Highly Efficient Multiview Depth Coding Based on Histogram Projection and Allowable Depth Distortion

Yun Zhang, *Senior Member, IEEE*, Linwei Zhu, Raouf Hamzaoui, *Senior Member, IEEE*,
Sam Kwong, *Fellow IEEE*, Yo-Sung Ho, *Fellow IEEE*

***Abstract*—Mismatches between the precisions of representing the disparity, depth value and rendering position in 3D video systems cause redundancies in depth map representations. In this paper, we propose a highly efficient multiview depth coding scheme based on Depth Histogram Projection (DHP) and Allowable Depth Distortion (ADD) in view synthesis. Firstly, DHP exploits the sparse representation of depth maps generated from stereo matching to reduce the residual error from INTER and INTRA predictions in depth coding. We provide a mathematical foundation for DHP-based lossless depth coding by theoretically analyzing its rate-distortion cost. Then, due to the mismatch between depth value and rendering position, there is a many-to-one mapping relationship between them in view synthesis, which induces the ADD model. Based on this ADD model and DHP, depth coding with lossless view synthesis quality is proposed to further improve the compression performance of depth coding while maintaining the same synthesized video quality. Experimental results reveal that the proposed DHP based depth coding can achieve an average bit rate saving of 20.66% to 19.52% for lossless coding on Multiview High Efficiency Video Coding (MV-HEVC) with different groups of pictures. In addition, our depth coding based on DHP and ADD achieves an average depth bit rate reduction of 46.69%, 34.12% and 28.68% for lossless view synthesis quality when the rendering precision varies from integer, half to quarter pixels, respectively. We obtain similar gains for lossless depth coding on the 3D-HEVC, HEVC Intra coding and JPEG2000 platforms.**

***Index Terms*—lossless coding, depth coding, HEVC, depth histogram projection, allowable depth distortion, view synthesis.**

Manuscript received on xxxx. This work was supported in part by the Shenzhen Science and Technology Program under Grant JCYJ20180507183823045, in part by Guangdong International Science and Technology Cooperative Research Project under Grant 2018A050506063, in part by the National Natural Science Foundation of China under Grant 61901459, in part by China Postdoctoral Science Foundation under Grant 2019M653127, in part by Membership of Youth Innovation Promotion Association, Chinese Academy of Sciences under Grant 2018392, in part by CAS President's International Fellowship Initiative (PIFI) under Grant 2020VTA008.

Y. Zhang and L. Zhu are with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China (email: {yun.zhang, lw.zhu}@siat.ac.cn).

R. Hamzaoui is with School of Engineering and Sustainable Development, Faculty of Computing, Engineering and Media, De Montfort University, Leicester, United Kingdom (email: rhamzaoui@dmu.ac.uk).

S. Kwong is with Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, China (email: cssamk@cityu.edu.hk).

Y. Ho is with the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 500-712, South Korea (e-mail: hoyo@gist.ac.kr).

I. Introduction

THREE Dimensional (3D) video [1] and 3D immersive telepresence are able to provide immersive 3D visual perception and seamlessly 3D arbitrary virtual view rendering. The 3D video has a large potential market and plays an important role in many areas of human life, such as immersive 3D communication, Virtual Reality (VR) and Augmented Reality (AR), 3D-TV, manufacturing, entertainment, and robotics. Alongside multiview color/texture videos, multiview depth maps are one of the most important components in 3D video representation. The multiview depth maps provide the geometrical information of a 3D scene, which enable 3D interactive functionalities and arbitrary virtual view rendering via Depth Image Based Rendering (DIBR) [2]. Depth information is also one of the key components in dynamic Video based Point Cloud Compression (V-PCC) that enables advanced immersive VR and AR applications, such as six Degree-of-Freedom (6DoF) VR. However, because the volume of the 3D visual data is hundreds or even thousands times of that of the conventional 2D videos, high efficiency compression is desired for 3D video transmission and storage.

In July 2012, Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) was established by experts from Moving Picture Expert Group (MPEG) and Video Coding Expert Group (VCEG) to develop and standardize 3D Video Coding (3DVC) algorithms. Two extensions of High Efficiency Video Coding (HEVC) [3] were developed, i.e., 3D-HEVC and Multiview HEVC (MV-HEVC) [1], to compress multiview video plus depth. The depth information was formatted as the luminance component in color video and then encoded. However, compared with traditional color/texture video, depth map represents the geometrical information of a video object and has unique characteristics, such as sharper contours and smoother contents [4]. In addition, the depth map is used for view synthesis instead of being watched. Thus, the conventional video encoder developed for natural color video may not be the optimal solution for depth coding. Highly efficient depth coding algorithms and tools are desired.

A. Related Works

A number of advanced coding tools have been proposed [5]-[9] for coding depth maps in 3DVC. For example, Depth

Modeling Modes (DMMs) [5][6] that exploit the sharp depth edge characteristics were proposed to preserve the depth edges. A depth lookup table [7][8] was proposed to exploit the property that in Intra prediction only a small number of depth levels may be presented due to strong quantization. Segment-wise depth coding [9] and Depth Wedge and Contour (DWC) for Intra modes [4][10] were proposed to exploit the property that depth maps contain many smooth areas with similar sample values. The texture characteristics of depth maps, which are quite different from those of color images, can be exploited to improve the coding efficiency [10][11].

To further improve the coding efficiency, a number of depth coding algorithms [12]-[20] were proposed by further exploiting depth map properties. Peng *et al.* [12] proposed spatial and temporal enhancement filters for the depth discontinuous regions, depth edge regions, and motion regions, which reduce the prediction residual error and coding complexity in mode decision. Since the depth edge is of greater importance, Shahriyar *et al.* [13] proposed a mono-view depth encoder, which preserves edges implicitly by limiting quantization to the spatial-domain. At the same time, the frame-level clustering tendency was exploited with a binary tree based decomposition to achieve higher efficiency in arithmetic coding. This scheme achieves lower bitrate at lossless to near-lossless quality range for mono-view coding. Georgiev *et al.* [14] proposed a down-sampling based depth coding scheme, where the misalignments of depth edges are preserved and refined with the help of super-pixel segmentation of the color video. To improve the depth coding efficiency, asymmetric depth coding algorithms [15][16] were proposed by encoding some of the depth views with reduced resolution and then reconstructing the depth map to the original resolution at the client side with up-sampling. To improve the quality of distorted multiview depth maps in asymmetric coding, residual learning framework [15], Convolutional Neural Network (CNN) based up-sampling [16] and cross-view multi-lateral filter [17] were proposed to enhance the up-sampling quality of depth maps, where the correlations among viewpoints and between color and depth channels were exploited. Stankiewicz *et al.* [18] proposed 3D depth coding algorithms using Nonlinear Depth Representation (NDR), where a power law transformation and a piece-wise function were used to nonlinearly remap the depth information according to its relative importance. For example, closer objects were given a higher dynamic range. Furthermore, fast coding algorithms were proposed to reduce the depth coding complexity by exploiting the depth coding statistics [19], smooth property [11], grayscale similarity and inter-view correlation [20]. The coding objective of these schemes aims at improving the depth map quality. However, since the depth maps are mainly used for virtual view rendering via DIBR rather than being viewed directly, the quality of the rendered view should be considered.

A number of works [21]-[33] have been devoted to improving the view synthesis image quality while encoding the multiview depth maps. Since depth distortion has different impacts on the view synthesis distortion according to the texture of corresponding color videos, Zhang *et al.* [21] proposed regional View Synthesis Distortion (VSD) prediction models for different regions in a depth map. Then, regional bit allocation [21] and sparse representation based depth map super-resolution [22] were proposed to improve the synthesized image quality with the regional VSD model, which exploited the relative importance of depth regions. Lei *et al.* [23] proposed rate control models for depth map coding based on the different depth distortion's regional impacts on virtual view rendering. Gao *et al*. [24] proposed an efficient rate distortion optimization scheme to minimize the view synthesis distortion, in which the texture and depth modes were jointly determined. Jin *et al.* [25] presented a depth bin based graphical model, where the process of view synthesis was formulated at depth bin level, such that fast VSD estimation could be performed. Different VSD prediction models [26]-[30] were proposed to improve the prediction accuracy of the VSD, which can be used as the objective in depth coding optimization. In addition, View Synthesis Optimization (VSO) [31] was proposed to search for the best matching mode and block by calculating the synthesized view distortion change subject to a given bit rate constraint. This approach is more accurate than the VSD prediction models but has higher computational complexity. These schemes aim to improve the quality of synthesized image in terms of Peak Signal-to-Noise Ratio (PSNR), which does not truly reflect human perceived quality.

To handle this problem, 3D Synthesized View Image Quality Metric (3DSwIM) [32] was proposed to measure the perceived quality of the synthesized image and a depth coding algorithm was proposed to improve the 3DSwIM value via preserving the depth edges. Furthermore, since the synthesized videos have annoying flicker due to temporal inconsistency of depth maps, video quality assessment models were proposed in [34] and [35]. Then, depth coding optimization [33] was applied to reduce the flickering artifacts in the synthesized video and improve the perceptual video quality. These works [21]-[33] are lossy depth coding algorithms aiming at minimizing the distortion in rendered views at a given bit rate.

To maintain high depth map quality, lossless and near-lossless depth coding are desired in some specific applications, such as point cloud processing, 3D reproduction, 3D modeling and editing, measuring, medicine and remote control. JPEG-LS [36], JPEG2000 [37], JPEG-XR [38], and HEVC [1], which support lossless encoding for natural color image/video, can be used to encode the depth information while regarding the depth map as the luminance component of color video. However, they might not be optimal since the depth characteristics were not considered. Since the depth maps are smooth and have less texture than the natural color images, Kim *et al.* [39] proposed a bit-plane-based lossless depth-map coding method, where the depth map was decomposed as a number of simple bit planes and then encoded independently from the most significant bit to the least significant bit. The method achieved significant coding gain as compared to H.264/AVC with Context-based Adaptive Binary Arithmetic Coding (CABAC) for Intra and Inter coding. Heo *et al.* [40] improved CABAC coding for lossless depth map coding based on the statistics of the depth residual, and a bit rate reduction of about 4% was achieved. Shahriyar *et al.* [41] proposed a binary tree based lossless depth coding scheme that arranged the residual frame into an integer or binary residual bitmap. High spatial correlation in depth residual frames was exploited by creating large homogeneous blocks with adaptive size, which were then coded as a unit using context based arithmetic coding.

These are lossless coding optimizations for depth map, whose coding performances were measured with the conventional depth map quality and bit rate. However, view synthesis distortion must be considered in the depth coding which targets rendering.

Due to the mismatch between the number of depth levels and disparity levels, not every depth distortion causes the geometrical distortion in view synthesis. Thus, Zhao *et al.* [42] proposed a Depth No-Synthesis-Error (D-NOSE) model to examine the depth distortions in view synthesis without introducing any geometrical changes. Zhang *et al.* [43] proposed an Allowable Depth Distortion (ADD) model for depth map coding, which modelled the relationship between depth distortion and rendering position error as a many-to-one mapping function. Then, the ADD model [43] was applied to the Rate-Distortion Optimization (RDO) in mode decision, bit allocation and Intra coding [44] in lossy depth coding for further bit reduction. Gao *et al.* [45] further exploited ADD for occlusion-inducing depth pixels in view synthesis, which was applied to depth coding for higher compression ratio. These D-NOSE and ADD models were proposed for lossy coding and might result in lossy or lossless view synthesis quality.

B. Contributions and Organizations of this Work

3D video systems do not display the depth map directly but use it to synthesize virtual views. Therefore, the ultimate aim of depth coding is to minimize the bit rate of the depth information without affecting the quality of the synthesized views. This can be achieved with lossless coding of the depth maps but also with lossy depth coding provided the quality of the synthesized video is not affected.

In this paper, we find that the histogram of the depth map is very sparse and redundant in representation. Thus, we propose a highly efficient depth coding scheme based on Depth Histogram Projection (DHP) and ADD model. Our main contributions are as follows:

1) We propose a framework of DHP based lossless depth coding that significantly improves the coding efficiency.
2) We theoretically analyze the cost and gain of DHP- based depth coding, providing a mathematical foundation for DHP-based lossless depth coding.
3) We propose a depth coding method by combining DHP and ADD, which further improves the depth coding efficiency without affecting the quality of the synthesized views.

The remainder of this paper is organized as follows. Section II analyzes the redundancies in depth and presents the proposed depth coding framework. Section III proposes DHP for lossless depth coding and Section IV presents the DHP and ADD model-based depth coding for lossless view synthesis quality. Section V analyzes DHP's key factors and presents the syntax of encoding overhead coefficients from DHP. Experimental results and analysis are presented in Section VI. Finally, Section VII draws the conclusions.

