Plenoptic Point Cloud Compression Using Multiview Extension of High Efficiency Video Coding

Li Li, *Member, IEEE*, Zhu Li, *Senior Member, IEEE*, Shan Liu, *Senior Member, IEEE*, and Houqiang Li, *Fellow, IEEE*

***Abstract*—Plenoptic point clouds are more complete representations of three-dimensional (3-D) objects than single-color point clouds, as they can have multiple colors per spatial point, representing colors of each point as seen from different view angles. They are more realistic but also involve a larger volume of data in need of compression. Therefore, in this paper, a multiview-video-based framework is proposed to exploit the correlations in color across different viewpoints to compress plenoptic point clouds efficiently. To the best of the authors' knowledge, this is the first work to exploit correlations in color across different viewpoints using a multiview-video-based framework. In addition, it is observed that some unoccupied pixels, which do not have corresponding points in plenoptic point clouds and are of no use to the quality of the reconstructed plenoptic point cloud colors, may cost many bits. To address this problem, a block-based group smoothing and a combined occupancy-map-based rate distortion optimization and four-neighbor average residual padding are further proposed to reduce the bit cost of unoccupied color pixels. The proposed algorithms are implemented in the moving pictures experts group (MPEG) video-based point cloud compression (V-PCC) and multiview extension of High Efficiency Video Coding (MV-HEVC) reference software. Compared with the V-PCC independently applied to each view direction, the proposed algorithms can provide a BD-rate reduction of over 70%.**

***Index Terms*—Multiview extension of high efficiency video coding, plenoptic point cloud, point cloud compression, rate distortion optimization, video-based point cloud compression.**

I. Introduction

A POINT cloud is a set of three-dimensional (3-D) points that can be used to represent a two-dimensional (2-D) surface embedded in 3-D space. Each point has a spatial position (x, y, z) and a vector of attributes such as colors and material reflectance. In this paper, we assume the attributes are colors represented as (R, G, B) tuples in RGB color space. Point clouds can be used in many applications involving the rendering of 3-D objects. For example, the point cloud is a better technical choice than the 360-degree video for virtual reality because it can support 6 degrees of freedom (DoF) rather than 3 DoF [1]. It can also be used in 3-D immersive telepresence due to its capability to reconstruct 3-D objects [2]. For a more thorough review of point cloud applications, refer to [3].

Point clouds can be divided into single-color point clouds and plenoptic point clouds depending on the number of colors per point. Single-color point clouds are simple yet sometimes unrealistic since colors of real-world objects may vary significantly along with the change of viewpoints. Plenoptic point clouds [4] are more complete representations of 3-D objects than single-color point clouds, as they can have multiple colors per spatial point, representing colors of each point as seen from different view angles. They are more realistic but also involve a larger volume of data. For example, for a static plenoptic point cloud with 13 views captured by 8i using the camera plus depth sensors, it has approximately three million points per frame. With each point represented by $12 \times 3 = 36$ and $13 \times 8 \times 3 = 312$ bits for the geometry and colors, respectively, a static plenoptic point cloud is as large as 124 MB (MB) without compression. For a dynamic plenoptic point cloud with 30 frames per second, the bitrate without compression is as high as 3.7 GB (GB) per second. Therefore, there is an urgent need to compress plenoptic point clouds efficiently.

Both the geometry and colors of plenoptic point clouds need to be compressed. Many methods [5]–[11] have been used to compress the geometry of plenoptic point clouds as it is the same as that of single-color point clouds. Among them, the octree and the trisoup have been adopted by the moving pictures experts group (MPEG) geometry-based point cloud compression (G-PCC) [11]. A patch-based projection method [12] has been adopted by the MPEG video-based point cloud compression (V-PCC) [11].

