omega_\varphi^2 \left(\beta_{\psi,\varphi} \cdot \left(k_\varphi \cdot \Delta v_{\varphi,\psi}\right)^2 + \gamma_{\psi,\varphi} \right), \quad (19)$$

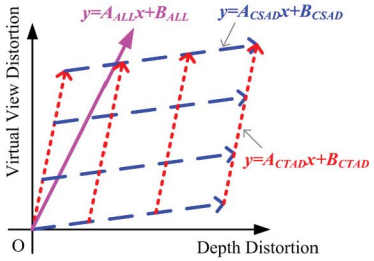


Fig. 6. Graph explanation for the relationship between depth distortion and virtual view distortion.

where $\psi \in \{CTAD, CSAD\}$. Suppose the MDVC is symmetrical and quantization errors introduced in each view are similar, the depth distortion of left view is similar to that of the right view, i.e. $\Delta v_{L,\psi} \approx \Delta v_{R,\psi} = \Delta v_{\psi}$. (19) can be rewritten as

$$MSE_{VS,\psi} = \Delta v_{\psi}^2 \sum_{\varphi \in \{L,R\}} \beta_{\psi,\varphi} \omega_{\varphi}^2 \cdot k_{\varphi}^2 + \sum_{\varphi \in \{L,R\}} \omega_{\varphi}^2 \gamma_{\psi,\varphi}. \tag{20}$$

On the other hand, let $MSE_{D,\psi}$ be the depth distortion and it equals to Δv_{ψ}^2. Meanwhile, since the image content and texture are quite similar among neighboring views, the slope $\beta_{\psi,\varphi}$ is similar among different viewpoint images and we have $\beta_{\psi} \approx \beta_{\psi,\varphi}$. Therefore, (20) can be rewritten as

$$MSE_{VS,\psi} = A_{\psi} MSE_{D,\psi} + B_{\psi}, \tag{21}$$

where $A_{\psi} = \beta_{\psi} \sum_{\varphi \in \{L,R\}} \omega_{\varphi}^2 \cdot k_{\varphi}^2$, $B_{\psi} = \sum_{\varphi \in \{L,R\}} \omega_{\varphi}^2 \gamma_{\psi,\varphi}$. $\psi \in \{CTAD, CSAD\}$, A_{ψ} and B_{ψ} are constants under a given 3D video content, camera setting, view synthesis and fusion algorithm, etc.

D. The Proposed Regional D_{VS}-D_D Model

Based on the above analyses, we can model the relation between distortion of synthesized virtual view, $D_{VS,\psi}$, and depth distortion,$D_{D,\psi}$, as linear over different regions, which can be presented as

$$D_{VS,\psi} = A_{\psi} D_{D,\psi} + B_{\psi}, \tag{22}$$

where $\psi \in \{CTAD, CSAD, ALL\}$, 'CSAD' and 'CTAD' represent CSAD and CTAD regions, respectively, and 'ALL' is for the entire image. The distortions $D_{VS,\psi}$ and $D_{D,\psi}$ are the distortions of synthesized virtual view and depth video, which are measured with MSE. Coefficient A_{ψ} is the gradient indicating the increasing ratio of view synthesis distortion; B_{ψ} is the coefficient that correlated with initial depth distortion in non-ψ region. Since the depth distortion for the CTAD regions usually has larger impacts on the depth distortion of CSAD regions, A_{ψ} satisfies

$$A_{CTAD} \geq A_{ALL} \geq A_{CSAD}, \tag{23}$$

$$A_{ALL} = \mu \cdot A_{CSAD} + (1 - \mu) A_{CTAD} \tag{24}$$

where μ is the ratio of the number of pixels in CSAD to the number of pixels in entire image. Fig. 6 shows a graph of the modeling relationship between depth distortion and virtual view distortion, where the dot lines illustrate the linear relationship between D_{VS} and D_{D} for CTAD regions, the dash lines are for the CSAD regions, and the solid line is for the entire image. In the next subsection, white noise injection and coding experiments are performed to verify the model.

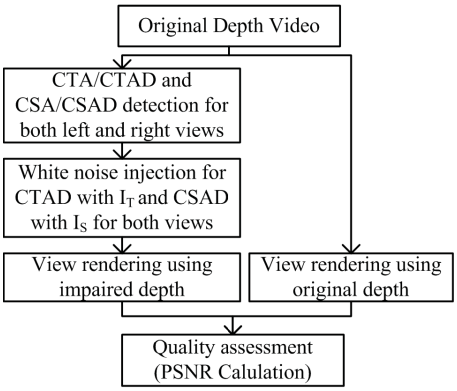
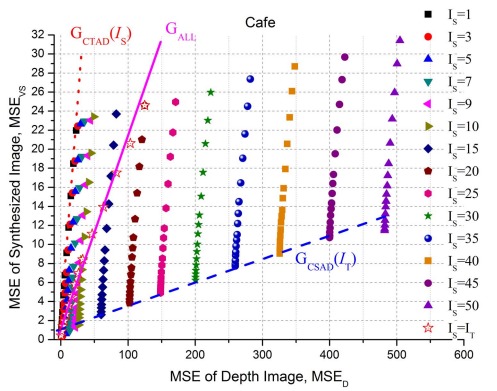


Fig. 7. Flowchart of the view synthesis distortion analysis experiment.

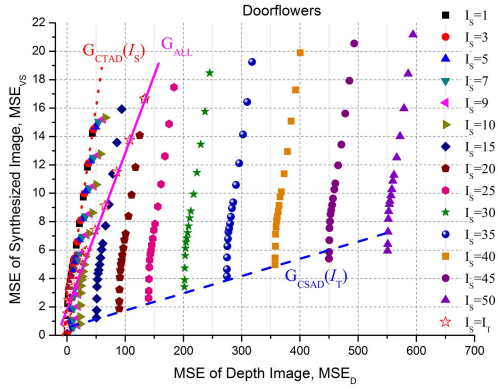
E. Experimental Verification on Depth Distortion Effects to View Synthesis Distortion

Since the quantization noise resulting from uniform scalar quantization can be modeled as a white noise model [30], white noise with zero mean is injected into the CTAD and CSAD regions for analyzing the depth distortion effect on synthesized image. Fig. 7 illustrates the flowchart of the view synthesis distortion analysis experiment. CTA and CSA regions were detected by Canny operator, and 16 × 16 macroblock (MB) was labeled as texture block when the number of detected edge pixels in the MB are larger than the threshold N_{e}. Smaller N_e indicates that more blocks will be classified as the texture area, i.e. CTAD for depth and CTA for color. Larger N_e indicates that more blocks will be classified as the smooth area, i.e. CSAD for depth and CSA for color. The N_e is empirically set to a small value, i.e. 5, in order not to miss the edge pixels. Let I_{T} be the white noise intensity in CTAD area and I_{S} be the white noise intensity in CSAD area. The I_{T} is set as {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14, 16, 18, 20} and I_{S} is set as {1, 3, 5, 7, 9, 10, 15, 20, 25, 30, 35, 40, 45, 50}. Two sequences (Café and Doorflowers) were tested. Finally, the Peak Signal-to-Noise Ratio (PSNR) was calculated between the VVIs synthesized from impaired depth and that from original depth. In this test, the image rendered by original depth video is regarded as the ground truth.

Fig. 8 illustrates view synthesis image quality analysis with different white noise intensities and areas (CTAD and CSAD). The y-axis is the view synthesis image distortion measured with MSE, denoted by MSE_{VS}, and the x-axis is the average depth image distortion, measured with MSE and denoted by MSE_{D}. Each symbol with different colors indicates one magnitude of white noise (I_{S}) is injected in CSAD regions and different magnitudes of white noise (I_{T}s) are injected to CTAD. Here, we formulate line by linear fitting for each kind of symbols, where different I_{T}s are injected to CTAD and one fixed I_{S} intensity is injected in CSAD. The slopes of these lines are denoted by $G_{\mathrm{CTAD}}(I_{\mathrm{S}})$, as shown by the dot line in

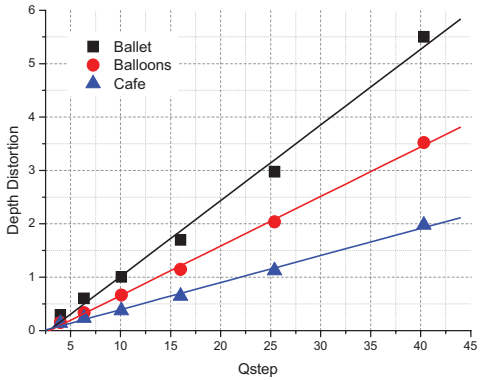


(a) Café

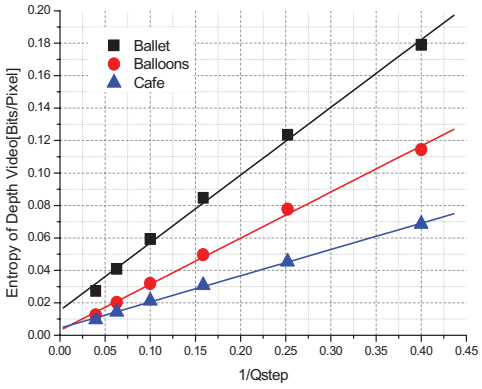


(b)Doorflowers

Fig. 8. View synthesis image quality analysis with white noise injection. (a) Café. (b) Doorflowers.