II. Depth Redundancy Analysis and the Proposed Multiview Depth Coding Framework

A. Analysis on Depth Map Redundancies

A color image represents a 2D scene of the 3D world, while a depth map represents the distance between the visual scene and the camera. The depth map is used as geometrical information for virtual view synthesis in 3D video systems. There are three main methods to generate depth maps. The first one is based on range imaging camera, which uses the Time-of-Flight (TOF) principle to measure the distance between the camera and each point of an object. While such depth cameras are accurate, they are very expensive and have limited capturing resolution (about 320×240). The second method is generating the depth map from 3D models via 3D animation or computer graphics, where depth maps with high quality and large resolution can be generated. However, they are animated videos and it is very challenging to generate depth maps for natural and realistic scenes. The third, and also the most commonly used method, is to use stereo-matching from two or more views. Although the stereo-matching method is not as accurate as the other two methods, it is less expensive, more practical and can generate depth maps with high resolution for natural scenes.

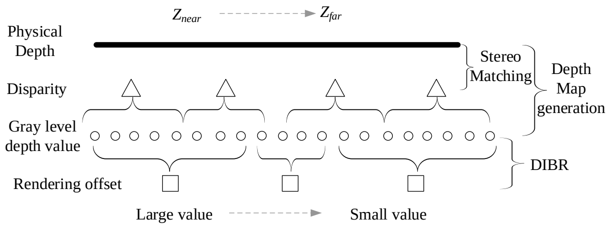


Fig.1. Mapping relationships among physical depth, disparity, depth value and rendering offset.

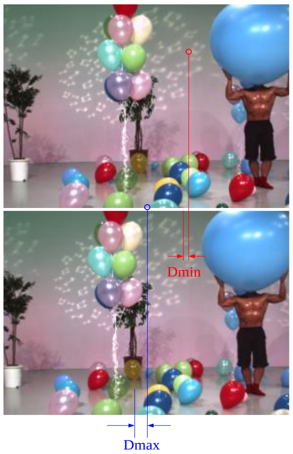


Fig.2. Example of D_{min} and D_{max} in the Balloons sequence.

In generating the depth map via stereo-matching, continuous physical depth is converted to pixel-wise disparity and then gray level depth value. However, there are fidelity mismatches among physical depth, disparity and depth value, as shown in Fig.1. Based on the stereo-matching algorithm for the parallel camera system, the physical depth of pixels at location (*x*,*y*), *z*(*x*,*y*), is calculated as

$$z\left(x,y\right)=\frac{fb}{d_{m,x}\left(x,y\right)}\ , \tag{1}$$

where $d_{m,x}(x,y)$ is the physical parallax between the CCD image planes of the two cameras, *f* is the focal length of the cameras, and *b* is the baseline distance between the two cameras. While capturing the multiview videos, there is a mapping between physical parallax and pixel-wise disparity, which is

$$d_{m,x}\left(x,y\right)=r\cdot d_{p,x}\left(x,y\right)\ , \tag{2}$$

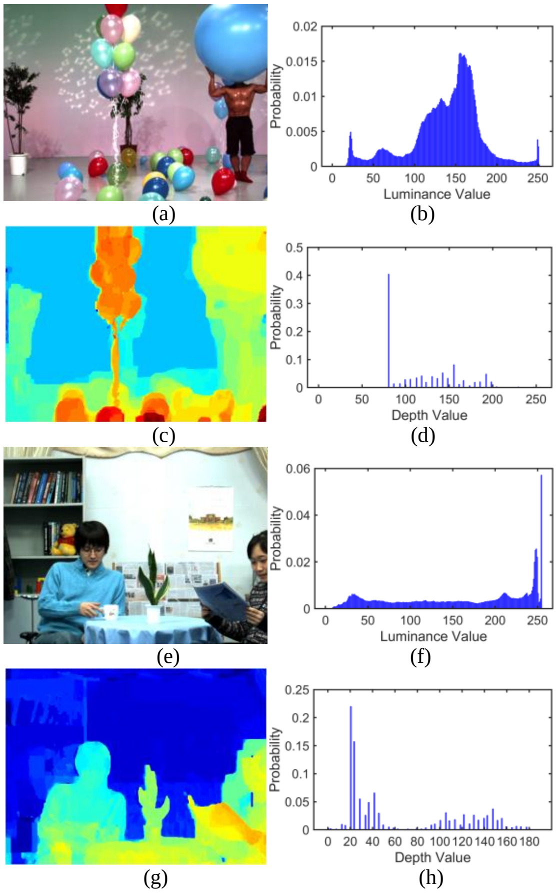


Fig.3. Histogram maps of the color images and depth maps. (a) (e) Color image, (b) (f) Histogram map of color image, (c) (g) Depth map, (d)(h) Histogram map of depth map, (a)-(d) Balloons, (e)-(h) Newspapers.

where $d_{p,x}(x,y)$ represents the pixel-wise disparity of 3D points imaged on the two cameras, and r represents the actual physical distance per pixel, which can be non-linear in case of non-linear depth mapping and quantization.

For the depth map in MPEG-3DV, a non-linear quantization is adopted to convert the physical depth z to an n-bit depth value v in [0, 2^n-1] as [2]

$$v = Q(z) = \left\lfloor 2^n \frac{z_{far}}{z} \frac{z_{near}}{z_{far} - z_{near}} + 0.5 \right\rfloor, \qquad (3)$$

where "$\lfloor \ \rfloor$" is the floor operation, and z_{near} and z_{far} are the distances from the camera to the nearest and furthest depth planes of a scene, respectively. z_{near} and z_{far} are

$$\begin{cases} z_{near} = \dfrac{b}{D_{max}} \cdot \dfrac{f}{r} \\ z_{far} = \dfrac{b}{D_{min}} \cdot \dfrac{f}{r} \end{cases}, \qquad (4)$$

where D_{max} and D_{min} are the maximum and minimum disparities of the 3D scene, *i.e.*, max($d_{p,x}(x,y)$) and min($d_{p,x}(x,y)$). If we substitute the right-hand side of Eq. (1) for z in Eq. (3), we get the depth value at (x,y), $v(x,y)$, from the disparity $d_{p,x}(x,y)$ as

$$v(x,y) = Q(z) = \left\lfloor 2^n \frac{z_{far}}{fb} r \cdot d_{p,x}(x,y) \frac{z_{near}}{z_{far} - z_{near}} + 0.5 \right\rfloor. \qquad (5)$$

The depth value v usually ranges from 0 to 2^n-1, that is, from 0 to 255 (*i.e.*, 256 levels) when n is 8. However, due to the rounding operation, the pixel-wise disparity $d_{p,x}(x,y)$ is an integer in the range [D_{min}, D_{max}], which is usually very limited as compared with the continuous physical depth and gray level depth value, as shown in Fig.1. In addition, Fig.2 shows an example of D_{min} and D_{max} for the *Balloons* sequence. We observe that D_{min} is about 20 in the background while D_{max} is about 70 in the foreground. There are about 50 levels from D_{min} to D_{max} at integer precision, which correspond to 256 levels for depth value v. So, there is a big mismatch between the number of levels in disparity $d_{p,x}$ and the number of scales in the depth value v, leading to depth representation redundancies to be exploited.

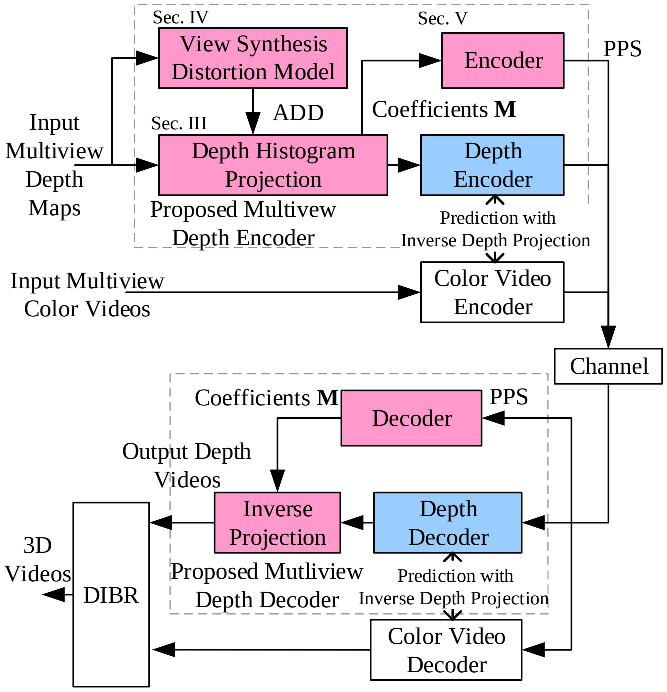

Sec. IV
Sec. V
View Synthesis Distortion Model
Encoder
PPS
Input Multiview Depth Maps
Sec. III
ADD
Coefficients M
Depth Histogram Projection
Depth Encoder
Proposed Multivew Depth Encoder
Prediction with Inverse Depth Projection
Input Multiview Color Videos
Color Video Encoder
Channel
Coefficients M
Decoder
PPS
Output Depth Videos
3D Videos
DIBR
Inverse Projection
Depth Decoder
Proposed Mutliview Depth Decoder
Prediction with Inverse Depth Projection
Color Video Decoder

Fig.4. Framework of the proposed DHP-based 3D video system.

Compared with the color image, the depth map is usually smoother and has less texture [9][33]. Fig.3 shows the histogram of color and depth for the *Balloons* and *Newspapers* sequences. For better visualization, the pseudo-color in the jet color map has been used to represent the grayscale version of the depth map. We observe that the histogram of color is dense and with continuous-tone from 0 to 255. However, the depth map is much smoother and its histogram is much sparser since many bins are empty. Also, in the depth histogram, only a very small number of bins show non-zero probabilities, which are much larger (about ten times) than those in the color image histogram.

In 3D video systems, the generated or captured multiview depth maps are treated as the luminance component and then encoded with the encoder. Then, these depth bit-streams are transmitted to the client and decoded for view synthesis. The depth map with sparse histogram will likely cause large residual errors from the Intra and Inter predictions in video coding. This leads to large coefficients after transform and quantization, which requires more encoding bits. To address this problem, we propose a depth map histogram projection to improve the depth coding efficiency.

B. Framework of the Proposed Depth Coding System

Due to the sparse representation and view synthesis redundancies in depth maps, we propose DHP to exploit the representation redundancies and use the ADD model to jointly exploit view synthesis redundancies, which are able to effectively improve lossless depth coding efficiency. Fig.4 shows the framework of the proposed DHP-based lossless 3D video coding system. The encoder side includes the DHP module, view synthesis distortion model, lossless depth encoder and coefficient encoder. The DHP module analyzes the depth map sequence, re-projects the histogram of the depth map, and outputs the array **M** of coefficients that characterizes the histogram projection of depth maps. The coefficients in **M** are encoded by a lossless encoder and the depth maps are encoded by the lossless video encoder. Finally, the bit-stream of coded depth and coefficients **M** are multiplexed and transmitted to the client with the bit-stream of associated coded color videos.

The 3D video encoder consists of a multiview color video encoder and a multiview depth encoder. In this work, we only optimize the multiview depth encoder by exploiting representation redundancies with DHP and view synthesis redundancies with the ADD model. The multiview color video encoder is not modified. Key modules of the proposed depth encoder will be presented in detail in Sections III to V.

At the client side, the proposed depth decoder includes a depth decoder, a coefficient decoder and an inverse depth projection. The depth maps and coefficients are decoded and reconstructed from the transmitted bit-streams, which are then used to reconstruct the final depth map by the inverse projection. Finally, the decoded multiview color videos from conventional color video decoder and the reconstructed depth maps are input to the DIBR module [2] for synthesizing the intermediate virtual view images required by the users.

III. Proposed DHP for Lossless Depth Coding

A. Proposed DHP for Depth Coding

Let **X** be the input depth maps, **H**(**X**) be the histogram of **X**, **Y** be the output depth maps obtained after applying the histogram projection on **H**(**X**), and **H**(**Y**) be the histogram of **Y**. So, the forward and inverse DHP processes are implemented as

$$\begin{cases}\mathbf{H}(\mathbf{Y})=\mathbf{H}(\mathbf{X})\otimes\mathbf{M}=\left\{y_i \mid y_i=LUT_{FWD}\left(x_i\right), i\in[0,2^n-1]\right\}\\ \mathbf{H}(\mathbf{X})=\mathbf{H}(\mathbf{Y})\oplus\mathbf{M}=\left\{x_i \mid x_i=LUT_{INV}\left(y_i\right), i\in[0,2^n-1]\right\}\end{cases}, \qquad (6)$$

where **M** is an array of coefficients characterizing the histogram projection, ⊗ and ⊕ are the forward and inverse histogram projections, respectively. In fact, Eq.(6) is a projection function for the DHP, which can be implemented as look-up table LUT_{FWD}() and LUT_{INV}(). The array **M** is generated while transforming **H**(**X**) to **H**(**Y**), which is denoted as **M**:**H**(**X**)→ **H**(**Y**). The forward and inverse DHP for the depth histograms are reversible, which are denoted as $\mathbf{H}(\mathbf{X})\overset{\otimes\mathbf{M}}{\Rightarrow}\mathbf{H}(\mathbf{Y})\overset{\oplus\mathbf{M}}{\Rightarrow}\mathbf{H}(\mathbf{X})$. The forward and inverse DHP for depth maps are lossless, which are denoted as $\mathbf{X}\overset{\otimes\mathbf{M}}{\Rightarrow}\mathbf{Y}\overset{\oplus\mathbf{M}}{\Rightarrow}\mathbf{X}$. However, compression distortion may be introduced in lossy depth coding. Let $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ be the reconstructed images from compressed **X** and **Y,** which have quantization errors. The original depth coding is $\mathbf{X}\overset{Enc}{\Rightarrow}\tilde{\mathbf{X}}$ and the coding distortion caused by the quantization is $D_{ENC}=\left\|\tilde{\mathbf{X}}-\mathbf{X}\right\|$. In the proposed framework, the depth maps in the depth coding process are changed as $\mathbf{X}\overset{\otimes\mathbf{M}}{\Rightarrow}\mathbf{Y}\overset{Enc}{\Rightarrow}\tilde{\mathbf{Y}}\overset{\oplus\mathbf{M}}{\Rightarrow}\mathbf{Z}$ and the distortion is D_N=||**Z**-**X**||. For lossless coding, D_N is zero since $\mathbf{Y}=\tilde{\mathbf{Y}}$ and **Z** = **X**.