In terms of colors of plenoptic point clouds, there are mainly two groups of methods designed to compress them. The first group is based on a combination of the region-based adaptive hierarchical transform and the Karhunen-Loève transform (RAHT-KLT) [13]. In the RAHT-KLT, the RAHT exploits correlations in color in the spatial dimension while the KLT exploits correlations in color across different viewpoints. For

Manuscript received 4 May 2021; revised 11 November 2021; accepted 5 January 2022. Date of publication 13 January 2022; date of current version 7 June 2023. This work was supported in part by the Natural Science Foundation of China under Grants 62171429 and 62021001, in part by the Natural Science Foundation under Grant 1747751, and in part by USTC Research Funds of the Double First-Class Initiative under Grant YD3490002001. The Associate Editor coordinating the review of this manuscript and approving it for publication was Dr. Sanjeev Mehrotra. *(Corresponding author: Li Li.)*

Li Li and Houqiang Li are with the CAS Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, University of Science and Technology of China, Hefei 230027, China (e-mail: lil1@ustc.edu.cn; lihq@ustc.edu.cn).

Zhu Li is with the Department of Computer Science and Electrical Engineering, University of Missouri-Kansas City Kansas City, MO 64110 USA (e-mail: lizhu@umkc.edu).

Shan Liu is with the Tencent America, Palo Alto, CA 94301 USA (e-mail: shanl@tencent.com).

Digital Object Identifier 10.1109/TMM.2022.3142528

correlations in the spatial dimension, it has been shown in [14] that the RAHT is less efficient in characterizing the correlations than the V-PCC for dense point clouds especially in lossy scenarios. For correlations across different viewpoints, the colors may be similar for some points while they can be quite different for the other points. The wide distribution of colors across viewpoints may make the KLT less efficient for each specified point. In addition, the RAHT-KLT has not taken the temporal correlations of dynamic plenoptic point clouds into consideration.

To overcome the limitations of the RAHT-KLT, the second group of plenoptic point cloud compression methods is based on the V-PCC. In [15], the V-PCC has been extended to support plenoptic point cloud compression. The V-PCC is applied to each view direction independently to exploit correlations in color in the spatial domain. However, as the color correlations across multiple viewpoints are not utilized, its coding performance is quite limited. To the best of the authors' knowledge, no report to date has used a video-based solution to exploit the correlations across different viewpoints to compress colors of plenoptic point clouds.

In addition, there are many unoccupied pixels in the projected videos under the V-PCC framework, which can cost many bits but are of no use to the qualities of reconstructed point clouds. Two groups of methods have been designed to handle those unoccupied pixels in the temporal dimension. One method tried to reduce their bit cost through pixel-based group smoothing [16] due to the high correlations in the temporal dimension. However, this method is unsuitable for the view dimension, in which the correlations are not as high as those in the temporal dimension. The other method introduced occupancy-map-based rate distortion optimization (RDO) [17] and block-level average residual padding [18] to minimize the bit cost of the unoccupied pixels. However, the block-level average residual padding may lead to unsmooth residual blocks, and thus result in serious quality degradations.

Therefore, in this paper, a multiview-video-based framework utilizing the high efficiency of the multiview extension of high efficiency video coding (MV-HEVC) [19] is proposed to compress plenoptic point cloud colors efficiently. In addition, two methods are proposed to reduce the bit cost of unoccupied pixels among various patches. The contributions of this paper are summarized as follows.

1) Using the proposed multiview-video-based framework, we first project a plenoptic point cloud onto its bounding box to generate multiple color videos. Then, we compress these color videos efficiently using the MV-HEVC to utilize the correlations across various viewpoints. To the best of the authors' knowledge, this is the first work that compresses plenoptic point cloud colors with a multiview-video-based framework.
2) We propose a block-based group smoothing instead of pixel-based group smoothing to unify the unoccupied color pixels across the view and temporal directions to reduce the bit cost of some unoccupied color blocks. In this way, the unoccupied blocks have zero residues after prediction, and thus fewer bits are spent on the unoccupied color pixels.
3) We propose a four-neighbor average residual padding instead of the block-level average residual padding to make the residue, which is obtained by subtracting the prediction from the origin, smooth to further reduce the bit consumption of the unoccupied color pixels. The proposed four-neighbor average residual padding is combined with the occupancy-map-based RDO to minimize the bit cost of the unoccupied color pixels.

The rest of this paper is organized as follows. The related work is reviewed in Section II. The proposed multiview-video-based compression framework is described in Section III in detail. The proposed methods to handle the unoccupied color pixels are introduced in Section IV. Section V presents the experimental results and detailed analysis. Section VI concludes the entire paper.