(a) Relationship between depth distortion and Q_{step}



(b) Relationship between depth bit rate and $1/Q_{\text{step}}$

Fig. 9. Relationship between Q_{step} and depth distortion/bit rate. (a) Relationship between depth distortion and Q_{step}. (b) Relationship between depth bit rate and $1/Q_{\text{step}}$.

Fig. 8. Similarly, $G_{\text{CSAD}}(I_{\text{T}})$ is the slope of these lines fitted from the points of different symbols with one fixed I_{T} and differentI_{S}s, shown as the dash line in Fig. 8. Additionally, the line fitted from star symbol indicates that noise is equally injected to the CTAD and CSAD, where the slope is denoted by G_{ALL}, shown as the solid line in Fig. 8. From the statistical experimental results, we have the following five observations.

1. The view synthesis distortion is generally in linear relation with depth distortion for CSAD, CTAD or entire image, respectively.
2. The slopes of CSAD, $G_{\text{CSAD}}(I_{\text{T}})$, are much smaller than those of CTAD regions, $G_{\text{CTAD}}(I_{\text{S}})$, which means the distortion in CTAD regions affects VVI quality more severely and higher priority should be given when bits are allocated.
3. The slopes of the linear function are different over regions and they usually increase as the color texture becomes complex.
4. For different I_{S}, the slopes $G_{\text{CTAD}}(I_{\text{S}})$ are almost the same. Similarly, the slopes $G_{\text{CSAD}}(I_{\text{T}})$ are almost the same for different I_{T}.
5. The slope of the linear function is different from sequence to sequence, which is video content dependent, e.g. color texture etc.

In addition to the white noise analyses, coding experiments are also performed to analyze the MSE_{D} effects to MSE_{VS} in terms of the quantization error effect. Similar results and relationship can be found. Therefore, CTAD and CSAD should be coded differently due to the fact that CSAD is more endurable to depth error. In the next section, we propose the new bit allocation and RDO algorithms for depth coding by utilizing this VSDM.

IV. VSDM Based RBA and RDO for MDVC

A. The D_{VS}-D_D-Q Model and R_D-Q Model

A number of works have been proposed in the literature [31], [32] for color video bit allocation in H.264/AVC. In [31], Kamaci *et al.* designed a Cauchy distribution based D-Q model as

$$D_D = \xi Q_{Step}^{\beta}. \tag{25}$$

Fig. 9(a) shows the relationship between depth distortion D_{D} and quantization step Q_{step}, in which the depth distortion is evaluated with MSE and it is in linear relationship with Q_{step} for different depth sequences. Therefore, an empirical linear D_{D}-Q model is proposed for H.264/AVC as [32]

$$D_D = \xi Q_{Step}, \tag{26}$$

where ξ is the model parameter. It is a special case of the Cauchy distribution based model with the parameter β equals to 1. For the depth video coding, the depth video content is regarded as the Y component of the color video and U, V components are set ot 128 when it is input into the video

encoder. Therefore, the traditional linear D-Q model for the color video coding is also suitable for the D_{D}-Q and it has also been proved in [24]. According to (26) and (22), we can obtain the regional VSDM of the view synthesis distortion and Q_{step}, i.e. D_{VS}-Q model, as

$$D_{VS,\psi} = A_{\psi}\xi Q_{Step,\psi} + B_{\psi}, \tag{27}$$

where $Q_{\mathrm{step},\psi}$ is the Q_{step} in region ψ, $\psi \in \{CTAD, CSAD\}$. Fig. 9(b) shows the relationship between the entropy of the depth bit rate and $1/Q_{\mathrm{step}}$. In the figure, the lines are linearly fitted from the points of each color for three test sequences with different properties. We can observe that the bit rate and $1/Q_{\mathrm{step}}$ have linear relationship for different sequences. Therefore, a linear R_{D}-Q model is modeled as

$$R_D = Km/Q_{Step} + C, \tag{28}$$

where m is the Mean Absolute Difference (MAD) of the residual between the original signal and predicted signal, K is the model parameter, C is a constant indicating the header bits. This R_{D}-Q model in (28) is valid in MB level and thus also valid for CTAD and CSAD respectively.

B. Optimal Group-of-MB Level Depth Bit Allocation Based on VSDM

Let N_{T} and N_{S} be the number of MBs in CTAD and CSAD in a frame, k_{T} and k_{S} are the MB indices in the coding order, $k_{\mathrm{T}} = 1, 2, 3 \ldots, N_{\mathrm{T}}$ and $k_{\mathrm{S}} = 1, 2, 3 \ldots, N_{\mathrm{S}}$. The optimization problem is to choose the optimal Q_{step} values for CTAD and CSAD, respectively, to minimize the total distortion of the picture subject to the total bit constraint R_{T}. The bit allocation problem for each frame is

$$\begin{cases} Q_T^* Q_S^* = \arg\min \left(\sum_{k_T=1}^{N_T} D_{CTAD,k_T}(Q_T) + \sum_{k_S=1}^{N_S} D_{CSAD,k_S}(Q_S) \right) \\ \sum_{k_T=1}^{N_T} B_{CTAD,k_T}(Q_T) + \sum_{k_S=1}^{N_S} B_{CSAD,k_S}(Q_S) < R_T \end{cases}, \tag{29}$$

where $D_{\mathrm{CTAD},k}()$ and $D_{\mathrm{CSAD,k}}()$ are the view synthesis distortion for the kth MB in CTAD and CSAD, respectively. $B_{\mathrm{CTAD},k}()$ and $B_{\mathrm{CSAD,k}}()$ indicate the buffer consumption of the encoding bits of the kth MB in CTAD and CSAD, respectively. Q_{T} and Q_{S} are Q_{step} for CTAD and CSAD, respectively. R_{T} is the buffer of the target bits. It is easy to prove that this optimization problem in (29) is convex. Therefore, to find the optimal quantization solution for (29), Largrangian multiplier is used to minimize the RD cost J, which can be expressed as

$$\begin{aligned} Q_T^* Q_S^* \lambda^* &= \arg\min J \\ &= \arg\min \begin{pmatrix} \sum_{k_T=1}^{N_T} D_{CTAD,k_T}(Q_T) \\ + \sum_{k_S=1}^{N_S} D_{CSAD,k_S}(Q_S) + \\ \lambda\Big(\sum_{k_T=1}^{N_T} B_{CTAD,k_T}(Q_T) \\ + \sum_{k_S=1}^{N_S} B_{CSAD,k_S}(Q_S) - R_T \Big) \end{pmatrix}. \end{aligned} \tag{30}$$

To get the optimal Q_T^*, Q_S^*, we compute the partial derivatives of J to Q_T, Q_S and λ, and set them zero, which are

$$\begin{cases} \frac{\partial J}{\partial Q_T} = \frac{\partial \sum_{k_T=1}^{N_T} D_{CTAD,k_T}(Q_T)}{\partial Q_T} \\ \qquad + \lambda \frac{\partial \sum_{k_T=1}^{N_T} B_{CTAD,k_T}(Q_T)}{\partial Q_T} = 0 \\ \frac{\partial J}{\partial Q_S} = \frac{\partial \sum_{k_S=1}^{N_S} D_{CSAD,k_S}(Q_T)}{\partial Q_S} \\ \qquad + \lambda \frac{\partial \sum_{k_S=1}^{N_S} B_{CSAD,k_S}(Q_S)}{\partial Q_S} = 0 \\ \frac{\partial J}{\partial \lambda} = \sum_{k_T=1}^{N_T} B_{CTAD,k_T}(Q_T) \\ \qquad + \sum_{k_S=1}^{N_S} B_{CSAD,k_S}(Q_S) - R_T = 0. \end{cases} \tag{31}$$