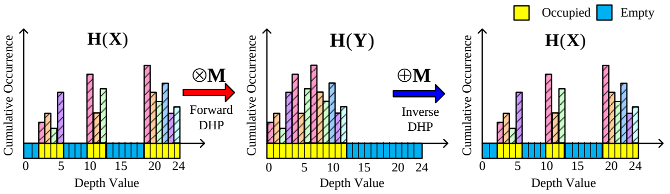


Fig.5. Example of the depth histogram projection.

The optimal solution **M** for lossy or lossless depth coding can be found via solving the optimization problem

$$\mathbf{M}^*=\arg\min_{\mathbf{M}}\left[D+\lambda\left(R_b+R_O\right)\right], \qquad (7)$$

where *D* is the depth distortion or view synthesis distortion caused by the mismatch between **Z** and **X**. Usually, D=D_N for conventional depth coding and D=f(D_N) for synthesized quality oriented depth coding, where *f*() is a view synthesis quality mapping function [21][26][27]. R_b is the bit rate from coding the transformed residual, motion vectors and block types, *etc.* R_O is the overhead bit rate used to encode the array **M.** For lossless coding, D_N is zero and Eq.(7) aims to find an optimal **M** that minimizes the overall bit rate R_b+R_O.

Since solving Eq.(7) is computationally expensive and requires involving the re-encoding process many times, we focus on one important special case in this paper. Fig. 5 shows an example of DHP, where the left and right histograms are **H**(**X**) and the middle one is **H**(**Y**). The horizontal and vertical axes indicate the depth value and the ratios of the depth values, respectively. The yellow area represents non-empty bins, while the blue area represents empty bins. All empty bins are removed and non-empty bins are shifted to the left. Their original positions are recorded in the array of coefficients $\mathbf{M}_S$ for forward and inverse DHP, which is a special case of **M**. This DHP is reversible and lossless. The i^{th} row of $\mathbf{M}_S$, denoted by $\mathbf{M}_S$[*i*], is a 1D array of coefficients for projecting the histogram of all depth frames in the i^{th} Group-of-Picture (GOP), which is presented as

$$\mathbf{M}_S[i]=\left[s_{init}\left(i\right)\quad a_{i,1}\quad b_{i,1}\quad\cdots\quad a_{i,m_i}\quad b_{i,m_i}\right], \qquad (8)$$

where $s_{init}(i)$ is an integer indicating the first non-empty bin in the depth histogram of the i^{th} GOP, $a_{i,j}$ and $b_{i,j}$ are the number of continuously non-empty bins or continuously empty bins in m_i^{th} clustered bins and i^{th} GOP, $j\in[1,m_i]$, m_i is the number of clusters of the continuously non-empty bins in **H**(**X**). In fact, m_i depends on image content and may be different from m_j when $i\neq j$. $\mathbf{M}_S$ is a 2D array that consists of a number of 1D array $\mathbf{M}_S[i]$, *i.e.*, $\mathbf{M}_S$={$\mathbf{M}_S[i]$|$i\in[1,l]$}, where *l* is the number of GOPs in a depth sequence. Fig. 5 shows an example of the histogram projection for the i^{th} GOP, where the depth value ranges from 0 to 24. The i^{th} row of array $\mathbf{M}_S$, *i.e.*, $\mathbf{M}_S[i]$, can be written as [2, 4, 4, 3, 6, 6, 0], where $s_{init}(i)$=2, $a_{i,1}$=4, $b_{i,1}$=4, $a_{i,2}$=3, $b_{i,2}$=6, $a_{i,3}$=6, $b_{i,3}$=0, m_i=3.

Let k denote the ratio of the range of non-empty bins in **H**(**X**) to its average number of non-empty bins, which is

$$k=\frac{1}{l}\sum_{i=1}^{l}\frac{\sum_{j=1}^{m_i}\left(a_{i,j}+b_{i,j}\right)-b_{i,m_i}}{\sum_{j=1}^{m_i}a_{i,j}}=\frac{1}{l}\sum_{i=1}^{l}\frac{2^n-s_{init}\left(i\right)-b_{i,m_i}}{\sum_{j=1}^{m_i}a_{i,j}}, \quad (9)$$

where n is the bit-depth of the depth map, *e.g.*, 8-bit per channel, l is the number of GOPs. The histogram becomes denser after applying DHP, and k indicates the representation redundancy in the histogram, usually $k\geq 1$. $k=1$ means the bins in histogram are continuous, *i.e.*, $b_{i,j}$ is 0 when $i\in[1,l]$, $j\in[1,m_i]$, and the density of the histogram is the same as the original one. Take the histogram in Fig. 5 as an example. The number of bins is 25, and k is calculated as $\frac{(4+4)+(3+6)+(6+0)-0}{4+3+6}=\frac{23}{13}$ or $\frac{25-2-0}{4+3+6}=\frac{23}{13}$.

B. Cost and Gain Analysis for DHP based Depth Coding

In this subsection, we theoretically analyze the cost and gain from DHP when it is applied to depth coding.

1) Bit Rate Gain Analysis

Let U be a random variable representing the quantizer input. Suppose U is mapped to a discrete-valued random variable V. The minimum entropy of V, denoted by H_{min}, can be expressed as [46]

$$H_{min}=H_0\text{-log}Q, \quad (10)$$

where Q is the quantization step and H_0 is the entropy of U. Let $f_U(u)$ be the probability density function of the random variable U. Generally, the DC and AC coefficients from Inter and Intra predictions are approximately uncorrelated and Laplace distributed with variance σ^2 [47], *i.e.*, $f_U\left(u\right)=\frac{1}{\beta}e^{-|u|/\beta}$, $\beta=\sigma/\sqrt{2}$. Thus, H_0 can be calculated as [47]

$$H_0=-\int_{-\infty}^{+\infty}f_U\left(u\right)\log f_U\left(u\right)du=\log\sqrt{2}e\sigma \ . \quad (11)$$

Therefore, the minimum entropy H_{min} after quantization, *i.e.*, the bit rate R, can be expressed as

$$H_{min}=R=\log\frac{\sqrt{2}e\sigma}{Q} \ . \quad (12)$$

In video coding, the variance of the residual error can be calculated as

$$\sigma^2=\frac{1}{N}\sum\left(r_i-\frac{1}{N}\sum r_i\right)^2=E\left(r^2\right)-E\left(r\right)^2, \quad (13)$$

where $r_i=X_i-\hat{X}_i$, N is the number of pixels, $E()$ is the expectation operator, which can be regarded as an average according to the Law of Large Number (LLN). Here, X_i is an original depth value and $\hat{X}_i$ is the predicted depth value with Intra or Inter prediction at the same position i in a depth map. So, Eq.(12) gives the bit rate of encoding depth maps with the conventional encoder.

For the depth map processed by forward DHP with array **M**s, as shown in Fig.5 and Eq. (8), the depth value X_i of the original depth map **X** is changed to Y_i in **Y**, which can be mathematically expressed as

$$Y_i=\frac{1}{k}\left(X_i-s_{init}\right)+\varepsilon_i \ , \quad (14)$$

where X_i and Y_i are seen as random variables to facilitate the statistical analysis, k is the scaling factor, ε_i is a random error with zero mean and independent of X_i, and s_{init} is a starting value. In predicting Y_i, the depth values of the spatial-temporal neighboring pixels are linearly combined to generate its prediction $\hat{Y}_i$. That is $\hat{Y}_i=\frac{1}{m_i}\sum_{j=1}^{m_i}\omega_{i,j}Y_{i,j}$, where $Y_{i,j}$ is a spatial or temporal neighboring pixel of Y_i used in the prediction, , m_i is the number of pixels used in the prediction for pixel i, and $\omega_{i,j}$ is a weighting factor that satisfies $\frac{1}{m_i}\sum_{j=1}^{m_i}\omega_{i,j}=1$. So, one can easily show that the predicted depth value $\hat{Y}_i$ also satisfies

$$\hat{Y}_i=\frac{1}{k}\left(\hat{X}_i-s_{init}\right)+\varsigma_i \, , \quad (15)$$

where ς_i is an error factor satisfying $\varsigma_i=\frac{1}{m_i}\sum_j\omega_{i,j}\varepsilon_{i,j}$, and $\varepsilon_{i,j}$ is the error factor of $Y_{i,j}$. It means ς_i and ε_i have zero mean and are dependent. Thus, $E(\varepsilon)=0$, $E(\zeta)=0$, and $E^2(\varepsilon\text{-}\zeta)=0$ based on the LLN.

The variance σ_N^2 after DHP is defined as

$$\sigma_N{}^2=\frac{1}{N}\sum\left(q_i-\bar{q}\right)^2=\frac{1}{N}\sum q_i{}^2-\bar{q}^2 \ , \quad (16)$$

where $q_i=Y_i-\hat{Y}_i=\frac{1}{k}\left(X_i-\hat{X}_i\right)+\varepsilon_i-\varsigma_i$, $\bar{q}=\frac{1}{N}\sum q_i$. Thus

$$\begin{aligned}\frac{1}{N}\sum q_i{}^2&=\frac{1}{N}\sum\left(\frac{1}{k}\left(X_i-\hat{X}_i\right)+\varepsilon_i-\varsigma_i\right)^2\\&=\frac{1}{k^2N}\sum\left(X_i-\hat{X}_i\right)^2+\frac{2}{kN}\sum\left(X_i-\hat{X}_i\right)\left(\varepsilon_i-\varsigma_i\right)+\frac{1}{N}\sum\left(\varepsilon_i-\varsigma_i\right)^2\\&=\frac{1}{k^2}E\left(r^2\right)+2\frac{1}{k}E\left(r\left(\varepsilon-\varsigma\right)\right)+E^2\left(\varepsilon-\varsigma\right)\overset{LLN}{=}\frac{1}{k^2}E\left(r^2\right)\end{aligned} \ , \quad (17)$$

and

$$\begin{aligned}\bar{q}&=\frac{1}{N}\sum q_i=\frac{1}{kN}\sum\left(X_i-\hat{X}_i\right)+E\left(\varepsilon\right)-E\left(\varsigma\right)\\&=\frac{1}{k}\bar{r}+E\left(\varepsilon\right)-E\left(\varsigma\right)\overset{LNN}{=}\frac{1}{k}\bar{r}\end{aligned} \ . \quad (18)$$

Substituting Eq.(13), Eq.(17) and Eq.(18) into Eq.(16),

$$\sigma_N{}^2=\frac{E\left(r^2\right)-E^2\left(r\right)}{k^2}=\frac{\sigma^2}{k^2} \ . \quad (19)$$

The depth bit rate after DHP, R_N, can be calculated by replacing σ^2 in Eq.(12) with σ_N^2, and then applying Eq.(19) to R_N, which gives

$$R_N=\log\frac{\sqrt{2}e\sigma_N}{Q}=R-\log k \ . \quad (20)$$

Eq.(20) shows that the bit rate R_N of the proposed scheme is equal to the original R when k is 1. Moreover, the bit rate saving $R\text{-}R_N$ is achieved when $k>1$ and it increases as k increases.

2) *Distortion Analysis for Depth Coding*

While the depth maps are encoded by the conventional depth encoder, the distortion D_{ENC} between the original and reconstructed depth maps $\mathbf{X}\overset{Enc}{\Rightarrow}\tilde{\mathbf{X}}$ is $D_{ENC}=\left\|\mathbf{X}-\tilde{\mathbf{X}}\right\|$. In particular, the Mean Squared Error (MSE) between $\mathbf{X}$ and $\tilde{\mathbf{X}}$ can be calculated as

$$MSE_{ENC}=\frac{1}{N}\sum\left(X_i-\tilde{X}_i\right)^2, \tag{21}$$

where X_i is the depth value of the original depth map $\mathbf{X}$ and $\tilde{X}_i$ is the depth value of the reconstructed depth map $\tilde{\mathbf{X}}$ after lossy coding, i is an index of depth pixel and N is the total number of pixels. This distortion MSE_{ENC} is caused by quantization, which is zero in lossless coding and becomes larger as the quantization parameter increases in lossy coding.