II. Related Work

In this section, the point cloud color compression methods are reviewed, covering the single-color and plenoptic point cloud color compression methods in subsection II-A and subsection II-B, respectively.

A. Single-Color Point Cloud Color Compression

Single-color point cloud color compression methods can be divided into two groups: 3-D-based methods and 2-D-based methods. 3-D-based methods compress colors in 3-D space and can be divided into transform-based methods and prediction-based methods. The transform-based methods utilize the point cloud geometry to create a geometry-aware transform, which is then applied to colors to remove the correlations among them. Zhang *et al.* [20] first proposed using the graph transform to exploit the correlations in the geometry to compress colors. Shao *et al.* [21] further introduced a group of graph transforms with different Laplacian sparsities and selected one for each local block. In addition, Queiroz and Chou [22] proposed the Gaussian Process Transform (GPT) which is essentially the KLT of the Gaussian process to compress colors. Although these methods lead to a good compression performance, a complex eigen-problem needs to be solved to derive the transform, which significantly increases the encoder and decoder complexities. To address this problem, Queiroz and Chou [23] introduced the RAHT to compress colors to obtain a better balance between the complexity and the performance. The RAHT was essentially a wavelet transform [24] weighted by the octree cell occupancy. The RAHT has been adopted by the G-PCC [11] because it shows a good trade-off between the compression efficiency and the complexity. Recently, Chou *et al.* [25] proposed a volumetric approach that generalizes the RAHT to higher orders to compress attributes. In addition, Gu *et al.* [26] proposed a geometry-guided sparse representation to compress colors.

In addition to transform-based methods, prediction is a common way to decorrelate signals. Cohen *et al.* [27] introduced a 3-D intra prediction method using neighboring reconstructed point colors to predict those of the current point. Shao *et al.* [28] proposed decomposing a point cloud into two slices and introduced intra prediction to predict them separately. These methods

divided point cloud colors into multiple blocks and performed intra prediction block by block. In addition, Mammou *et al.* [12] introduced a layer-based structure and used point cloud colors with coarse granularity to predict those with fine granularity. A lifting scheme was further proposed in [29] to improve its performance. This method shows a good performance and supports both lossy and lossless compression, and thus, it has been adopted by the G-PCC [11]. During the standardization process, several reports focus on deriving a better layer-based structure using the kd-tree [30] or the neighboring information [31].

In 2-D-based coding methods, a single-color point cloud is first projected onto single-view images or videos [32] and then compressed using image or video compression standards such as Joint Photographic Experts Group (JPEG) or H.265/HEVC [33]. A variety of methods have been proposed to project a single-color point cloud to single-view videos. Mekuria *et al.* [34] first proposed projecting point colors into a color image through a depth-first tree traversal. The color image was then compressed using JPEG. Budagavi *et al.* [35] proposed sorting point colors directly into single-view videos using an octree or point position in a lossless manner. However, the generated video is unsuitable for the video compression framework due to its limited spatial and temporal correlations. To address this problem, Schwarz *et al.* [36] and He *et al.* [9] proposed projecting single-color point cloud colors to single-view videos using a cube and a cylinder, respectively. The generated videos have high spatial and temporal correlations. However, many points are lost during the projection process due to occlusion. In addition, the generated videos may have large variances, which lead to large compression distortions. Lasserre *et al.* [37] proposed combining the projection and octree together to reduce the number of occluded points. Mammou *et al.* [12] proposed a patch-based method to project a single-color point cloud to a cube in a patch-by-patch manner and organized all the patches into a video. This method won in the MPEG call for proposals for dynamic single-color point cloud compression and became the base of the MPEG V-PCC. In addition, it has been demonstrated as an effective method to compress static single-color point clouds [38].