Apply (27) and (28) to (31) and solve (31), we can obtain the optimal Q_{step} for CTAD and CSAD, Q_T^* and Q_S^*, as

$$Q_T^* = K\sqrt{\bar{m}_T}\frac{N_T\sqrt{\bar{m}_T} + N_S\sqrt{\frac{A_{CSAD}}{A_{CTAD}}}\sqrt{\bar{m}_S}}{R_T - C_{CTAD} - C_{CSAD}}, \tag{32}$$

$$\begin{aligned} Q_S^* &= K\sqrt{\bar{m}_S}\frac{N_T\sqrt{\frac{A_{CTAD}}{A_{CSAD}}}\sqrt{\bar{m}_T} + N_S\sqrt{\bar{m}_S}}{R_T - C_{CTAD} - C_{CSAD}} \\ &= \sqrt{\frac{A_{CTAD}\bar{m}_S}{A_{CSAD}\bar{m}_T}}Q_T^*, \end{aligned} \tag{33}$$

where $\bar{m}_T$ and $\bar{m}_S$ are the average MAD for CTAD and CSAD regions, respectively, which is $\bar{m}_T = \frac{1}{N_T}\sum_{k_T=1}^{N_T} m_{CTAD}(k_T)$, $\bar{m}_S = \frac{1}{N_S}\sum_{k_S=1}^{N_S} m_{CSAD}(k_S)$, C_{CTAD} and C_{CSAD} are the total coding head bits for CTAD and CSAD, respectively.

C. RDO for View Synthesis Oriented Depth Video Coding

As the depth video is coded with traditional H.264/AVC based video coding standard as illumination component of color video, traditional R-D model can be applied to depth coding and it is represented as

$$R(D) = k\ln\left(\sigma^2/D\right), \tag{34}$$

where D is output distortion and σ^2 is the variance a picture. Taking the derivative of $R(D)$ with respect to D and setting its value to $-1/\lambda_{MODE}$ yields [33]

$$dR(D)/dD \equiv -1/\lambda_{MODE}. \tag{35}$$

By substituting (34) into (35), we can get the Lagrangian multiplier as

$$\lambda_{MODE} = D/k. \tag{36}$$

The reconstructed depth video from decoding is used to generate virtual view, and thus, the distortion of synthesized virtual view, D_{VS}, shall be actually taken into account in the new distortion model for depth encoders. However, as for the bit rate, compressed color and depth bit rates are actually transmitted, that is total bit rate $R_{\mathrm{T}}(D_{\mathrm{VS}})$ equals to the sum of depth bit rate $R(D_{\mathrm{D}})$ and color texture bit rate R_{C}. In this paper, we assume that color and depth videos are separately encoded, and R_{C} is a constant which is independent of D_{D} for the depth encoder. Therefore, the $R_{\mathrm{T}}(D_{\mathrm{VS}})$ can be formulated as

$$R_T(D_{VS}) = R(D_D) + R_C = k_D \ln\left(\sigma_D^2/D_D\right) + R_C, \tag{37}$$

where k_{D}, D_{D} and σ_{D}^2 are weighted coefficient, output distortion and input variance for depth video, respectively. Similarly, to calculate the Lagrangian factor for view synthesis based RD model (λ_{MODE}^{VS}), we take the derivative of $R_{\mathrm{T}}(D_{\mathrm{VS}})$ with respect to D_{VS} and set it to $-1/\lambda_{MODE}^{VS}$. Thereafter, applying (22), we have

$$\begin{aligned}\frac{dR_T(D_{VS})}{dD_{VS}} &= \frac{d(R(D_D)+R_C)}{dD_{VS}} \\ &= \frac{dR(D_D)}{d(A_{ALL}D_D + B_{ALL})} \equiv -\frac{1}{\lambda_{MODE}^{VS}}.\end{aligned} \tag{38}$$

By solving (38), we can obtain

$$\lambda_{MODE}^{VS} = A_{ALL}D_D/k_D. \tag{39}$$

When the depth video is coded by the H.264/AVC based video codec, the D and k in (36) equal to D_{D} and k_{D}, respectively. Then, (36) is substituted into (39), and the VSDM based Lagrangian multiplier for mode decision and reference frame selection is

$$\lambda_{MODE}^{VS} = A_{ALL}\lambda_{MODE}. \tag{40}$$

Also, the VSDM based Lagrangian multiplier for motion/disparity estimation is

$$\lambda_{MOTION}^{VS} = \sqrt{A_{ALL}}\lambda_{MOTION}, \tag{41}$$

where λ_{MODE} and λ_{MOTION} are Lagrangian multiplier for mode decision and motion/disparity estimation in traditional H.264 based video codec, respectively. Based on RDO theory, the goal for depth coding optimization is to minimize the new R_{D} and D_{VS} cost function as below

$$\begin{aligned}\min J_{VS}, \quad J_{VS} &= D_{VS} + \lambda^{VS}R_D \\ &= A_\psi D_D + A_{ALL}\lambda R_D + B_\psi,\end{aligned} \tag{42}$$

where $\psi \in \{CTAD, CSAD\}$, B_ψ is a constant. Since we also intend to use the H.264/AVC based codec to encode the depth video, the distortion part is also evaluated with depth distortion D_{D}. Thus, (42) can be rewritten as

$$\min J, \ J = D_D + \left(A_{ALL}/A_\psi\right)\lambda R_D. \tag{43}$$

The (43) indicates that the VSDM based Lagrangian multiplier is $(A_{\mathrm{ALL}}/A_\psi)\lambda$ when we use the H.264/AVC based video codec to encode the MDV.

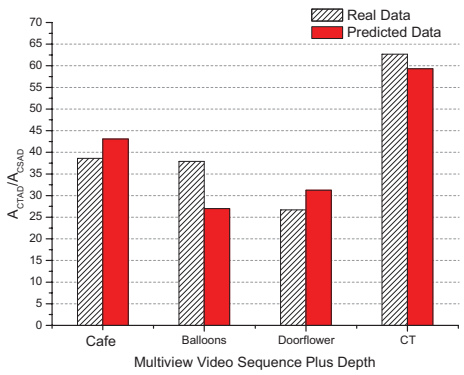


Fig. 10. Comparison on the real and predictive $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$ for different MDV sequences.

D. VSDM Based MDVC and Model Parameter Estimation

To calculate the optimal Q_{step} in (33) in the coding process, we need to calculate $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$ first. To obtain the $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$, the view synthesis distortion can be measured from the difference between rendered image by distorted depth and that by the original depth. However, the rendering process requires to be performed repeatedly by dozens of times, which is of extremely high complexity. To tackle this problem, we proposed a model parameter estimation strategy for VSDM. Since $A_\psi = \beta_\psi \sum_{\varphi\in\{L,R\}} \omega_\varphi^2 \cdot k_\varphi^2$, $\psi \in \{CTAD, CSAD\}$ and the right part $\sum_{\varphi\in\{L,R\}} \omega_\varphi^2 \cdot k_\varphi^2$ is constant for a given MCV and MDV, we can calculate $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$ as

$$\frac{A_{CTAD}}{A_{CSAD}} = \frac{\beta_{CTAD}}{\beta_{CSAD}}, \tag{44}$$

where β_{CTAD} and β_{CSAD} can be calculated by applying linear regression to the procedure presented in subsection III.C. Based on (24) and (44), parameters $A_{\mathrm{ALL}}/A_{\mathrm{CTAD}}$ and $A_{\mathrm{ALL}}/A_{\mathrm{CSAD}}$ can be presented as

$$\frac{A_{ALL}}{A_{CTAD}} = \mu\frac{\beta_{CSAD}}{\beta_{CTAD}} + 1 - \mu, \tag{45}$$

$$\frac{A_{ALL}}{A_{CSAD}} = \mu + (1-\mu)\frac{\beta_{CTAD}}{\beta_{CSAD}}. \tag{46}$$

Since the image texture of the multiview video is usually stable and it does not change much within a GOP, the model parameters β_{CTAD} and β_{CSAD} are updated for every GOP in order to achieve a good balance point between the computation complexity and the estimation accuracy.