While encoding the re-projected depth map, the process is $\mathbf{X}\overset{\otimes\mathbf{M}}{\Rightarrow}\mathbf{Y}\overset{Enc}{\Rightarrow}\tilde{\mathbf{Y}}\overset{\oplus\mathbf{M}}{\Rightarrow}\mathbf{Z}$, where $\mathbf{Y}$ is the depth map after DHP from $\mathbf{X}$, $\tilde{\mathbf{Y}}$ is the reconstructed depth map from encoding $\mathbf{Y}$, and $\mathbf{Z}$ is converted from $\tilde{\mathbf{Y}}$ with inverse DHP with one array $\mathbf{M}$s. The new distortion between the original and reconstructed depth maps is $D_N=\left\|\mathbf{Z}-\mathbf{X}\right\|$, whose MSE_N can be expressed as

$$MSE_N=\frac{1}{N}\sum\left(X_i-Z_i\right)^2, \tag{22}$$

where Z_i is the depth value of the reconstructed depth map $\mathbf{Z}$. The encoding distortion $MSE_{N,ENC}$ between $\mathbf{Y}$ and $\tilde{\mathbf{Y}}$ can be expressed as

$$MSE_{N,ENC}=\frac{1}{N}\sum\left(Y_i-\tilde{Y}_i\right)^2, \tag{23}$$

where Y_i and $\tilde{Y}_i$ are depth values in $\mathbf{Y}$ and $\tilde{\mathbf{Y}}$, respectively. Thus, similar to Eq.(14), Y_i and $\tilde{Y}_i$ can be expressed as

$$\begin{cases} Y_i=\dfrac{1}{k}\left(X_i-s_{init}\right)+\varepsilon_i \\ \tilde{Y}_i=\dfrac{1}{k}\left(Z_i-s_{init}\right)+\varepsilon_i \end{cases}. \tag{24}$$

Applying Eq.(24) and Eq.(22) to Eq.(23), we obtain

$$MSE_{N,ENC}=\frac{1}{k^2}\frac{1}{N}\sum\left(X_i-Z_i\right)^2=\frac{1}{k^2}MSE_N. \tag{25}$$

Suppose we have the same quantization step and distortion in encoding the original depth map $\mathbf{X}$ and the projected depth map $\mathbf{Y}$, *i.e.*, $MSE_{N,ENC}=MSE_{ENC}$. Then, we get

$$MSE_N=k^2\times MSE_{ENC}, \tag{26}$$

which means the quantization distortion in terms of MSE in encoding the depth map $\mathbf{X}$ will be increased k^2 times if processed with DHP. Although DHP followed by inverse DHP is lossless, the depth encoding distortion for lossy coding is introduced after the DHP, which will then be enlarged by the inverse DHP.

Fig.6 shows an example of the additional distortion cost caused by DHP in lossy depth coding. We observe that the lossy coding may introduce some distortions to the image and histogram map. However, if the histogram projection is activated, the depth distortion will be increased in the inverse projection, such as the red bar in the bottom subfigure in Fig.6, which is k^2 times the coding distortion MSE_{ENC}, as indicated in Eq. (26). For lossy coding, the compression distortion will be increased k^2 times on average with DHP.

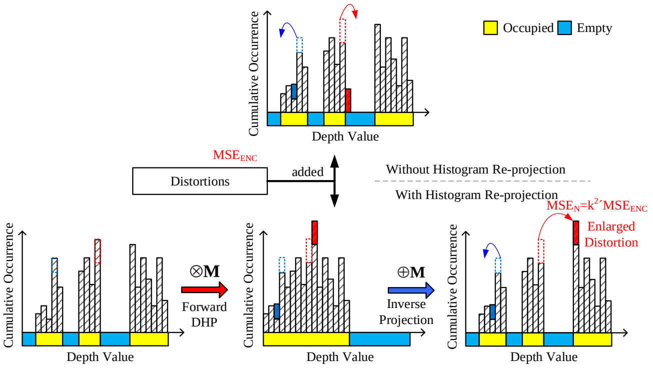


Fig.6. Distortion cost in lossy depth coding based on the DHP.

3) *Theoretical Rate-Distortion Cost Analyses for the DHP based Depth Coding*

The objective of depth coding is to minimize the distortion subject to a given bit rate. The distortion refers to the depth distortion or the view synthesis distortion in 3DVC [33][43]. Therefore, the RD cost can be expressed as

$$J=f\left(D\right)+\lambda R, \tag{27}$$

where R is the coding bit rate, λ is a Lagrange multiplier, D is depth distortion, and $f(D)$ is the synthesized distortion. Since the view synthesis distortion can be approximately modelled as a linear function of the depth distortion D [33], $f()$ is a linear mapping, *i.e.*, $f(D)=\chi D$, where the parameter χ is 1 for depth distortion and an arbitrary real number for view synthesis distortion. Similarly, the RD cost for the DHP based depth coding is

$$J_N=f\left(D_N\right)+\lambda R_N, \tag{28}$$

where D_N and R_N indicate the distortion and bit rate of coding the histogram re-projected depth maps, respectively. In addition, if DHP can improve the depth coding efficiency, the new RD cost J_N will be smaller than J, *i.e.*, $J_N-J\le 0$. Applying Eq.(20) and Eq.(26) to Eq.(28), this requirement can be written as

$$\begin{aligned} J_N-J&=k^2f\left(D\right)+\lambda\left(R-\log k\right)-f\left(D\right)-\lambda R \\ &=\left(k^2-1\right)f\left(D\right)-\lambda\log k\le 0 \end{aligned}, \tag{29}$$

where k is a real number defined in Eq.(9) and it is greater than or equal to 1, and distortion D is measured with MSE. We distinguish the following three cases satisfying Eq.(29):

1) When k=1, the equality is achieved, *i.e., J_N –J* = 0. In this case, DHP is inactivated and the depth coding performance is the same as that of the original depth encoder.
2) When *f(D)*=0, *inequality* Eq.(29) is satisfied since J_N-J=0 -$\lambda\log k\le 0$ when $k\ge 1$. Here, *f(D)*=0 means the depth distortion or the view synthesis distortion is zero, *i.e.*, the depth coding is lossless or view synthesis quality lossless. So, in this case, the depth coding performance using the DHP can be improved when k is larger than 1.
3) There are some other possible conditional solutions that may satisfy this *inequality*. For example, the inequality

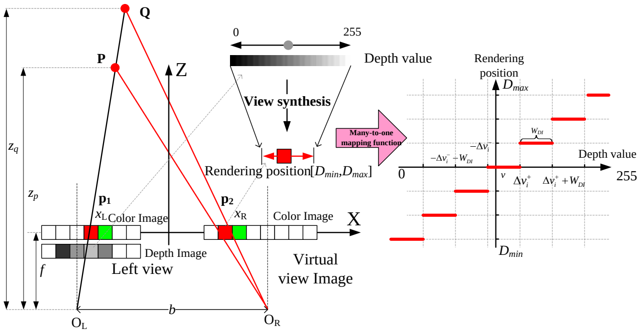


Fig.7. Piecewise relation between depth value and rendering position.

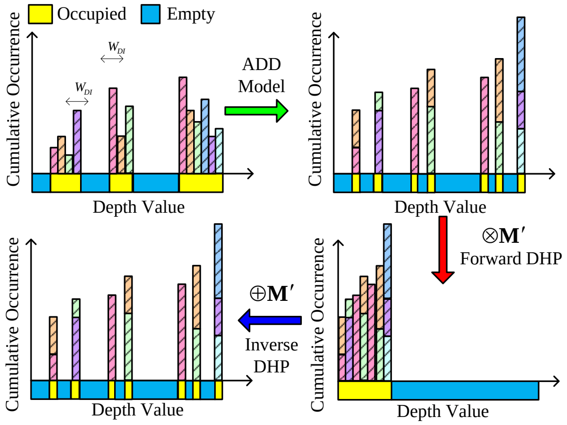


Fig.8. Example of ADD based DHP.

may be satisfied when λ is sufficiently large by using a large quantization. Or in the view synthesis oriented depth coding, the inequality may be satisfied when χ in $f(D)$ is small enough, which means the depth distortion has little impact on the view synthesis quality. This situation could happen in textureless regions in 3D video, which is an extreme case. However, this category of solutions has many uncertainties that will correlate with the coding techniques, video contents and rendering process.

In this paper, two techniques based on solution 2) are used to improve the depth coding performance. The first one is the histogram projection based lossless depth coding using large k, which was presented in Section III.A. The second one is the depth coding with lossless view synthesis quality, which further enlarges the values of k by exploiting view synthesis redundancies. It will be presented in Section IV.

IV. DHP Plus ADD Model Depth Coding for Lossless View Synthesis Quality

Since the depth map is used for virtual view rendering, the ultimate goal of depth coding is to minimize the depth bit rate while maintaining the same view synthesis quality. There exist depth redundancies in view synthesis that can be considered to improve depth coding efficiency. As shown in Fig.7, when the red pixel in the left view is rendered to the right virtual view, the depth map pixel corresponding to the red pixel provides the depth information. If it is distorted via depth coding or processing, other pixels will be mapped to the red pixel in the virtual view image, which causes geometrical distortion. Fortunately, when the depth value varies from z_p to z_q, the 3D point varies from **P** to **Q** in the world coordinate system. These points will be projected to the same red pixel in the right view, *i.e.*, the view synthesis quality will not be affected. Mapping 256 levels of depth values to a small number of rendering positions in [D_{min}, D_{max}] may result in a many-to-one mapping function [43][44]. Due to the mismatch between the depth value and the rendering offset/position, as shown in Fig.1 and Fig.7, there are redundancies in view synthesis, so called ADD, which will be exploited for depth coding.

In DIBR, the virtual view image pixel $\mathbf{p}_2=[a,b,c]^T$ can be rendered from its neighboring reference image pixel $\mathbf{p}_1=[x,y,1]^T$ as [2]

$$\mathbf{p_2} = z_1\mathbf{A_1R_2R_1^{-1}A_1^{-1}p_1} - \mathbf{A_2R_2R_1^{-1}t_1} + \mathbf{A_2t_2}\,, \qquad (30)$$

where z_1 is the depth for $\mathbf{p}_1$, and $\mathbf{A}_1$ and $\mathbf{A}_2$ are two 3×3 matrices of camera intrinsic parameters for the virtual camera and real camera, respectively. $\mathbf{R}_1$ and $\mathbf{R}_2$ are the rotation matrices, $\mathbf{t}_1=[t_{10},t_{11},t_{12}]^T$ and $\mathbf{t}_2=[t_{20},t_{21},t_{22}]^T$ are the translation vectors. Suppose the real and virtual cameras are well calibrated, *i.e.*, $\mathbf{A}_1=\mathbf{A}_2=\begin{pmatrix} f_x & \lambda & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}$, $\mathbf{R}_1=\mathbf{R}_2$, f_x and f_y are focal lengths in horizontal and vertical directions, u_0 and v_0 are principal point offsets, λ is an axis skew. The location (U, V) of $\mathbf{p}_2$ in the virtual view, which is called rendering position, can be expressed as

$$\begin{cases} U = R\left(x - \dfrac{z_1 x + f_x\left(t_{20}-t_{10}\right)+\lambda\left(t_{21}-t_{11}\right)+u_0\left(t_{22}-t_{12}\right)}{z_1+t_{22}-t_{12}}\right) \\ V = R\left(y - \dfrac{z_1 y + f_y\left(t_{21}-t_{11}\right)+v_0\left(t_{22}-t_{12}\right)}{z_1+\left(t_{22}-t_{12}\right)}\right) \end{cases}, \qquad (31)$$

where $R(x)=\dfrac{\left\lfloor x2^m+2^{m-1}\right\rfloor}{2^m}$, m is the rendering precision, *e.g.*, 0 for integer, 1 for half-pixel and 2 for quarter-pixel precision. If the cameras are parallel and well calibrated, $t_{12}=t_{22}$, $t_{21}=t_{11}$ and $\lambda=0$. Then, Eq.(31) is rewritten as

$$\begin{cases} U = R\left(\dfrac{f_x d_x}{z_1}\right) \\ V = 0 \end{cases}, \qquad (32)$$