B. Plenoptic Point Cloud Color Compression

There are not many reports focusing on plenoptic point cloud color compression. Sandri *et al.* [39] [13] first introduced the concept of plenoptic point clouds and provided some variations of the RAHT to compress them. The RAHT is used to exploit the spatial correlations, while the RAHT, the KLT, or the Discrete Cosine Transform (DCT) is used to exploit the color correlations across different viewpoints. As reported by the authors, the RAHT-KLT leads to the best compression performance. Most recently, Krivokuća *et al.* [40] applied prior color clustering and specular/diffuse component separation to derive a better KLT for each specified point. Some performance improvements were observed compared with the original RAHT-KLT. In addition, Zhang *et al.* [41] proposed fitting multiple colors from different viewpoints using a continuous interpolating function and compressed its coefficients using the RAHT or the V-PCC. Furthermore, Naik *et al.* [15] extended the V-PCC to support the plenoptic point cloud compression. They used the same projection process with the V-PCC to generate multiple color videos that are compressed individually using the H.265/HEVC. However, the correlations between multiple videos are not exploited, and thus the coding performance is limited.

III. Proposed Multiview-Video-Based Framework for Plenoptic Point Cloud Compression

Fig. 1 provides a high-level description of the proposed multiview-video-based framework for plenoptic point cloud compression. It can be roughly divided into two steps: multiview video generation and multiview video compression.

A. Multiview Video Generation

A plenoptic point cloud is projected onto its bounding box to generate multiple single-view videos using the same approach employed in the V-PCC [11]. To guarantee the completeness of the proposed algorithm, the video generation process of the V-PCC is briefly introduced as follows. The projection from a point cloud to 2-D geometry and color videos in the V-PCC can be roughly divided into three steps: patch generation, patch packing, and patch padding.

1) Patch Generation: A point cloud is first divided into several patches by projecting it onto its 3-D bounding box. Generally, each patch is generated by clustering the neighboring points with similar normals together. Compared with the global method in [36], the patch-based method has the following two benefits. First, more points are projected so that the reconstructed point cloud has a better quality. Second, the points with similar normals are clustered together so that a geometry video with fewer variances is generated. Note that each 3-D patch is projected onto two frames to handle the case of multiple points being projected to the same pixel. The values of the co-located pixels in these two frames have small differences when one point obstructs the other point. Most co-located pixels have the same values, leading to strong temporal correlations between the two video frames.

2) Patch Packing: A simple packing strategy is used to organize the patches into frames. The patch location is determined through an exhaustive search in a raster scan order. The first position that can guarantee an overlapping-free insertion of the patch is selected, and all the grid cells covered by the patch are marked as used. To utilize the temporal correlations among neighboring frames, the patches are packed in a temporally consistent manner. In addition, patch rotation is supported to allow more flexible packing.

3) Patch Padding: A substantial amount of empty space exists among various patches after patch packing. The patch padding aims to fill the empty space to make the generated frames more suitable for video compression. A variety of methods have been proposed for the padding of geometry and color videos. For geometry videos, the padding is performed block by block in a raster scan order using the neighboring occupied pixels. For color videos, the push-pull algorithm [42] and the harmonic background filling method [43] have been proposed to improve its spatial continuity. The unoccupied pixels in the first

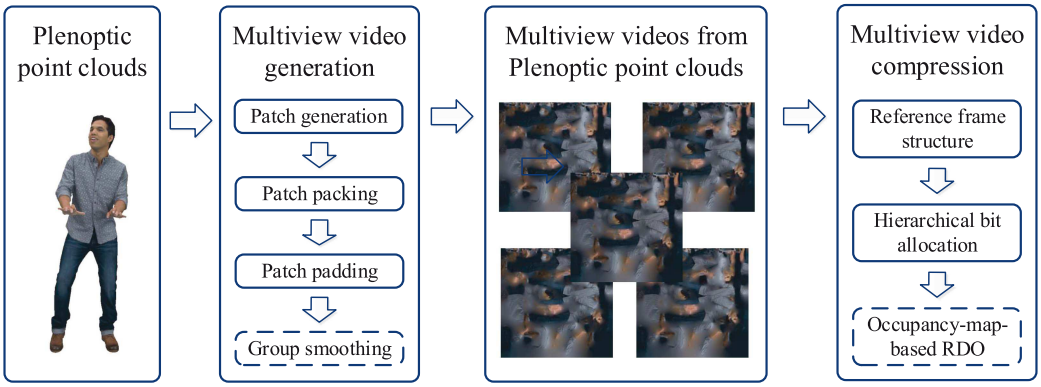


Fig. 1. High-level description of the proposed multiview-video-based framework for plenoptic point cloud compression. The solid and dotted squares indicate mandatory and optional modules in the proposed framework.