To verify the effectiveness and accuracy of the prediction with (44), we collect $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$ from the statistical analyses in section III.E. This collected $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$ is regarded as the real data. The predictive $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$ is predicted by (44). Fig. 10 shows the comparison of the real and predictive $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$ for the 3D video sequences with different resolution, motion properties and camera settings, etc. Though there are still some small mismatches for the two values of $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$, which is probably caused by the fitting error in data processing, occlusion/disclosure regions and the hole filling process in the view synthesis. We can observe that the predicted $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$ is close to the

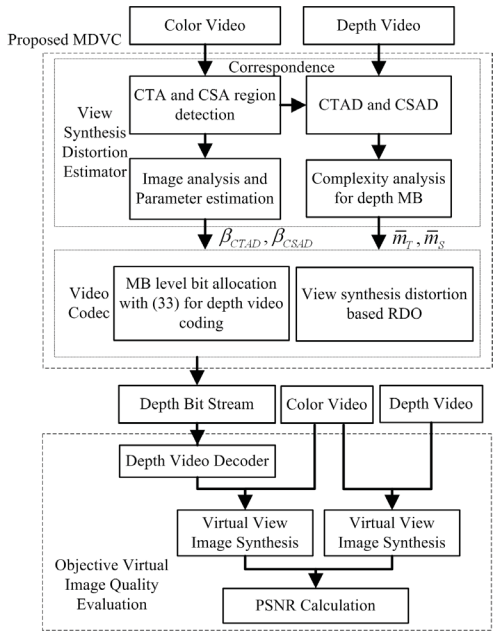


Fig. 11. Flow of the proposed VSDM based MDVC algorithm and its VVI quality evaluation.

real $A_{\mathrm{CTAD}}/A_{\mathrm{CSAD}}$ which varies significantly over different sequences in general.

Fig. 11 shows the flow of the proposed VSDM based MDVC algorithm and VVI quality evaluation, where the proposed MDVC can be divided into two parts, the image analysis part (bottom) and video coding part (top). The basic coding steps of the proposed MDVC can be described as

Step 1. CTA and CSA region detection. Apply an edge detection algorithm, e.g. Canny operator, in color video image, and label one MB as CTA region if the number of edge pixels in the MB is larger than N_{e}, otherwise, label the MB as CSA region.

Step 2. Label MB in the depth video as CTAD if its corresponding MB in color video is CTA, otherwise, label it as CSAD.

Step 3. If current frame is the anchor frame, perform color image analyses and calculate/update model parameter β_{CTAD} and β_{CSAD}, otherwise use previous obtained β_{CTAD} and β_{CSAD} for (33)

Step 4. Complexity analyses of the depth video. In this paper, we use INTER16x16 mode to pre-analyze the complexities of depth content and get $\bar{m}_T$ and $\bar{m}_S$ for the current depth frame for high accuracy.

Step. 5. Calculate the optimal Q_{T} and Q_{S} for MB level bit allocation and update the Lagrangian multiplier for view synthesis distortion based RDO

Step. 6. Encode the current frame.

Step. 7. Go to Step.1 for the next frame.

The bottom dash rectangle is the VVI quality evaluation part for the proposed MDVC, where the objective PSNR are calculated by comparing the VVIs rendered by color video plus the reconstructed depth with the VVIs rendered by color video plus the original depth.

V. Experimental Results and Analyses

A. Coding Experiments and Analyses

The recent H.264/AVC based MVC reference software JMVC 8.0 [34] was used to evaluate the proposed depth coding algorithms. Fast ME/DE is enabled and their search ranges are ± 64. The number of bi-prediction iteration is 4 and search range for the iterations is 8. The maximum number of reference frames is 2 for each memory list and the GOP length is 12. Nine MDV test sequences, including Ballet, Breakdancers, Kendo, Balloons, Cafe, Champ.Tower, Pantomime, Dog and Doorflowers, with different motion properties and camera settings (parallel and toed-in, different baselines), are used and 8 GOPs for each sequence are encoded. Two depth videos were encoded by the depth coding algorithms and one intermediate view at the middle position of the two views was rendered. The middle view image that rendered by the original color and depth videos was used as reference for measuring VVI quality. In these sequences, the depth videos of Café, Ballet, Breakdancers, Kendo and Balloons are available, the rest of depth sequences were generated by Depth Estimation Reference Software, DERS 3.0 [35]. View Synthesis Reference Software, VSRS 3.0 [36] was adopted for view synthesis. Basis Quantization Parameters (QPs) were set as 12, 16, 20, 24, 28 and 32. In the experiment, the original color video was used to synthesize the virtual views and thus only the depth videos were encoded to analyze the depth compression effects to the view synthesis. Other settings and evaluation metrics are also analyzed in the next subsection.

Six schemes were implemented, including the original JMVC, Lee's scheme [17] (denoted by LeeCSVT), Kim's scheme [25] (denoted by KimICIP), the proposed VSDM based bit allocation scheme, denoted as 'RBA', the proposed VSDM based RDO, denoted as 'RDO', and the proposed overall algorithm as 'ALL' where both the proposed RBA and RDO are both enabled. In Lee's scheme, color video was compressed with the original JMVC before depth video coding and QP was 12, 16, 20, 24, 28 and 32. Then, the reconstructed color video was used in threshold calculation [17] for the corresponding depth video coding. But in view synthesis, the original color video was used for fair comparison, which is the same as JMVC and our proposed schemes. For Kim's scheme, its new distortion metric and related RD optimization is implemented for comparison.

For the image quality evaluation, average PSNR of VVI is used to measure the depth image quality. The PSNR of synthesized image is calculated as

$$PSNR_{VS,\chi} = 10\log_{10}\left(255^2/MSE_{VS,\chi}\right), \tag{47}$$

$$MSE_{VS,\chi} = \frac{1}{MN}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\left|V_{\chi}(i,j) - V_{O}(i,j)\right|^2, \tag{48}$$

where $V_O(i,\ j)$ is VVI pixel at $(i,\ j)$ generated by the original color and the original depth videos. $V_{\chi}(i,\ j)$ is VVI pixel at $(i,\ j)$ generated by the original/reconstructed color video and reconstructed depth video, χ indicates different coding schemes, $\chi \in$ {JMVC, LeeCSVT, KimICIP,

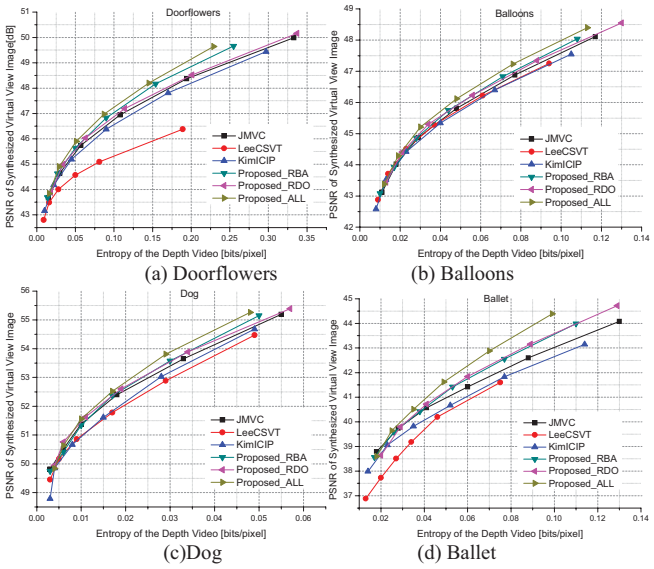


Fig. 12. RD curves of JMVC, Lee's scheme, Kim's scheme and proposed algorithms. (a) Doorflowers. (b) Balloons. (c) Dog. (d) Ballet.

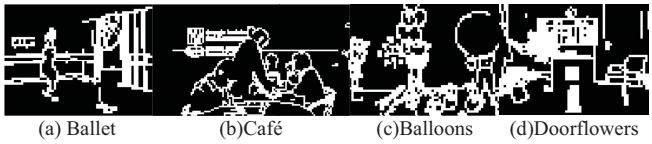


Fig. 13. Example of CSAD and CTAD segmentations. (a) Ballet. (b) Café. (c) Balloons. (d) Doorflowers.