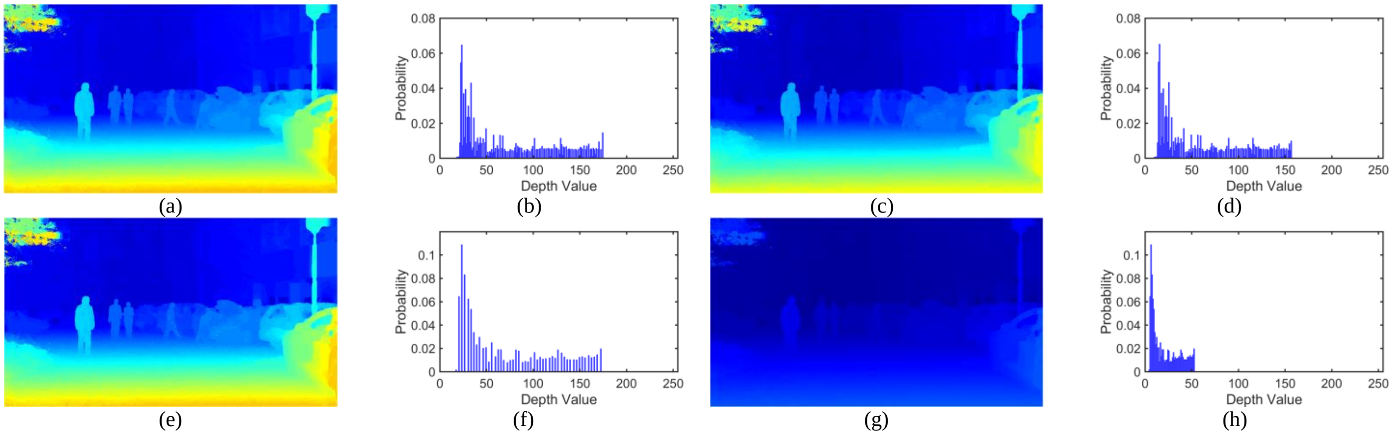
where $d_x=t_{20}-t_{10}$ is the baseline in the horizontal direction. Based on the depth quantization from depth z_1 to depth value v using Eq.(3), we can get a relation between the depth value v and the rendering horizontal position U from Eq.(32) as

$$U = R\left(d_x f_x\left(C_1 v + C_2\right)\right), \qquad (33)$$

where $C_1=\dfrac{1}{2^n}\left(\dfrac{1}{z_{near}}-\dfrac{1}{z_{far}}\right)$ and $C_2=\dfrac{1}{z_{far}}$. Due to the rounding operation $R()$, when there is a small change in the depth value v, *i.e.*, $v+\Delta v$ and $\Delta v\in[-\Delta v^-,\Delta v^+]$, the rendering position U may not change. So, a $\Delta v\in[-\Delta v^-,\Delta v^+]$ that does not change position U is the ADD in view synthesis, which leads to no-synthesis-error [42]. The range $W_{DI}=\Delta v^-+\Delta v^+$ can be expressed as [44]

$$W_{DI} = \left\lfloor \frac{1}{2^m L f_x C_1} - \zeta \right\rfloor + 1, \qquad (34)$$

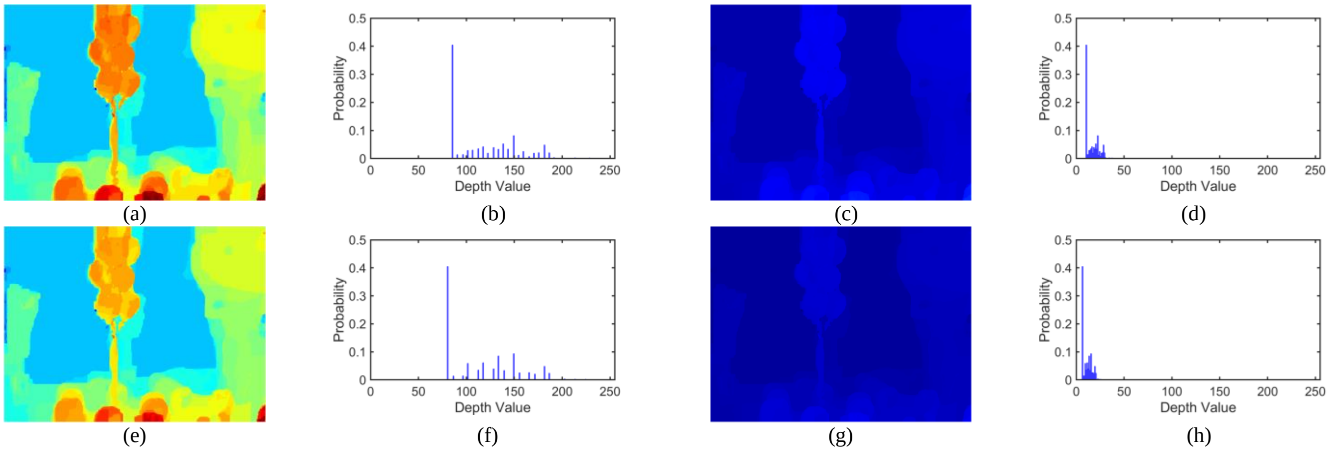
where L is the interval distance between the reference and



(a) (b) (c) (d)

(e) (f) (g) (h)

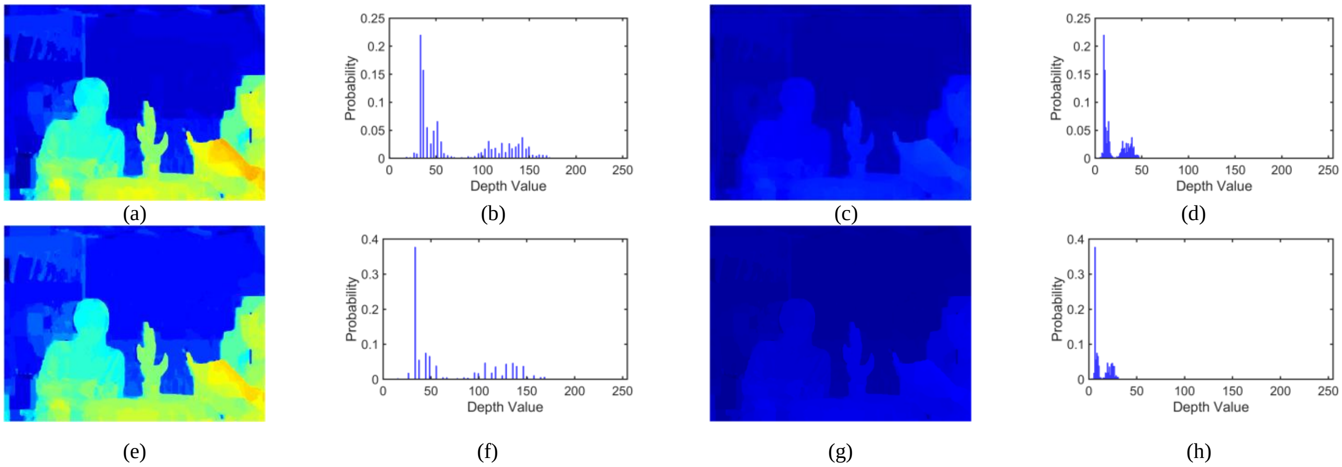
Fig.9. Histograms of the processed depth maps in *PoznanCarpark*. (a) original depth, RMSE=0 , (b) histogram of (a), (c) depth map with histogram projection, RMSE=10.61, (d) histogram of (c), (e) depth map with ADD, RMSE=1.49, (f) histogram of (e), (g) depth map with ADD and histogram projection, RMSE=61.05, (h) histogram of (g).



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig.10. Histograms of the processed depth maps in *Balloons*, (a) original depth, RMSE=0, (b) histogram of (a), (c) depth map with histogram projection, RMSE=107.96 (d) histogram of (c), (e) depth map with ADD, RMSE=6.25, (f) histogram of (e), (g) depth map with ADD and histogram projection, RMSE=113.8, (h) histogram of (g).



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 11. Histograms of the processed depth maps in *Newspapers*, (a) original depth, RMSE=0, (b) histogram of (a), (c) depth map with histogram projection, RMSE=58.66, (d) histogram of (c), (e) depth map with ADD, RMSE=9.19, (f) histogram of (e), (g) depth map with ADD and histogram projection, RMSE=66.67, (h) histogram of (g).

synthesized views, and ζ is a small positive constant. So, the depth value v and its neighbors $v+\Delta v$, $\Delta v\in[-\Delta v^{-},\Delta v^{+}]$ will map to the same rendering position. Meanwhile, if the depth distortion added to v is within the range $[-\Delta v^{-},\Delta v^{+}]$, the distortion will not affect the quality of synthesized videos, which can be exploited to further improve the coding efficiency of the DHP based depth coding.

Fig.8 shows an example of ADD based DHP. Based on the ADD model, the histogram bins are merged when their distances are within W_{DI} and the depth histogram becomes sparser for projection. The ADD model is mathematically lossy in depth, as we can observe that the final histogram is different from the original one. However, it is lossless in terms of view synthesis quality since the synthesized images rendered from the two depth maps are identical. Figs.9 to 11 show the processed depth maps and their histograms for *PoznanCarpark*, *Balloons* and *Newspapers*. In addition, the Root Mean Squared Error (RMSE) is calculated with respect to the ground truth. Comparing (b) with (d) and (f) with (h), we observe that the bins gather to the left part of the histogram, which becomes denser. Meanwhile, after processing by the ADD model, the number of bins is further reduced in the histogram when comparing (d) and (h). Note that (c) and (g) can be recovered as (a) and (e) respectively, via inverse DHP. The depth maps processed by ADD cannot be recovered to (a), which is a lossy process. However, the quality of synthesized videos from (e) will be the same as those rendered from (a).

V. Lossless Coefficients Encoding and the Scaling Factor Analysis

In Section V.A, we present the syntax design for encoding the coefficients in $\mathbf{M}_\mathrm{S}$ and in Section V.B, we analyze the scaling factor and its impacts on the coding performances in terms of different GOP lengths.

A. *Encoding Syntax of Coefficients in* $\boldsymbol{M_S}$

In HEVC, the video bit stream consists of a number of bits for parameter sets, which indicates the sequence and coding information, and coding bits for Intra or Inter frames. The parameter sets include the Video Parameter Set (VPS), Sequences Parameter Set (SPS), Picture Parameter Set (PPS) and Supplement Enhancement Information (SEI) [3]. The coefficients in $\mathbf{M}_\mathrm{S}$ are added to the SPS in Intra frames in a GOP and encoded with CABAC, which includes 1 flag bit indicating whether we use DHP or not, 8 bits indicating the start value s_{init}, and a number of bits for the number of non-empty and empty bins in the j^{th} GOP, $n_{\mathrm{bins}}(j)$ × (8+8), as shown in Table I. Each GOP corresponds to one row of the array $\mathbf{M}_\mathrm{S}$, *i.e.*, $\mathbf{M}_\mathrm{S}[i]$. Here, 8 bits are used to represent the number of empty/non-empty bins. These numbers could be smaller than 2^8 for fewer bits while the number of empty/non-empty bins is much smaller based on the statistics. Therefore, the total number of bits is $\sum_{j=1}^{N_{GOP}}\left[1+8+n_{bins}(j)\times(8+8)\right]$, where N_{GOP} and $n_{\mathrm{bins}}(j)$ are the numbers of GOPs and bins in the histogram of j^{th} GOP, respectively.

Fig.12 depicts the relationship between the coding bit rate of coefficients $\mathbf{M}_\mathrm{S}$ and the GOP length, where the *y*-axis is the bit rate and *x*-axis is the GOP length in logarithmic scale. We can observe that the bit rate of coefficients $\mathbf{M}_\mathrm{S}$ is less than 1.6 kbps for all test sequences, which is relatively small. In addition, the bit rate decreases significantly as the GOP length increases, because fewer GOPs and coefficients were generated.

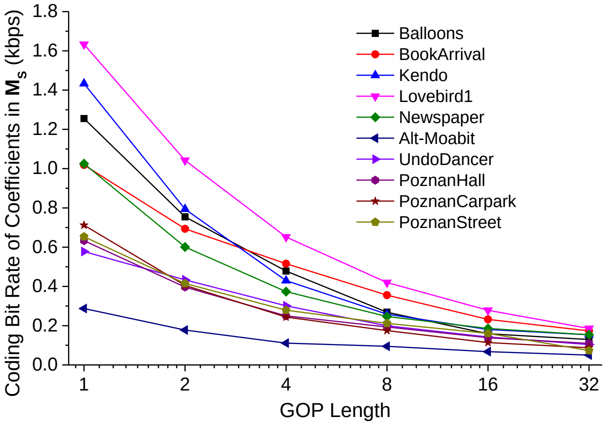


Fig.12. Relationship between coding bits of coefficient in $\mathbf{M}_\mathrm{S}$ and GOP length.

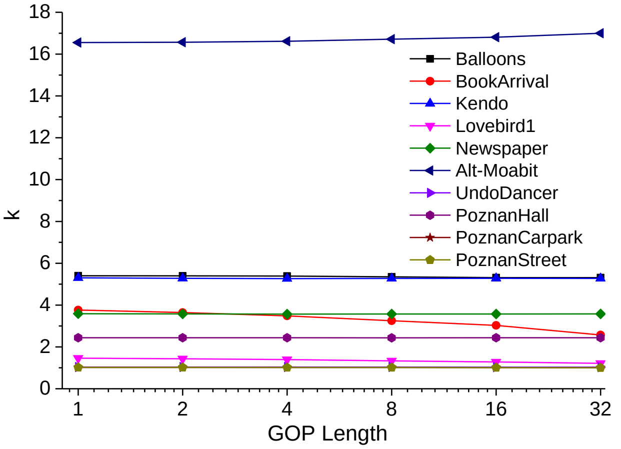


Fig.13. Relationship between *k* and GOP length.