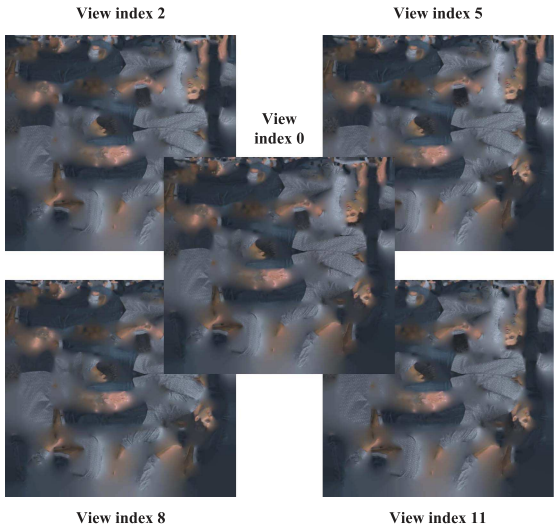


Fig. 2. Examples of the projected views from the plenoptic point cloud "Loot". These views are from the view index 0, 2, 5, 8, and 11. They have strong correlations to be exploited. Their spatial resolutions are 5120×4800.

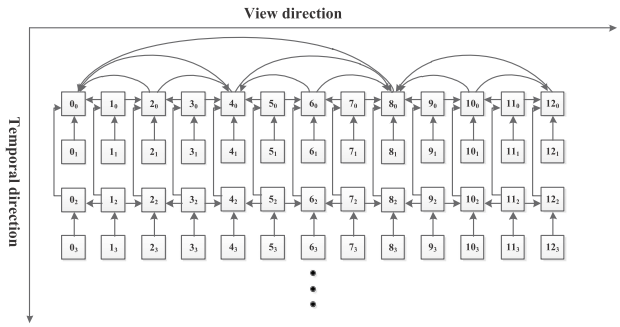


Fig. 3. Proposed multiview-video-based framework using 13 views as an example. The squares with indices 0 to 12 in the horizontal direction represent the 13 views. Subscripts 0 to 3 in the vertical direction indicate the frame index in the temporal direction. The arrows indicate the reference relationships between different views and frames.

and second video frames projected from the same point cloud frame undergo a further group smoothing scheme to increase their temporal correlations [16].

As mentioned in Section I, the main difference between single-color and plenoptic point clouds is the number of colors per point. Therefore, the generated geometry video from a plenoptic point cloud is the same as that from a single-color point cloud. However, in contrast to the single-color point cloud compression that generates only one color video, the plenoptic point cloud compression generates many color videos. The number of color videos is determined by the number of cameras from various directions. Fig. 2 shows one typical example of the color frames generated from multiple viewpoints. It can be seen that these frames are similar to each other despite some pixel differences. The view correlations need to be fully utilized to improve the compression efficiency of plenoptic point clouds. In addition to the view correlations, each point cloud frame in each view is projected onto two video frames to handle the case of multiple points being projected to the same pixel. These two video frames have strong temporal correlations to be exploited.

B. Multiview Video Compression

To fully utilize the temporal and view correlations, we propose using MV-HEVC to compress these color videos efficiently. The reference frame structure and bit allocation in MV-HEVC are carefully designed to adapt the color videos generated from a plenoptic point cloud.

1) Reference Frame Structure: We propose to use hierarchical [44] and low delay coding structures in the view and temporal directions, respectively. Using a plenoptic point cloud with 13 views as an example, Fig. 3 shows the proposed reference frame structure. In Fig. 3, the squares with indices 0 to 12 in the horizontal direction represent the 13 views. Subscripts 0 to 3 in the vertical direction indicate the frame index in the temporal direction. Note that each point cloud frame is projected onto two video frames. Therefore, the video frames 0 and 1 are projected from the same point cloud frame. The arrows between different views and frames indicate the reference relationships.

(a) Frame difference (b) View difference

Fig. 4. Comparison between the view and frame differences for the plenoptic point cloud "Loot". Their spatial resolutions are 5120×4800. The frame difference is calculated using frame 1 minus frame 0. The view difference is calculated using view 1 minus view 0.