RBA, RDO, ALL}, M and N are the width and the height of VVI, respectively.

Fig. 12 shows the RD performance comparison among the original JMVC, Lee's scheme and the proposed schemes including VSDM based RBA, VSDM based RDO, and the overall algorithm for four selected sequences. The y-axis is the average PSNR of VVIs against the VVI generated from the original depth, i.e. $PSNR_{VS,\chi}$, and x-axis is the entropy of total encoded depth. From the experimental results, we can observe that Lee's scheme reduces the depth bit rate significantly. However, the VVI quality also degrades significantly due to the SKIP mode. Lee's scheme may be suitable for the low bit rate in which the color video is also compressed with the same QP as the depth video. But for the case that the color video is coded with different QPs, it can not achieve acceptable RD performance. From our observations, the main reasons are: 1) a hard threshold was utilized to exploit the temporal and inter-view correlation, and in the final view synthesis stage, virtual view was rendered based on the reconstructed color video and reconstructed depth video, which were coded with the same QP. However, in this paper, only the depth distortion impacts on the virtual view is analyzed, and the original color video is used for view synthesis. In this case, the hard threshold is not effective; 2) The VVI distortion can not be hidden in the quantization error of color video when the original color video is used in the view synthesis; 3) Lee's scheme may cause error propagation that affected the rendering quality for high level B pictures. For Kim's scheme, we can observe that it superiors to Lee's scheme but inferior to the original JMVC for most cases. That because the parameters in Kim's model are not adaptive to different content characteristics in sequences and thus the overall performance degrades.

For the proposed VSDM based RBA algorithm, it assigned fewer bits on the CSAD regions for bit rates saving, since they are not as important as CTAD regions for the quality of the rendered images. Fig. 13 depicts an example of the CTAD and CSAD segmentations of the four test sequences. As shown in Fig. 13, the dark pixels are CSAD and bright pixels are CTAD. We can observe that the CTAD covers the most of the texture area of the color image, and CSAD covers the smooth area. Based on this segmentation, the proposed RBA algorithm maintains almost the same PSNR while the bit rate is saved significantly. For the proposed RDO, we can observe that it also improves the coding efficiency for all the test sequences. Since the depth distortion in CTAD regions is regarded more important, fine motion estimation and smaller data partition are used in mode decision while the Lagrangian multiplier is adjusted to be smaller. Additionally, for the proposed overall algorithm which integrates the VSDM based RBA and RDO, it can not only achieve more bit rate saving, but also improves VVI quality. This indicates that the performance improvement from RBA and RDO are additive. The proposed algorithms act as relative bit allocation scheme which gives higher priority to CTAD and lower priority to CSAD. They mainly exploit the spatial redundancies and can be applied to both INTRA and ITNER (B and P) frames.

The Bjontegaard Delta Bit Rate (BDBR) and Bjontegaard Delta PSNR (BDPSNR) [37] are also utilized to measure the overall improvement of the proposed algorithm. First column of Table I, labeled as UC_D, shows the BDBR and BDPSNR among JMVC, LeeCSVT, KimICIP and the proposed overall algorithm at high and low bit rate, where high bit rate means $QP \in \{12, 16, 20, 24\}$ and low bit rate means $QP \in \{20, 24, 28, 32\}$. Negative BDBR indicates percentage of bit rate reduction and positive BDPSNR indicates quality increase compared to JMVC. We observe that Lee's scheme is inferior to the original JMVC for most sequences. For Champ.Tower and Café, the BDBR at high bit rate is not available (labeled as 'NA') because the fit algorithm is not suitable for the situation that bit rate difference is too large. For Pantomime, Lee's scheme is better than the JMVC, which achieves 27.95% and 8.42% bit rate saving for high and low bit rate, respectively. For Kim's scheme, it is inferior to the original JMVC for most sequences. Only for the Café sequence, Kim's scheme can achieve 6.54% and 9.26% bit rate reduction on the depth component, or 0.13 dB and 0.20 dB PSNR increase at high and low bit rate, respectively. For the proposed MDVC algorithm, the BDBR reduction of high bit rate is from 6.54% to 59.06% and 25.13% on average, meanwhile, the BDBR reduction of low bit rate is from 8.03% to 58.53% and 21.67% on average. It means that 25.13% and 21.67% bit rate can be saved on the depth component while maintaining the same VVI quality at high and low bit rate, respectively. On the other hand, comparing with the

original JMVC, the average BDPSNR gain achieved by the proposed MDVC are 0.75 dB and 0.41 dB at the high and low bit rate, respectively. Therefore, the proposed MDVC significantly improves the depth coding efficiency. In fact, the proposed MDVC can achieve better compression efficiency for the depth sequences that have relative large noise from depth generation, e.g. Champ.Tower and Pantomime which are generated by unsupervised DERS. Also, it has relative small improvements for the depth sequences that have small noise, e.g. Café and Balloons, which are generated from depth capturing and supervised approach. Overall, the experimental results have proved that the proposed VSDM based RBA and RDO improve the depth coding efficiency significantly for the test sequences.

In terms of the computational complexity, the proposed overall algorithm increases from 3.61% to 26.28%, 13.67% on average compared with the original JMVC due to additional computations in the MAD pre-analysis of using INTER16 × 16 mode and parameter estimation for $\beta_{\mathrm{CTAD}}/\beta_{\mathrm{CSAD}}$.

B. Analyses and Discussions on RD Performance Evaluation Metrics for Depth Coding

In the above utilized evaluation metric, uncompressed color video and reconstructed depth video are used to synthesize the VVI, and the depth bits only are counted in the x-axis. This metric is denoted as UC_D (Uncompressed Color, Depth bits only). In addition to this metric, another four metrics are also utilized to extensively analyze the RD performance of the benchmark schemes and the proposed algorithm. Color and depth videos are separately compressed with QP_{C} and QP_{D}, where color video is encoded with the original JMVC and depth video is encoded with different coding schemes. QP_{C}, $QP_{\mathrm{D}} \in \{12, 16, 20, 24, 28, 32\}$. Their reconstructed color and depth from decoding are used to synthesize VVIs. Based on this setting, we derive four metrics:

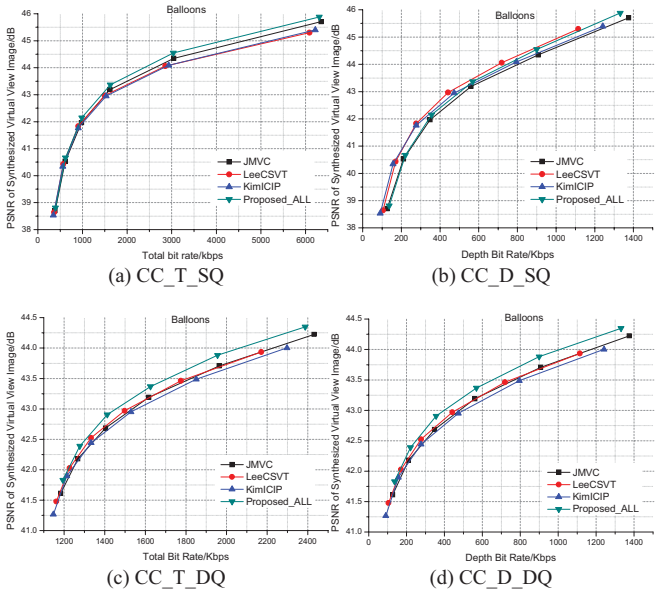
a) Reconstructed color and depth video input to view synthesis are encoded with the same QP, i.e. $QP_{\mathrm{C}} = QP_{\mathrm{D}}$, and total bits are counted in x-axis, denoted by CC_T_SQ (Compressed Color, Total bits, Same QP).

b) $QP_{\mathrm{C}} = QP_{\mathrm{D}}$, but only depth bits are counted in x-axis, denoted by CC_D_SQ (Compressed Color, Depth bits only, Same QP).

c) Since the QP_{C} is not always equal to QP_{D} in the 3D video applications, the coding situation that color and depth are coded with different QPs. Here, QP_{C} is set as a constant value, i.e. 20, and QP_{D} varies from 12 to 32, and total bits are counted in x-axis, denoted by CC_T_DQ (Compressed Color, Total bits, Different QP).

d) QP_{C} is 20, QP_{D} varies from 12 to 32, and only depth bits are counted in x-axis, denoted by CC_D_DQ. (Compressed Color, Depth bits only, Different QP).