Table I. The coding syntax of coefficients in $\mathbf{M}_\mathrm{S}$.

Histogram_Proj_ depth_coding_flag	u(1)
if(Histogram_ Proj_ depth_coding_flag){	
Initial_start s_{init};	u(8)
for(i=0;i<num_bins;i++){	
Num_non_empty_bin;	u(8)
Num_empty_bin;	u(8)
}	

B. *Relation between Scaling Factor k and GOP Length*

From the above analysis, it is found that when the scaling factor *k* increases, the depth bit rate R_N will decrease for lossless coding, *i.e.*, the coding gain increases. Factor *k* is actually determined by three key factors: (1) the depth map histogram which depends on the content and its generation method, (2) the ADD in view synthesis which correlates with the cameras, rendering position and rendering accuracy and (3) the number of depth frames and views in the histogram projection, *i.e.*, the GOP length. To improve inter and inter-view prediction in Inter coding, multiple depth frames of different time and views in a GOP are input to do DHP simultaneously and the coefficient array $\mathbf{M}_\mathrm{S}$ is calculated. For the all Intra coding settings, the array $\mathbf{M}_\mathrm{S}$ will be either GOP or frame based. In fact, the GOP size can be smaller than 1, which means the histogram is calculated for part of the depth map. For example, a GOP size of 1/2 means the histogram calculation unit is half the depth map. In this paper, we mainly consider the case where the GOP length is not smaller than 1.

Statistical experiments were performed to test the relationship between the scaling factor *k*, bit cost of the coefficients in $\mathbf{M}_\mathrm{S}$ and different GOP lengths of DHP. Fig.13 depicts the relationship between *k* and the GOP length for different test sequences. We can observe that *k* varies from 1 to 17 and depends on the sequences. For most of the sequences, *k* is almost the same as the GOP length increases. We conclude that *k* is generally dependent on the properties of the depth content and has less impact on the GOP size. If the GOP size

Table II. Parameters and settings for 3D video sequences.

Sequences	Provider	Properties	Resolution	Frame Rate	Baseline	Encoded Views	Synthesized View
BookArrival	HHI	Indoor, Stereo-matching	1024×768	16.67 fps	6.5cm	6,10	8
Alt-Moabit		Outdoor, Stereo-matching					
Lovebird1	ETRI	Outdoor, slow motion		30fps	3.5cm	4,8	6
Kendo	Nagoya University	Indoor, Stereo-matching	1024×768	29.4fps	5cm	1,5	3
Balloons		Indoor, Stereo-matching					
Newspapers	GIST	Enhanced depth map from TOF depth camera	1920×1080	30fps	5cm	2, 6	4
PoznanCarpark	Poznan University	Stereo-matching, enhanced	1920×1088	25 fps	13.75cm	3, 5	4
PoznanHall2		Indoor, Stereo-matching, enhanced				5,7	6
PoznanStreet		Stereo-matching enhanced				3,5	4
UndoDancer	Nokia	Computer graphic animation	1920×1088	25fps	synthetic	1,9	5

decreases, k may increase since the number of bins will likely be reduced as the number of pixels in the histogram calculation unit decreases.

VI. Experimental Results and Analysis

To assess the coding efficiency of the proposed methods, experiments were performed in three phases. First, the coding performance with the depth histogram projection was validated. Second, the coding performance with ADD plus DHP was tested. Two coding experiments were conducted on the latest multiview and 3D video coding reference model, which is the test model version 16.3 (HTM-16.3) [5][48] configured with MV-HEVC and 3D-HEVC. Both Intra and Inter frame coding were tested. Third, in addition to MV-HEVC and 3D-HEVC, we also encoded the original and processed depth maps with other lossless coding standards for static images, including JPEG2000 and HEVC Intra coding, to test the coding efficiency of the proposed methods. Three benchmark schemes, including NDR [18] and the scheme in [44] for preprocessing, Intra depth wedge plus intra contour scheme [4] (denoted as DWC) for coding optimization, were implemented and compared on different reference platforms. Ten standard 3D sequences *Bookarrival, Alt-Moabit, Lovebird1, Kendo, Balloons, Newspapers, PoznanCapark, PoznanHall2, PoznanStreet* and *Undodancers*, were used in the coding experiments. These sequences have various contents, texture, camera settings and properties. 96 frames per view were encoded. Details of the 3D video sequences are given in Table II. The two views of depth maps were encoded in the lossless scenario without quality degradation, where the two parameters *TransquantBypassEnableFlag* and *CUTransquantBypassFlag-Force* were fixed as 1 in the two-view coding configuration. IBBBP coding structure was used in 3D and MV-HEVC. The encoded views and synthesized views are shown in the rightmost two columns in Table II. In addition, the coefficients $\mathbf{M}_S$ of DHP were also encoded in lossless mode. A workstation running an Intel Core i7-6950X CPU, with a 64GB memory, Windows 10 Enterprise 64-bit operating system, was used as the computing platform in the experiments.

A. Coding Performance on MV-HEVC and 3D-HEVC with DHP

The performance of DHP-based depth coding is evaluated first. Since a two-view coding configuration is adopted in MV-HEVC, the two views of depth information are processed together by DHP to share the histogram. Two views of depth information are encoded by MV-HEVC jointly with inter-view

Table III. Comparison of coding performance between MV-HEVC and the proposed MV-HEVC with DHP for GOP length 8.

Seq.	Depth Bit Rate R_{Org} (kbps)	NDR*[18]		Proposed DHP	
		Depth PSNR (dB)	P_{NDR}(%)	*O (%)*	P_{DHP}(%)
Balloons	11599.472	47.06	2.92	0.0035	33.90
BookArrival	4613.504	45.03	6.11	0.0052	20.59
Kendo	8979.63	46.16	5.66	0.0046	36.73
Lovebird1	4937.47	43.79	21.58	0.0093	8.59
Newspapers	10285.67	44.77	8.08	0.0035	31.57
Alt-Moabit	2141.291	45.85	1.87	0.0049	51.46
PoznanCarpark	29438.85	45.42	16.28	0.0006	0.28
UndoDancer	6497.19	45.22	20.21	0.0030	-0.79
PoznanHall2	3753.42	43.81	16.57	0.0068	24.05
PoznanStreet	24020.97	45.05	26.08	0.0011	0.16
Average			**12.54**	**0.0042**	**20.66**

*Note that NDR is lossy and non-reversible projection for the depth maps, which may cause depth distortion. The projected depth maps with NDR were then encoded with lossless coding for comparison.

Table IV. Comparison of coding performance between MV-HEVC and the proposed MV-HEVC with DHP for GOP length 16.

Seq.	Depth Bit Rate R_{Org} (kbps)	NDR[18]		Proposed DHP	
		Depth PSNR (dB)	P_{NDR}(%)	*O (%)*	P_{DHP}(%)
Balloons	11519.79	47.06	2.85	0.0021	34.07
BookArrival	4301.253	45.03	6.25	0.0036	20.82
Kendo	8950.27	46.16	5.61	0.0032	36.69
Lovebird1	3839.44	43.79	20.38	0.0076	5.10
Newspapers	9710.56	44.77	7.80	0.0028	31.20
Alt-Moabit	1835.883	45.85	0.91	0.0040	50.60
PoznanCarpark	25792.82	45.42	15.09	0.0004	0.26
UndoDancer	6426.15	45.22	19.92	0.0022	-0.50
PoznanHall2	3759.77	43.81	16.50	0.0048	24.00
PoznanStreet	21089.48	45.05	26.25	0.0009	0.07
Average			**12.16**	**0.0032**	**20.23**

Table V. Comparison of coding performance between MV-HEVC and the proposed MV-HEVC with DHP for GOP length 32.

Seq.	Depth Bit Rate R_{Org} (kbps)	NDR[18]		Proposed DHP	
		Depth PSNR (dB)	P_{NDR}(%)	*O* (%)	P_{DHP}(%)
Balloons	11439.10	47.06	2.84	0.0017	34.16
BookArrival	4131.659	45.03	6.35	0.0027	16.96
Kendo	8913.07	46.16	5.53	0.0027	36.50
Lovebird1	3287.73	43.79	19.45	0.0058	2.46
Newspapers	9388.55	44.77	7.60	0.0024	30.88
Alt-Moabit	1680.672	45.85	0.22	0.0032	49.92
PoznanCarpark	23921.73	45.42	14.33	0.0004	0.37
UndoDancer	6382.02	45.22	20.18	0.0016	-0.05
PoznanHall2	3755.27	43.81	16.38	0.0038	23.91
PoznanStreet	19577.29	45.05	26.33	0.0004	0.02
Average			**11.92**	**0.0025**	**19.51**

prediction. Because this work focuses on depth coding, the bit rate of texture/color information is not recorded. Only depth bit rate is counted and compared to evaluate the depth coding efficiency. As for the lossless depth coding, the reconstructed depth map quality is identical to the original one. The depth bit rate saving ratio (P_Φ) is used to indicate the coding gain of the proposed depth coding and benchmark algorithms. P_Φ is calculated as

$$P_\Phi = \frac{R_{Org} - R_\Phi - \eta R_O}{R_{Org}} \times 100\% , \tag{35}$$

where R_{Org} is the depth bit rate with the original depth coding scheme using 3D-HEVC, MV-HEVC, HEVC, or JPEG2000, R_Φ is the depth bit rate with the proposed algorithm or benchmark schemes when they are applied on 3D-HEVC, MV-HEVC, HEVC, or JPEG2000 platforms, $\Phi \in$ {NDR [18], DWC [4], scheme in [44], Proposed DHP, Proposed ADD+ DHP}, R_O is the overhead bit rate for coding the coefficient array **M**s, η is 1 for the proposed DHP or ADD+DHP and 0 for the benchmark schemes. Meanwhile, the ratio of overhead bits (O) of the proposed algorithm is

$$O = \frac{R_O}{R_\Phi + R_O} \times 100\% \ , \tag{36}$$

where $\Phi \in$ { Proposed DHP, Proposed ADD+DHP}.

Tables III to V compare the depth bit rate of MV-HEVC, NDR [18] with MV-HEVC and that of MV-HEVC with the proposed DHP for GOP lengths varying from 8 to 32. In NDR with MV-HEVC, the depth maps were projected with NDR and then encoded with lossless MV-HEVC. Note that NDR is a lossy and non-reversible projection for the depth maps, which causes depth distortion, as shown in the PSNR column. Since the NDR was applied to pre-process all frames in one sequence, their depth quality degradations are irrelevant and independent of the GOP length settings. The depth rate R_{Org} in the second column is the total depth bit rate of two views. We can observe that compared with the MV-HEVC, the NDR can achieve bit rate savings from 1.87% to 26.08% and 12.54% on average for GOP length 8. However, although the coding is lossless, the depth quality degrades from 43.79dB to 47.06 dB while using the NDR projection. Similarly, it achieves 12.16% and 11.92% bit rate saving on average when GOP lengths are 16 and 32.

Moreover, three observations can be made for the proposed DHP: 1) The proposed DHP based depth coding can save bit rate from -0.79% to 51.47%, and 20.66% on average compared with the original depth coded by MV-HEVC when the GOP length is 8. Similarly, it achieves 20.20%, and 19.52% on average bit rate saving when GOP lengths are 16 and 32, which is slightly smaller than that of GOP length 8. It is because k will slightly decrease as the GOP length increases. Compared with the NDR, the proposed DHP can achieve more bit rate savings while maintaining lossless depth quality. 2) For some sequences such as *Balloons, Alt-Moabit, Newspapers*, and *Kendo*, the bit rate saving ratios varied from 30.88% to 51.47%, which is significant. This is because these depth sequences are generated from stereo-matching and have very sparse histograms which leads to a larger k. For *PoznanCarpark, UndoDancer* and *PoznanStreet* sequences, their gains are less than 1%. This is because these depth sequences are generated with computer graphics or stereo-matching with complicated post-processing. They have dense or continuous histograms. Their ks approach 1 and the room for depth bit rate saving by using DHP is very limited. 3) The overhead bit ratios over different settings are 0.0042%, 0.0032% and 0.0025% on average, respectively, which are negligible. In addition, fewer overhead bits are required for the larger GOP size, because more frames share the projection coefficients.

In addition to the depth coding experiments on the MV-HEVC, comparative studies on 3D-HEVC were also performed under two-view plus depth configuration. Besides the depth maps, the associated color videos were also encoded in lossless. The parameters *TransquantBypassEnableFlag* and *CUTransquantBypassFlagForce* were set as 1 for the lossless scenario. Depth coding tools in 3D-HEVC, such as VSO among color and depth channels, *IntraWedgeFlag* and *IntraContourFlag*, were disabled and it was regarded as the anchor 3D-HEVC. Meanwhile, the proposed DHP and two benchmark schemes, *i.e.*, NDR [18] and DWC [4], were implemented on the anchor 3D-HEVC and compared. In NDR, the depth maps were pre-processed with NDR and then encoded with lossless 3D-HEVC. In the DWC scheme, both *IntraWedgeFlag* and *IntraContourFlag* were enabled. Two views and 96 frames per view were encoded with GOP size 8 and the remaining settings were the default ones. Note that the GOP length for DHP in this experiment is 96, which reduces the bit rate of coefficient array **M**s.