TABLE I
QP SETTINGS OF VARIOUS VIEWS AND FRAMES

Hierarchical level	Frame 0	Frame 1
0	QP_I+1	QP_I+4
1	QP_I+2	QP_I+5
2	QP_I+3	QP_I+6
3	QP_I+4	QP_I+7

In the view direction, since we have 13 views that are more than 8 but less than 16, we set the Group of Pictures (GoP) size of the hierarchical coding structure to 8 instead of 16. Here the GoP indicates the period of the reference frame structure. The even and odd views are coded as reference and non-reference frames, respectively. All the views are coded with B frames using uni-directional and bi-directional prediction to fully utilize the correlations between different views.

In the temporal direction, the GoP size of the low delay coding structure is set to 2. For frame 1, only frame 0 with the same view index is used as its reference frame since the temporal correlations are much higher than the view correlations as indicated by Fig. 4. It can be seen that the view difference is much larger than the frame difference. For frame 2, frame 0 with the same view index and the other views in the current frame are used as its references.

2) Bit Allocation: In addition to the reference frame structure, the bit allocation between different views and frames is important to the compression performance. In the hierarchical coding structure with GoP size 8, all the views are divided into four hierarchical levels. The more the view is referenced, the lower the hierarchical level is, and vice versa. The view with index 8 is assigned level 0. The view with index 4 is assigned level 1. The views with indices 2 and 6 are assigned level 2. The views with indices 1, 3, 5, and 7 are assigned level 3. The quantization parameters (QPs) of various views and frames are set based on their hierarchical levels and frame indices according to the QP of the intra frame QP_I as follows. First, the higher the hierarchical level is, the larger the QP. In the view direction, we set the QP of each hierarchical level as $QP_I + level + 1$. Second, the second non-reference frame uses a higher QP than the first reference frame from the same point cloud frame. In the temporal direction, we set the QP of QP_1 as $QP_0 + 3$. The detailed settings of the QPs of all the frames and views are shown in Table I.

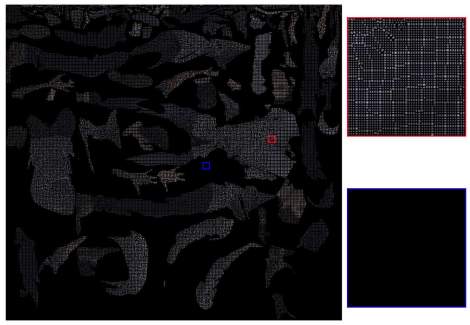


Fig. 5. Typical example of a projected view with the unoccupied pixels set as black from the plenoptic point cloud "Loot". Its spatial resolution is 5120×4800. The enlarged red square shows some examples of the isolated unoccupied pixels. The enlarged blue square shows some examples of the continuous unoccupied pixels.

IV. EFFICIENT UNOCCUPIED PIXEL COMPRESSION

After patch packing as discussed in Section III-A3, there are empty spaces in generated video frames to be padded, which contain unoccupied pixels. The unoccupied pixels are of no use to qualities of reconstructed plenoptic point clouds, and therefore, their bit cost needs to be minimized. Since the number of unoccupied pixels in multiple color videos is much more than that in one color video, this problem becomes much more serious in the multiview-video-based framework for plenoptic point cloud compression. In this section, we design two methods to minimize the bit cost of the unoccupied pixels: block-based group smoothing and combined occupancy-map-based RDO and four-neighbor average residual padding.

A. Block-Based Group Smoothing

In the V-PCC reference software, several padding methods [42], [43] have been integrated to minimize the bit cost of unoccupied pixels in the first frame. Then, a pixel-based group smoothing is used to smooth all the unoccupied pixels using the average values of the first frame and the second frame to minimize the bit cost of the unoccupied pixels in the second frame. This method exploits the temporal correlations of the unoccupied pixels in the first and second frames to improve compression efficiency. However, the difference between various views in the view direction is much larger than that in the temporal direction as shown in Fig. 4. The frame difference is difficult to recognize while the view difference is large. The large difference may render this method unworkable in the view direction.