According to the analyses in [20], it is found that if the VVI is rendered by compressed color and depth video, the rendered distortion in y-axis depends on the color and depth distortion. Thus, it is required to take color and depth bits, i.e. total bits, rather than depth bits only in x-axis. Therefore,



(a) CC_T_SQ (b) CC_D_SQ

(c) CC_T_DQ (d) CC_D_DQ

Fig. 14. RD comparison of using different evaluation metrics (Balloons). (a) CC_T_SQ. (b) CC_D_SQ. (c) CC_T_DQ. (d) CC_D_DQ.

metrics CC_T_SQ and CC_T_DQ are reasonable evaluation schemes, but CC_D_SQ and CC_ D_DQ are not. For UC_D metric, since the rendered distortion in y-axis depends on the depth distortion only, it is reasonable to use the depth bits only in x-axis.

All these five metrics are utilized to evaluate the RD performance of JMVC, LeeCSVT, KimICIP and the proposed overall scheme. Fig. 14 shows the RD comparison of using the four different evaluation metrics for Balloons sequence, and its RD performance evaluated with UC_D is illustrated in Fig. 12(b). We can observe that the proposed overall scheme is better than JMVC for all these five metrics. On the other hand, the proposed overall scheme is the best one among the compared schemes, except the case that their RDs are evaluated with CC_D_SQ. However, CC_D_SQ is regarded as an improper metric and can hardly reflect the true improvements. For CC_T_DQ and CC_D_DQ, though the shape of the RD curves are the same, the bit rate saving is not the same because base bits are larger by counting total bits in x-axis. Table I shows the BDPSNR and BDBR comparisons using the five metrics for all the nine test sequences. In the table, CC_D_SQ and CC_D_DQ are regarded as improper schemes for RD evaluation according to analyses in [20], which are labeled as '$*$'. Label '#' indicates that data labeled as 'NA' is not take into account in calculating the average value. For CC_T_SQ metric, the proposed overall scheme reduces 9.99% and 14.51% total bit rate on average at low and high bit rate, respectively. It improves 0.22 dB and 0.24 dB of the VVI quality in terms of BDPSNR. For CC_T_DQ metric, the proposed algorithm reduces total bit rate 6.71% and 12.77% on average at high and low bit rate. When measured with BDPSNR, it achieves 0.30 dB and 0.42 dB gain. Except for the CC_D_SQ and CC_D_DQ which are not reasonable [20], the proposed overall scheme is also the best one among the compared schemes for the test sequences. On the other hand,

TABLE I

BDBR AND BDPSNR COMPARISONS AMONG LEECSVT, KIMICIP AND PROPOSED SCHEME WHEN USING DIFFERENT METRICS (UNIT: %/DB)

Schemes			UC_D	CC_T_SQ	CC_D_SQ*	CC_T_DQ	CC_D_DQ*
LeeCSVT VS JMVC	Balloons	Low	−1.46/0.03	−2.85/0.09	−19.16/0.53	−0.99/0.07	−6.10/0.06
		High	4.55/−0.09	7.29/−0.14	−14.50/0.37	−0.23/0.01	−1.59/0.02
	Kendo	Low	46.95/−0.79	−3.01/0.13	−19.31/0.69	5.75/−0.26	3.73/−0.03
		High	58.21/−1.35	5.40/−0.11	−13.18/0.43	2.97/−0.06	8.00/−0.06
	Doorflowers	Low	68.39/−0.69	−0.84/0.09	−50.22/0.84	22.42/−0.26	42.02/−0.27
		High	61.35/−1.41	17.97/−0.69	−21.45/0.68	37.03/−0.52	53.34/−0.51
	Dog	Low	24.65/−0.35	0.98/−0.03	−5.41/0.19	1.59/−0.08	25.42/−0.08
		High	20.62/−0.42	4.23/−0.09	−7.30/0.07	3.29/−0.06	23.09/−0.07
	Champ. Tower	Low	47.18/−0.26	51.46/−0.33	15.77/−0.07	23.73/−0.11	45.51/−0.23
		High	NA/−1.34	NA/−1.10	NA/−1.14	NA/−1.24	NA/−1.25
	Pantomime	Low	−8.42/0.07	−2.92/0.05	−14.01/0.18	1.08/0.00	12.67/−0.02
		High	−27.95/0.65	−7.48/0.59	−29.41/0.05	−10.21/0.04	−19.31/0.04
	Cafe	Low	80.13/−1.02	45.44/−0.67	−5.69/0.24	12.36/−0.65	92.46/−0.67
		High	NA/−1.64	NA/−1.37	NA/−0.35	NA/−0.74	NA/−0.96
	Break.	Low	108.61/−1.21	127.88/−1.37	37.40/−0.81	9.22/−1.16	72.09/−1.17
		High	48.20/−1.06	124.20/−1.58	30.71/−0.83	11.55/−1.03	62.58/−1.03
	Ballet	Low	80.82/−1.16	167.18/−1.31	24.48/−0.57	6.57/−0.75	37.62/−0.75
		High	84.98/−1.04	139.12/−1.70	19.85/−0.39	12.07/−1.01	38.60/−0.63
	Average	**Low**	49.65/−0.60	42.59/−0.37	−4.02/0.14	9.08/−0.36	36.16/−0.35
		High[#]	35.71/−0.86	41.53/−0.69	−5.04/−0.12	8.07/−0.51	23.53/−0.49
KimICIP VS JMVC	Balloons	Low	−2.18/0.04	−1.80/0.07	−21.90/0.62	0.24/−0.01	−0.66/0.01
		High	5.90/−0.13	7.68/−0.16	−6.43/0.18	2.08/−0.06	5.00/−0.06
	Kendo	Low	14.42/−0.48	−2.60/0.11	−22.60/0.83	10.87/−0.64	8.97/−0.12
		High	10.25/−0.38	4.18/−0.10	−7.66/0.26	3.44/−0.08	10.62/−0.09
	Doorflowers	Low	5.93/0.10	−6.74/0.17	−23.32/0.49	1.25/−0.04	3.10/−0.02
		High	7.74/−0.19	2.31/−0.04	−9.51/0.24	3.26/−0.05	6.34/−0.05
	Dog	Low	25.18/−0.39	−0.41/0.02	−23.36/0.76	2.22/−0.04	35.17/−0.08
		High	13.19/−0.31	3.10/−0.07	−14.60/0.10	2.08/−0.04	15.66/−0.05
	Champ. Tower	Low	−3.42/0.03	3.24/−0.05	−10.18/0.15	0.52/−0.40	−3.09/0.04
		High	4.79/−0.08	9.30/−0.14	2.61/−0.04	2.75/−0.07	5.21/−0.07
	Pantomime	Low	45.52/−0.31	−0.22/0.01	−16.52/0.23	16.46/−0.09	132.74/−0.10
		High	25.94/−0.43	9.67/−0.06	−0.83/0.01	25.46/−0.08	79.98/−0.08
	Cafe	Low	−9.00/0.18	1.33/−0.03	−13.83/0.47	0.01/−0.02	0.88/−0.01
		High	−4.88/0.10	10.28/−0.13	−7.29/0.22	1.23/−0.03	2.91/−0.02
	Break.	Low	50.02/−1.48	29.12/−0.52	4.20/−0.11	2.16/−0.33	21.37/−0.39
		High	32.79/−1.03	31.20/−0.46	2.99/−0.09	2.88/−0.22	12.08/−0.22
	Ballet	Low	12.85/−0.32	12.05/−0.21	−2.70/0.08	0.59/−0.05	0.56/−0.01
		High	13.28/−0.37	36.85/−0.44	2.77/−0.08	2.78/−0.14	6.10/−0.14
	Average	**Low**	15.48/−0.29	3.77/−0.05	−14.47/0.39	3.81/−0.18	22.12/−0.08
		High	12.11/−0.31	12.73/−0.18	−4.22/0.09	5.11/−0.09	15.99/−0.09
Proposed ALL VS JMVC	Balloons	Low	−11.16/0.22	−3.50/0.11	−0.72/0.02	−3.36/0.17	−16.15/0.17
		High	−12.97/0.33	−9.57/0.19	−6.90/0.18	−6.38/0.18	−18.30/0.18
	Kendo	Low	−14.79/0.47	−4.88/0.18	−8.31/0.31	−1.60/0.04	−41.36/0.23
		High	−14.86/0.59	−6.07/0.13	−11.68/0.39	−8.30/0.14	−25.50/0.14
	Doorflowers	Low	−16.44/0.32	−7.68/0.19	−11.11/0.28	−7.80/0.22	−30.72/0.21
		High	−20.85/0.61	−12.34/0.22	−31.90/0.89	−18.07/0.24	−37.50/0.24
	Dog	Low	−8.03/0.13	−0.63/0.02	−0.20/0.00	−0.80/0.04	−10.47/0.04
		High	−14.25/0.35	−2.28/0.05	−10.31/0.15	−2.80/0.05	−18.84/0.05
	Champ. Tower	Low	−41.07/0.59	−21.83/0.42	−36.99/0.64	−12.99/0.65	−42.80/0.56
		High	−43.66/1.00	−26.49/0.48	−43.53/1.04	−26.57/0.98	−44.81/0.89
	Pantomime	Low	−58.53/0.80	−14.60/0.24	−35.62/0.55	−19.73/0.15	−73.05/0.15
		High	−59.06/1.86	−24.10/0.18	−51.28/0.56	−24.23/0.23	−57.63/0.15
	Cafe	Low	−9.26/0.20	−4.05/0.10	3.32/−0.14	−2.08/0.15	−12.40/0.16
		High	−6.54/0.13	−7.56/0.11	−1.56/0.04	−2.99/0.09	−10.14/0.10
	Break.	Low	−26.64/0.67	−14.55/0.31	−25.52/0.81	−4.47/0.65	−28.31/0.64
		High	−28.97/0.90	−10.76/0.18	−28.90/1.08	−8.91/0.68	−30.16/0.66
	Ballet	Low	−9.15/0.31	−18.19/0.40	−19.81/0.63	−7.54/0.59	−21.59/0.60
		High	−24.99/1.01	−31.39/0.58	−30.57/1.28	−16.70/1.19	−32.22/1.10
	Average	**Low**	−21.67/0.41	−9.99/0.22	−15.00/0.34	−6.71/0.30	−30.76/0.31
		High	−25.13/0.75	−14.51/0.24	−24.07/0.62	−12.77/0.42	−30.57/0.39