Table VI shows depth coding performance comparisons between the proposed DHP and the benchmarks on 3D-HEVC, where the associated color videos were encoded with the anchor 3D-HEVC and unchanged among different depth coding schemes. We can observe that the NDR can achieve bit rate savings from 0.31% to 25.52%, and 11.46% on average, which is similar to those achieved in MV-HEVC. Similarly, the depth quality degradations from 43.79dB to 47.06 dB are caused by the non-linear and non-reversible NDR. For the DWC, it achieves bit rate savings from 5.53% to 21.00%, and 10.43% on average as compared with the anchor 3D-HEVC in lossless depth coding.

The proposed DHP achieves depth bit rate saving from 0.02% to 44.70% and 14.30% on average as compared with the anchor 3D-HEVC, which outperforms the NDR and DWC schemes. Meanwhile, the number of overhead bits is negligible. Note that the coding gain achieved by DHP over MV-HEVC is smaller. There are two main reasons for this: 1) the depth coding tools in 3D-HEVC already exploit some depth redundancies and reduce the original depth rate R_{Org}. 2). The GOP length for DHP is 96, which leads to smaller k and less representation redundancy is exploited.

B. *Coding Performance on MV-HEVC with ADD plus DHP*

The coding performance on MV-HEVC with ADD plus DHP was also evaluated and compared with the original lossless MV-HEVC. In rendering the virtual view image while using the reconstructed depth maps, View Synthesis Reference Software (VSRS) was used with *1DFast* mode, *HoleFillingMode* = 1, *RenderDirection* = 0, and *BlendMode* = 0, which are default settings. For view synthesis, the middle view was synthesized from the left and right views, as shown in Table II. Three rendering precisions were tested in VSRS when synthesizes the virtual views, where *ShiftPrecision* is 0, 1 and 2

Table VI. Depth bit rate saving between the proposed DHP and the benchmarks on 3D-HEVC.

Sequences	Anchor 3D-HEVC		NDR[18]			Depth wedge plus contour [4] [10]		Proposed DHP		
	Color Bit Rate (kbps)	Depth Bit Rate R_{Org} (kbps)	Depth PSNR (dB)	Depth Bit Rate R_{NDR} (kbps)	Saving Ratio P_{NDR} (%)	Depth Bit Rate R_{DWC} (kbps)	Saving Ratio P_{DWC} (%)	Depth Bit Rate R_{DHP} (kbps)	Overhead O (%)	Saving Ratio P_{DHP}(%)
Balloons	187529.2	9514.08	47.06	9245.19	2.83	8887.86	6.58	6901.57	0.0017	27.46
BookArrival	122034.9	3727.2	45.03	3540.87	5.00	3221.02	13.58	3509.66	0.0036	5.84
Kendo	172882.3	7456.24	46.16	7052.25	5.42	6627.07	11.12	5346.36	0.0025	28.30
Lovebird1	172133.7	3188.88	43.79	2650.8	16.87	2778.81	12.86	3104.91	0.0044	2.63
Newspapers	160624.3	8091.31	44.77	7598.29	6.09	7259.82	10.28	6189.46	0.0021	23.50
Alt-Moabit	117531.5	1541.76	45.85	1537.00	0.31	1217.99	21.00	852.56	0.0052	44.70
PoznanCarpark	463548	23632.3	45.42	20201.88	14.52	22295.67	5.66	23551.3	0.0003	0.34
UndoDancer	250793.9	4290.34	45.22	3314.29	22.75	4053.13	5.53	4289.15	0.0018	0.03
PoznanHall2	414769	3322.77	43.81	2812.85	15.35	3028.15	8.87	2985.95	0.0031	10.14
PoznanStreet	452574	19338.99	45.05	14404.16	25.52	17632.02	8.83	19334.99	0.0004	0.02
Average					**11.46**		**10.43**		**0.0025**	**14.30**

Table VII. Depth bit rate saving on MV-HEVC with ADD plus DHP under different rendering precisions (GOP length is 8). [Unit:%].

Seq.	Coded/ Rendered Views	Integer Pixel			Half Pixel			Quarter Pixel		
		[44]	Proposed ADD plus DHP		[44]	Proposed ADD plus DHP		[44]	Proposed ADD plus DHP	
		Saving Ratio $P_{[44]}$ (%)	Saving Ratio $P_{\mathrm{DHP+ADD}}$(%)	Overhead O(%)	Saving Ratio $P_{[44]}$(%)	Saving Ratio $P_{\mathrm{DHP+ADD}}$ (%)	Overhead O(%)	Saving Ratio $P_{[44]}$ (%)	Saving Ratio $P_{\mathrm{DHP+ADD}}$(%)	Overhead O(%)
Balloons	1,5/3	18.19	33.90	0.0027	0.71	33.90	0.0035	-0.46	33.90	0.0036
BookArrival	6,10/8	13.78	50.16	0.0030	-0.10	24.38	0.0033	-0.09	21.70	0.0048
Kendo	1,5/3	17.00	36.74	0.0043	0.47	36.74	0.0047	-0.43	36.74	0.0047
Lovebird1	4,8/6	28.31	52.32	0.0106	5.67	26.64	0.0090	0.00	8.59	0.0093
Newspapers	2,6/4	18.92	31.57	0.0031	-0.48	31.57	0.0035	0.00	31.57	0.0035
Alt-Moabit	8,10/9	54.37	65.71	0.0073	-0.33	51.47	0.0056	-0.79	51.47	0.0055
PoznanCarpark	3,5/4	16.54	40.01	0.0009	5.26	17.79	0.0007	0.00	0.28	0.0006
UndoDancer	1,9/5	77.74	80.45	0.0111	66.91	70.19	0.0090	48.72	54.11	0.0074
PoznanHall2	5,7/6	0.00	24.06	0.0068	0.00	24.06	0.0068	0.00	24.06	0.0068
PoznanStreet	3,5/4	26.66	51.98	0.0016	11.69	24.41	0.0012	0.00	24.41	0.0014
Average		**27.15**	**46.69**	**0.0051**	**8.98**	**34.12**	**0.0047**	**4.70**	**28.68**	**0.0048**

Table VIII. Depth bit rate saving (P_{DHP} and $P_{\mathrm{DHP+ADD}}$) on JPEG2000 with the proposed DHP and ADD plus DHP. [Unit:%].

Seq.	DHP			ADD plus DHP		
	GOP8	GOP16	GOP32	Integer	Half	Quarter
Balloons	58.53	58.52	58.52	67.34	58.53	58.53
BookArrival	41.32	39.12	33.06	63.78	47.53	43.25
Kendo	59.82	59.82	59.82	68.20	59.82	59.82
Lovebird1	23.57	21.60	16.88	57.11	36.34	23.57
Newspapers	52.00	52.00	52.00	63.83	52.00	52.00
Alt-Moabit	76.29	76.29	76.29	89.45	76.29	76.29
PoznanCarpark	0.90	0.85	0.80	50.31	25.61	0.90
UndoDancer	0.31	0.30	0.29	78.43	65.69	48.16
PoznanHall2	46.26	46.26	46.26	46.26	46.26	46.26
PoznanStreet	0.70	0.51	0.21	54.01	27.01	0.70
Average	**35.97**	**35.53**	**34.41**	**63.87**	**49.51**	**40.95**

Table IX. Depth bit rate saving (P_{DHP} and $P_{\mathrm{DHP+ADD}}$) on HEVC AI coding with the proposed DHP and ADD plus DHP. [Unit:%].

Seq.	DHP			ADD plus DHP		
	GOP8	GOP16	GOP32	Integer	Half	Quarter
Balloons	38.33	38.33	38.33	52.34	38.33	38.33
BookArrival	25.88	24.51	20.34	47.55	28.91	26.73
Kendo	37.11	37.11	37.11	50.48	37.11	37.11
Lovebird1	15.72	14.27	11.14	53.68	28.96	15.72
Newspapers	32.61	32.61	32.61	50.27	32.61	32.61
Alt-Moabit	53.46	53.46	53.46	80.84	53.46	53.46
PoznanCarpark	0.64	0.60	0.58	49.00	22.23	0.64
UndoDancer	0.14	0.14	0.13	79.22	68.79	52.71
PoznanHall2	22.17	22.17	22.17	22.17	22.17	22.17
PoznanStreet	0.44	0.33	0.13	51.56	23.18	0.44
Average	**22.65**	**22.35**	**21.60**	**53.71**	**35.58**	**27.99**

representing integer pixel, 1/2 pixel and 1/4 pixel precision, respectively.

Table VII shows the bit rate saving for MV-HEVC with ADD plus DHP and the benchmark scheme [44] under different rendering precisions. In the coding results, it has been validated that the synthesized images rendered using the coded depth are identical to those rendered with the original depth information, *i.e.*, lossless view synthesis quality. We observe from Table VII that the scheme in [44] is able to achieve average bit rate savings of 27.15%, 8.98% and 4.70% when compared with the original depth map with the rendering precision of integer, half and quarter pixel, respectively. For the *Undodancer* sequence, a significant coding gain is achieved due to a large ADD. In addition, the proposed algorithm achieves average bit rate savings of 46.69%, 34.12%, and 28.68%, which significantly outperforms the scheme in [44]. As the rendering precision increases from integer to quarter pixel, the bit rate saving *P* decreases. This is because as the rendering precision becomes more accurate, the ADD interval becomes smaller, and the processed depth map approaches the original one. Overall, the coding gain becomes higher when combining DHP with ADD. In addition, the average overhead bit ratio is about 0.0050%, which is negligible. Although the depth maps have been distorted due to the ADD based projection, the synthesized videos are distortion free as compared with the videos synthesized from the original multiview depth and color videos.

C. Coding Performance of JPEG2000 and HEVC Intra coding with DHP and ADD plus DHP

In addition to the MV-HEVC and 3D-HEVC coding experiments, we also tested the depth processing algorithm with the commonly used static image coding standard JPEG2000 [37] and HEVC All Intra (AI) coding, which were set in lossless coding mode.

Table VIII shows the bit rate saving for JPEG2000 with DHP and ADD plus DHP. We can observe that when the depth maps are processed with DHP and coded by JPEG2000, the coding efficiency improves about 35.97%, 35.53% and 34.41% on average for GOP lengths 8, 16 and 32, respectively, as compared with the original depth map coded with lossless JPEG2000. The coding gains are similar and insensitive to GOP length. When the depth map is processed with ADD plus DHP, the bit rate savings are 63.87%, 49.51% and 40.95% on average for integer, half and quarter-pixel rendering precisions, which are higher compared with the savings with DHP only. Table IX shows the depth bit rate saving on lossless HEVC All Intra (AI) coding with DHP and ADD plus DHP. The proposed DHP based lossless HEVC encoder achieves 22.65%, 22.35% and 21.60% bit rate saving on average, respectively, when compared with the original HEVC encoder. In addition, when the depth is processed with ADD plus DHP, the bit rate saving achieves 53.71%, 35.58% and 27.99% on average respectively for different rendering precisions. The bit rate saving is larger for JPEG2000 than for HEVC AI coding. In summary, the coding performance is significantly improved for image lossless coding and HEVC AI coding. In addition, the proposed DHP and ADD are independent and can be individually or jointly applied to different image/video coding standards, such as HEVC and JPEG2000, to improve their lossless coding performances.

VII. Conclusions

We propose efficient lossless 3D depth coding algorithms based on Depth Histogram Projection (DHP) and Allowable Depth Distortion (ADD) in view synthesis. Firstly, we presented the problem of the current depth map and proposed DHP for depth coding. We theoretically analyzed the cost and gain of DHP based depth coding, and proved that significant coding gain can be expected. Secondly, since the depth map is used for rendering the virtual view, not every distortion in the depth maps will affect the quality of the rendered images, which is regarded as ADD in view synthesis. Based on this ADD model and DHP, we proposed depth coding with lossless view synthesis quality to further improve the depth coding efficiency. The experimental results showed that the proposed algorithm achieves significant coding gain in lossless depth coding when compared with the three state-of-the-art coding standards, 3D/MV-HEVC, HEVC and JPEG2000.

References

[1] G. Tech, Y. Chen, K. Müller, J.R. Ohm, A. Vetro, and Y. K. Wang, “Overview of the multiview and 3D extensions of high efficiency video coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol.26, no.1, pp. 35-49, Jan. 2016.

[2] C. Fehn, “Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV”, in Proc. SPIE, Stereoscopic Displays and Virtual Reality Systems XI, vol.5291, San Jose, USA, pp.93-104, 2004.

[3] D. Flynn, D. Marpe, M. Naccari, T. Nguyen, C. Rosewarne, K. Sharman *et al.*, “Overview of the range extensions for the HEVC standard: tools, profiles, and performance,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 4-19, Jan. 2016.

[4] P. Merkle, K. Müller, D. Marpe, and T. Wiegand, “Depth intra coding for 3D video based on geometric primitives,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 3, pp. 570-582, Mar. 2016.

[5] Y. Chen, G. Tech, K. Wegner, and S. Yea, “Test model 11 of 3D-HEVC and MV-HEVC,” JCT3V-K1003, *MPEG & VCEG JCT-3V*, May 2015.

[6] Y. Song and Y.-S. Ho, “Unified depth intra coding for 3D video extension of HEVC,” *Signal, Image & Video Process.*, vol. 8, no. 6, pp. 1031-1037, Sep. 2014.

[7] Y. Chen, H. Liu, Y. Cai, S. Ma, M. Li, and P. Wu, “Depth lookup table coding for 3D-HEVC,” JCT3V-F0131, *MPEG & VCEG JCT-3V*, Oct. 2013.

[8] X. Chen, X. Zheng ,Y. Lin, J. Zheng, S. Yoo, S. Yeo, *et al.*, “Single depth intra mode simplification,” JCT3V-J0115, *MPEG & VCEG JCT-3V*, Oct. 2014.

[9] F. Jager, M. Wien, Mathias Wien, and P. Kosse, “Model-based intra coding for depth maps in 3D video using a depth lookup table,” in *Proc. 3DTV-Conference (3DTV-CON)*, 2012.

[10] G. Sanchez, J. Silveira, L. Agostini, and C. Marcon, “Performance analysis of depth intra coding in 3D HEVC”, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no.8, pp.2509-2520. Aug. 2019.

[11] Y. Chan, C. Fu, H. Chen, and S. Tsang, “Overview of current development in depth map coding of 3D video and its future,” *IET Signal Process.*, vol. 14, no. 1, pp. 1-14, Feb. 2020.

[12] Z. Peng, H. Han, F. Chen, G. Jiang, and M. Yu, “Joint processing and fast encoding algorithm for multi-view depth video,” *EURASIP J. Image Video Process.* vol.2016, no.24, Sep. 2016.

[13] S. Shahriyar, M. Murshed, M. Ali and M. Paul, “Depth sequence coding with hierarchical partitioning and spatial-domain quantization,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 3, pp. 835-849, Mar. 2020.

[14] M. Georgiev and A. Gotchev, “Improved depth compression by depth downsampling guided by color super-pixel refinement segmentation,” in *Proc. Data Compress. Conf. (DCC)*, Snowbird, UT, 2018, pp. 409-409.

[15] S. Chen, Q. Liu and Y. Yang, “Multi-view multi-modality priors residual network of depth video enhancement for bandwidth limited asymmetric coding framework,” in *Proc. Data Compress. Conf. (DCC)*, Snowbird, UT, USA, 2019, pp. 560-560.

[16] J. Lei, X. Liu, K. Zhang, G. Li and N. Ling, “Convolutional neural network based up-sampling for depth video intra coding,” in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Sydney, Australia, 2019, pp. 1-4.

[17] Y. Yang, Q. Liu, X. He, and Z. Liu, “Cross-view multi-lateral filter for compressed multiview depth video,” *IEEE Trans. Image Process.,* vol.28, no.1, pp.302-315, Jan. 2019.

[18] O. Stankiewicz, K. Wegner and M. Domański, “Study of 3D video compression using nonlinear depth representation,” *IEEE Access*, vol. 7, pp. 31110-31122, Mar. 2019.

[19] M. Saldanha, G. Sanchez, C. Marcon and L. Agostini, “Fast 3D-HEVC depth map encoding using machine learning,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 3, pp. 850-861, Mar. 2020.

[20] J. Lei, J. Duan, F. Wu, N. Ling, and C. Hou , “Fast mode decision based on grayscale similarity and inter-view correlation for depth map coding in 3D-HEVC,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 3, pp. 706-718, Mar. 2018.

[21] 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.

[22] H. Zhang, Y. Zhang, H. Wang, and Y.-S. Ho, and S. Feng, “WLDISR: Weighted local sparse representation based depth image super-resolution scheme for 3D video system,” *IEEE Trans. Image Process.*, vol. 28, no.2, pp. 561-576, Feb. 2019.

[23] J. Lei, X. He, H. Yuan, F. Wu , N. Ling, and C. Hou, “Region adaptive R-λ model-based rate control for depth maps coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 6, pp.1390-1405, Jun. 2018.

[24] P. Gao, and M. Paul, “Rate-distortion optimal joint texture and depth map coding for 3D video streaming,” *IEEE Trans. Multimedia*, vol. 22, no. 3, pp. 610-625, Mar. 2020.

[25] J. Jin, J. Liang, Y. Zhao, C. Lin, C. Yao, and A. Wang, “A depth bin based graphical model for fast view synthesis distortion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 6, pp. 1754-1766, Jun. 2019.

[26] W.-S. Kim, A. Ortega, P.L. Lai, and D. Tian, “Depth map coding optimization using rendered view distortion for 3D video coding,” *IEEE Trans. Image Process.*, vol. 24, no.11, pp. 3534-3545, Nov. 2015.

[27] H. Yuan, S. Kwong, X. Wang, Y. Zhang, and F. Li, “A virtual view PSNR estimation method for 3-D videos,” *IEEE Trans. Broadcast.,* vol. 62, no.1, pp.134-140, Mar. 2016.

[28] Z. Zheng, J. Huo, B. Li, H. Yuan, and W. Lin, “A novel distortion criterion of rate-distortion optimization for depth map coding,” *J. Vis. Commun. Image R.*, vol. 54, pp. 145-154, Jul. 2018.

[29] M. Yang, C. Zhu, X. Lan and N. Zheng, “Efficient estimation of view synthesis distortion for depth coding optimization,” *IEEE Trans. Multimedia*, vol. 21, no. 4, pp. 863-874, Apr. 2019.

[30] J. Jin, J. Liang, Y. Zhao, C. Lin, C. Yao, and L. Meng, “Pixel level view synthesis distortion estimation for 3D video coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 7, pp. 2229-2239, Jul. 2020.

[31] H. Zhang, C. Fu, Y. Chan, S. Tsang and W. Siu, “Probability-based depth intra-mode skipping strategy and novel VSO metric for DMM decision in 3D-HEVC,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 2, pp. 513-527, Feb. 2018.

[32] Y. Yuan, G. Cheung, P. Le Callet, P. Frossard and H. V. Zhao, “Object shape approximation and contour adaptive depth image coding for virtual view synthesis,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 12, pp. 3437-3451, Dec. 2018.

[33] Y. Zhang, X. Yang, X. Liu, Y. Zhang, G. Jiang, and S. Kwong, “High efficiency 3D depth coding based on perceptual video quality of synthesized view,” *IEEE Trans. Image Process.*, vol. 25, no. 12, pp.5877-5891, Dec. 2016.

[34] Y. Zhang, H. Zhang, M. Yu, S. Kwong, and Y. S. Ho, “Sparse representation based video quality assessment for synthesized 3D videos,” *IEEE Trans. Image Process.*, vol.29, pp.509-524, Dec. 2020.

[35] X. Liu, Y. Zhang, S. Hu, S. Kwong, C.-C. J. Kuo, and Q. Peng, “Subjective and objective video quality assessment of 3D synthesized view with texture/depth compression distortion,” *IEEE Trans. Image Process.*, vol.24, no.12, pp.4847-4861, Dec. 2015.

[36] ISO/IEC 14495-1|ITU-T T.87, Information technology - Lossless and near-lossless compression of continuous-tone still images: Baseline, 1999.

[37] ISO/IEC IS 15444-5 | ITU-T T.804, JPEG 2000 image coding system: Reference software, 2003

[38] ISO/IEC IS 29199-2 | ITU-T T.832, JPEG XR image coding system - Image coding specification, 2012.

[39] K. Y. Kim, G.H. Park, and D. Y. Suh “Bit-plane-based lossless depth-map coding,” *Optical Eng.*, vol. 49, no.6, page 067403, Jun. 2010.

[40] J. Heo, and Y.-S. Ho, “Improved context based adaptive binary arithmetic coding over H.264/AVC for lossless depth map coding,” *IEEE Signal Process. Lett.*, vol. 17, no. 10, pp. 835-838, Oct. 2010.

[41] S. Shahriyar, M. Murshed, M. Ali, and M. Paul, “Lossless depth map coding using binary tree based decomposition and context-based arithmetic coding,” *IEEE Int’l Conf. Multimedia Expo (ICME)*, Seattle, WA, 2016, pp. 1-6.

[42] 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.

[43] Y. Zhang, S. Kwong, S. Hu, and C.-C.J. Kuo, “Efficient multiview depth coding optimization based on allowable depth distortion in view synthesis,” *IEEE Trans. Image Process.*, vol.23, no.11, pp.4879-4892, Nov. 2014.

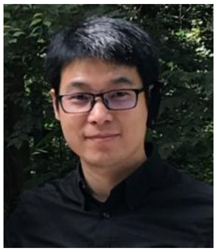
[44] Y. Zhang, L. Zhu, X. Liu, and G. Jiang, “Allowable depth distortion based depth filtering for high efficiency 3D video coding,” in *Proc. IEEE Int’l Symp. Circuits Syst. (ISCAS)*, Montreal, Canada, July 2016.

[45] P. Gao and A. Smolic, “Occlusion-aware depth map coding optimization using allowable depth map distortions,” *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5266-5280, Nov. 2019.

[46] H. Gish, J.N. Pierce, “Asymptotically efficient quantizing,” *IEEE Trans. Inf. Theory*, vol.14, no.5, pp. 676-683, Sep. 1968.

[47] L. Xu, X. Ji, W. Gao, and D. Zhao, “Laplacian distortion model (LDM) for rate control in video coding,” in *Proc. Pacific-Rim Conf. Multimedia (PCM)*, LNCS 4810, pp. 638-646, 2007.

[48] https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-16.3/



Yun Zhang (M’12-SM’16) 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 Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2010. From 2009 to 2014, he was a Visiting Scholar with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong. From 2010 to 2017, he was an Assistant Professor and an Associate Professor in Shenzhen Institutes of Advanced Technology (SIAT), CAS, where he is currently a Professor in SIAT, CAS, Shenzhen, China. His research interests are video compression, 3D video processing, visual perception and machine learning.

Linwei Zhu received received the B.S. degree in applied physics from Tianjin University of Technology, China, in 2010, the M.S. degree in signal and information processing from Ningbo University, China, in 2013, and the Ph.D. degree from the Department of Computer Science, City University of Hong Kong, Hong Kong, China, in 2019. Now, he is a Postdoctoral Fellow with Shenzhen Institutes of Advanced Technology (SIAT), Chinese Academy of Sciences (CAS). His research interests mainly include depth image based rendering, depth estimation and machine learning based video coding/transcoding.

Raouf Hamzaoui (M’02–SM’07) received the M.Sc. degree in mathematics from University of Montreal, Montréal, QC, Canada, in 1993; the Dr.rer.nat. degree from University of Freiburg, Freiburg im Breisgau, Germany, in 1997; and the Habilitation degree in computer science from University of Konstanz, Konstanz, Germany, in 2004. He was an Assistant Professor with the Department of Computer Science, University of Leipzig, Leipzig, Germany, and with the Department of Computer and Information Science, University of Konstanz. He joined De Montfort University, Leicester, U.K., in 2006, where he is currently a Professor in Media Technology. His research interests include image and video coding, multimedia communication systems, error control systems, and machine learning.

Sam Kwong (M’93–SM’04-F’13) received the B.S. and M.S. degrees in electrical engineering from the State University of New York at Buffalo in 1983, the University of Waterloo, Waterloo, ON, Canada, in 1985, and the Ph.D. degree from the University of 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 Member of Scientific Staff. In 1990, he became a Lecturer in the Department of Electronic Engineering, City University of Hong Kong, where he is currently a Professor in the Department of Computer Science. His research interests are video and image coding and evolutionary algorithms.

Yo-Sung Ho (SM’06–F’16) received the B.S. and M.S. degrees in electronic engineering from Seoul National University, Seoul, Korea, in 1981 and 1983, respectively, and the Ph.D. degree in electrical and computer engineering from the University of California, Santa Barbara, in 1990. He joined ETRI (Electronics and Telecommunications Research Institute), Daejeon, Korea, in 1983. From 1990 to 1993, he was with North America Philips Laboratories, Briarcliff Manor, New York, where he was involved in development of the Advanced Digital High-Definition Television (AD-HDTV) system. In 1993, he rejoined the technical staff of ETRI and was involved in development of the Korean DBS Digital Television and High-Definition Television systems. Since 1995, he has been with Gwangju Institute of Science and Technology (GIST), where he is currently Professor of School of Electrical Engineering and Computer Science. Since

August 2003, he has been Director of Realistic Broadcasting Research Center at GIST in Korea. He has served as Associate Editors of IEEE Transactions on Multimedia (T-MM) and IEEE Transactions on Circuits and Systems Video Technology (T-CSVT). His research interests include Digital Image and Video Coding, Image Analysis and Image Restoration, Three-dimensional Image Modeling and Representation, Advanced Source Coding Techniques, Augmented Reality (AR) and Virtual Reality (VR), Three-dimensional Television (3DTV), and Realistic Broadcasting Technologies. He is a Fellow of IEEE.