the bit rate saving and PSNR gain values of using UC_D, CC_T_SQ and CC_T_DQ are different because of different view synthesis settings and base bits. However, their values are generally in direct proportion, which means a better scheme indicated by one metric will generally be proved to be better by another.

Generally, CC_T_SQ and CC_ T_DQ metrics can be used to evaluate both color and depth optimization. However, a minor problem is when they are applied to individual color or depth optimization evaluation, different combinations of QP_C and QP_D in view synthesis will influence the evaluation result. For UC_D, it reflects the coding optimization on depth component and can also be utilized to evaluate depth optimization algorithm.

C. Subjective Test for the VVI

In addition, the subjective visual test is implemented to measure the virtual view video clip consists of VVIs. Tested VVIs were rendered by the original color and the distorted depth maps from different depth coding algorithms. In addition to the four coding algorithms, i.e. JMVC, LeeCSVT, KimICIP and proposed overall scheme, the VVI rendered by the original color and depth map was also compared, which is regarded as ground truth [20]. Since five schemes were tested, there were ten comparison pairs for each sequence. Nine different test sequences were used and each clip has 97 frames

The Stimulus Comparison Adjectival Categorical Judgment (SCACJ) method [38] was used. Experimental environment meets the requirements in ITU-R BT.500-11 [38]. Two VVI clips are displayed simultaneously in a screen, and after playback subject is asked to give his/her opinions of which is perceptually better based on the overall quality comparisons. Seven options are provided to indicate the quality of the left image compared to the right image, where values $\{-3, -2, -1, 0, 1, 2, 3\}$ represent "much worse", "worse", "slightly worse", "same", "slightly better", "better" and "much better", respectively. The MSU perceptual video quality tool [39] was used to control the experiment flow, where the ten comparison pairs were compared in a random order. The display is 23 inch Dell E2311H with resolution 1920×1088.

Ten volunteers recruited on campus participated the subjective testing, whose ages range from 24 to 30 with a mean of 27. All subjects of the experiment meet the minimum criteria of acuity of 20:30 vision and pass the color vision test. All the subjects are non-experts, who have no experience of video quality assessment in recent three months. Examples and instructions were given before the visual testing.

Fig. 15 shows the Mean Option Score (MOS) for the subjective VVI evaluation, where the higher MOS means the corresponding clip have relative higher visual quality. We can observe that 1) For the sequences such as Café, Dog, Doorflowers and Kendo, the MOSs are quite close for the five schemes, which means that the subjects can hardly tell the difference among them. 2) Except Ballet and Breakdancers, the ground truth might not be the best one. 3) According to the average MOS, the proposed algorithm is a little bit better than LeeCSVT and KimICIP, and a little bit inferior to the JMVC.

The main reasons for the above observations are: A) The qualities of VVIs rendered by different depth maps are quite high, which are larger than 42 dB compared to the ground truth, as shown in Fig. 12. Thus, human eye can hardly distinguish the visual difference for these sequences. Similar

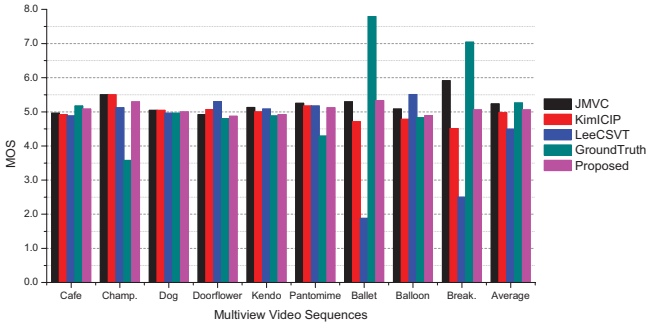


Fig. 15. MOS of the subjective VVI evaluation.

MOS indicates that there is almost no difference among the tested schemes. B) In objective evaluation, the VVI rendered by original color and depth video is used as the ground truth. However, in subjective evaluation, this ground truth might not have the best quality due to the synthesis distortion from the view synthesis tool. People usually evaluate the VVI quality with their own common sense and subconsciously use the original view image as the ground truth in their minds. There is a mismatch between the ground truths in objective and subjective evaluation. The proposed algorithm is based on the assumption that the ground truth is of the best quality and tries to keep its VVI consistence with the ground truth as much as possible. Thus, the proposed algorithm will be failed if the ground truth is not the best in the subjective evaluation. C) The synthesis distortion from the view synthesis tools has big impacts on the VVI quality. Thus, the depth distortion effects to the VVI are concealed by the synthesis distortion, which may make the subjective improvement of the proposed algorithm inconspicuous.

According to the subjective experiment, we can find that the subjective scores of the proposed algorithm are close to that of the original sequences in most cases, which confirms the effectiveness of the proposed algorithm. On the other hand, we also find that there is a disparity between the PSNR measure and the human subjective measure for the VVI evaluation. In addition, there is a disparity between the ground truths in the objective and subjective measurements, which also causes some inconsistency between the subjective and objective results. Since currently there is no effective quality metrics developed for 3D video, it is reasonable to use PSNR to assess the video quality in the coding optimization framework. The proposed solution is applicable if such an effective 3D quality metric has been developed.

VI. Conclusion

In this paper, we proposed a new View Synthesis Distortion Model (VSDM) in which the relationship between depth distortion and view synthesis distortion has been theoretically established for different types of regions, i.e. CTAD and CSAD. Generally, it has a linear relationship between the depth distortion and the view synthesis distortion for CTAD, CSAD and entire image, respectively. Furthermore, the depth distortion in the CTAD regions has more important impacts on the view synthesis distortion than that in CSAD regions. Based on this VSDM, we proposed regional bit allocation and

rate-distortion optimization schemes for multiview depth video coding by allocating relative more bits on CTAD for high rendering quality and fewer bits on CSAD high compression ratio. The experimental results via five different evaluation metrics have proved that the proposed algorithms can improve the depth coding efficiency significantly.

REFERENCES

[1] K. Müller, P. Merkle, and T. Wiegand, "3-D video representation using depth maps," *Proc. IEEE*, vol. 99, no. 4, pp. 643–656, Apr. 2011.

[2] 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.

[3] K. J. Oh, A. Vetro, and Y. S. Ho, "Depth coding using a boundary reconstruction filter for 3-D video systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 3, pp. 350–359, Mar. 2011.

[4] K. J. Oh, S. Yea, A. Vetro, and Y. S. Ho, "Depth reconstruction filter and down/up sampling for depth coding in 3-D video," *IEEE Signal Process. Lett.*, vol. 16, no. 9, pp. 747–750, Sep. 2009.

[5] E. Ekmekcioglu, M. Mrak, S. Worrall, and A. Kondoz, "Utilization of edge adaptive upsampling in compression of depth map videos for enhanced free-viewpoint rendering," in *Proc. IEEE Int. Conf. Image Process.*, Nov. 2009, pp. 733–736.

[6] G. Tech, K. Muller, and T. Wiegand, "Diffusion filtering of depth maps in stereo video coding," in *Proc. Picture Coding Symp.*, Dec. 2010, pp. 306–309.

[7] J. Choi, D. Min, D. Kim, and K. Sohn, "3D JBU based depth video filtering for temporal fluctuation reduction," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 2777–2780.

[8] D. Min, J. Lu, and M. Do, "Depth video enhancement based on weighted mode filtering," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1176–1190, Mar. 2012.

[9] E. Ekmekcioglu, V. Velisavljević, and S. T. Worrall, "Content adaptive enhancement of multi-view depth maps for free viewpoint video," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 2, pp. 352–361, Apr. 2011.

[10] Y. Zhao, C. Zhu, Z. Chen, D. Tian, and L. Yu, "Boundary artifact reduction in view synthesis of 3D video: From perspective of texture-depth alignment," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 510–522, Jun. 2011.

[11] Y. Zhao, C. Zhu, Z. Chen, and L. Yu, "Depth no-synthesis-error model for view synthesis in 3-D video," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2221–2228, Aug. 2011.

[12] D. D. Silva, W. Fernando, S. T. Worrall, H. K. Arachchi, and A. Kondoz, "Sensitivity analysis of the human visual system for depth cues in stereoscopic 3-D displays," *IEEE Trans. Multimedia*, vol. 13, no. 3, pp. 498–506, Jun. 2011.

[13] D. D. Silva, E. Ekmekcioglu, W. 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] S. T. Na, K. J. Oh, C. Lee, and Y. S. Ho, "Multi-view depth video coding using depth view synthesis," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2008, pp. 1400–1403.

[15] H. A. Karim, N. S. Mohamad, A. Shah, N. M. Arif, A. Sali, and S. Worrall, "Reduced resolution depth coding for stereoscopic 3D video," *IEEE Trans. Consumer Electron.*, vol. 56, no. 3, pp. 1705–1712, Aug. 2010.

[16] Y. Liu, Q. Huang, S. Ma, D. Zhao, W. Gao, S. Ci, and H. Tang, "A novel rate control technique for multiview video plus depth based 3D video coding," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 562–571, Jun. 2011.

[17] J. Lee, H. Wey, and D. Park, "A fast and efficient multiview 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.

[18] 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.

[19] J. Zhang, M. M. Hannuksela, and H. Li, "Joint multiview video plus depth coding," in *Proc. 17th IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 2865–2868.

[20] 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.

[21] Y. H. Lin and J.-L. Wu, "A depth information based fast mode decision algorithm for color plus depth-map 3D videos," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 542–550, Jun. 2011.

[22] B. Kamolrat, W. Fernando, and M. Mrak, "Adaptive motion estimation mode selection for depth video coding," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Mar. 2010, pp. 702–705.

[23] D. D. Silva, W. Fernando, and H. K. Arachhi, "A new mode selection technique for coding depth maps of 3D video," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Mar. 2010, pp. 686–689.

[24] 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.

[25] 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 Int. Conf. Image Process.*, Nov. 2009, pp. 721–724.

[26] 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.

[27] 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.

[28] M. Tanimoto, T. Fujii, and K. Suzuki, "Improvement of depth map estimation and view synthesis," *MPEG (ISO/IEC JTC1/SC29/WG11)*, Antalya, Turkey, Tech. Rep. M15090, Jan. 2008.

[29] P. Lai, A. Ortega, C. C. Dorea, P. Yin, and C. Gomila, "Improving view rendering quality and coding efficiency by suppressing compression artifacts in depth-image coding," *Proc. SPIE*, vol. 7257, p. 725700, Jan. 2009.

[30] 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.

[31] N. Kamaci, Y. Altinbasak, and R. M. Mersereau, "Frame bit allocation for the H.264/AVC video coder via cauchy density-based rate and distortion model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 8, pp. 994–1006, Aug. 2005.

[32] H. Wang and S. Kwong, "A rate-distortion optimization algorithm for rate control in H.264," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2007, pp. 1149–1152.

[33] K. Takagi, Y. Takishima, and Y. Nakajima, "A study on rate distortion optimization scheme for JVT coder," *Proc. SPIE*, vol. 5150, pp. 914–923, Jul. 2003.

[34] *Joint Multiview Video Coding (JMVC) 8.0 Software Package*, CVS Server for JMVC Software, Boston, MA, USA, Mar. 2010.

[35] M. Tanimoto, T. Fujii, M. P. Tehrani, and M. Wildeboer, "Depth estimation reference software (DERS) 5.0," *MPEG (ISO/IEC JTC1/SC29/WG11)*, Xian, China, Tech. Rep. M16923, Oct. 2009.

[36] M. Tanimoto, T. Fujii, and K. Suzuki, "View synthesis algorithm in view synthesis reference software 3.0 (VSRS3.0)," *MPEG (ISO/IEC JTC1/SC29/WG11)*, Lausanne, Switzerland, Tech. Rep. M16090, Apr. 2009.

[37] G. Bjontegaard, "Calculation of average PSNR differences between RD-Curves," *ITU-T VCEG (ITU-T SG16 Q.6)*, Austin TX, US, Document VCEG-M33, April 2001.

[38] *Methodology for the Subjective Assessment of the Quality of Television Pictures*, ITU-R Standard BT.500-11, Jun. 2002.

[39] MSU Graphics & Media Lab. *Video Group, MSU Filters and Codecs*, Moscow, Russia [Online]. Available: http://www.compression.ru/video/, 2006.

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 2013, he was a Visiting Scholar with the Department of Computer Sciences, City University of Hong Kong, Kowloon, Hong Kong. In 2010, he joined the Shenzhen Institutes of Advanced Technology, CAS, as an Assistant Professor. Since 2012, he has served as an Associate Professor. His current research interests include multiview video coding, video object segmentation, and content based video processing.

Sam Kwong (M'93–SM'04) received the B.S. and M.S. degrees in electrical engineering from the State University of New York at Buffalo, Buffalo, NY, USA, in 1983, 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. He joined Bell Northern Research Canada as a Scientific Staff member. In 1990, he became a Lecturer with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong, where he is currently a Professor with the Department of Computer Science. His current research interests include video and image coding and evolutionary algorithms.

Long Xu received the M.S. degree in applied mathematics from Xi'dian University, Xi'an, China, in 2002, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2009. He was a Post-Doctoral Researcher with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, and the Department of Electronic Engineering, Chinese University of Hong Kong, Shatin, Hong Kong, from August 2009 to December 2012. He is currently with the School of Automation and Electrical Engineering, University of Science and Technology, Beijing. His current research interests include image/video coding, wavelet-based image/video coding, and computer vision.

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, Kowloon, 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 current research interests include image and video compression, rate control, scalable video coding, and 3-D video coding.

Gangyi Jiang (M'10) received the M.S. degree in electronics engineering from Hangzhou University, Hangzhou, China, in 1992, and the Ph.D. degree in electronics engineering from Ajou University, Suwon, South Korea, in 2000. He is currently a Professor with the Faculty of Information Science and Engineering, Ningbo University, Ningbo, China. His current research interests include video compression and multi-view video coding. He has published over 100 technical articles in refereed journals and proceedings.

C.-C. Jay Kuo (F'99) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1980, and the M.S. and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1985 and 1987, respectively. He is 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 current research interests include digital image/video analysis and modeling, multimedia data compression, communication and networking, and biological signal/image processing. He is the co-author of over 200 journal papers, 850 conference papers, and ten books. He is a fellow of the American Association for the Advancement of Science and the International Society for Optical Engineers.