Fig. 5 shows a typical example of a projected view with the unoccupied pixels set as black. It can be seen that the unoccupied pixels can be divided into two groups: the continuous unoccupied pixels, as indicated by the enlarged blue square, and the isolated unoccupied pixels, as indicated by the enlarged red square. The continuous unoccupied pixels can be smoothed using the average value of all the unoccupied co-located pixels across all the view directions. However, smoothing of the isolated unoccupied pixels may destroy the spatial continuity of a block containing both occupied and unoccupied pixels, as shown in Fig. 6. It can be seen that the difference between the original block and the prediction block is larger but smoother without

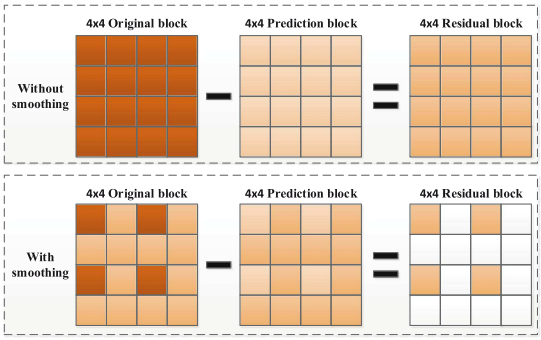


Fig. 6. Influence of smoothing of isolated unoccupied pixels. The residual block becomes unsmooth after smoothing.

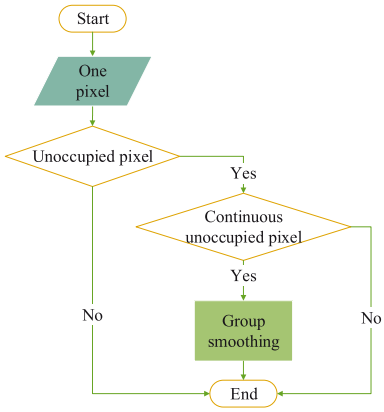


Fig. 7. Flowchart of the proposed block-based group smoothing scheme.

smoothing. After smoothing, the current block and its prediction block are the same for the unoccupied pixels but different for the occupied ones. In this way, the residual block has some singular values, which leads to inefficiency for the following transform process.

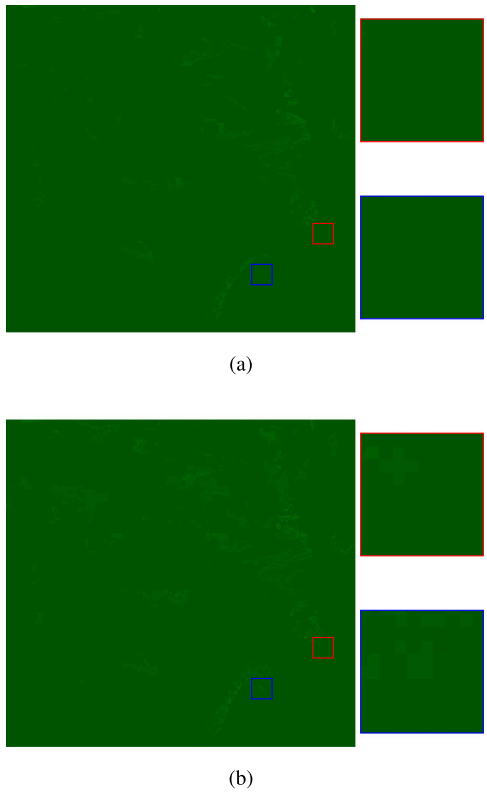
In this paper, a block-based group smoothing scheme is proposed to address this problem as shown in Fig. 7. Using the proposed block-based group smoothing scheme, an unoccupied pixel goes through the following two steps: continuous unoccupied pixel detection and smoothing.

1) Continuous Unoccupied Pixel Detection: For each unoccupied pixel, a $K \times K$ block with it as the center pixel is first found. Only when all the pixels in the $K \times K$ block are unoccupied, is the center pixel smoothed across the view direction. In this way, the isolated unoccupied pixels are not smoothed, and thus the spatial continuity of a block containing both occupied and unoccupied pixels is kept. The continuous unoccupied pixels are smoothed to reduce their bit cost to improve the compression efficiency. Note that the proposed algorithm degenerates from block-based group smoothing to pixel-based group smoothing [16] when K is equal to 1. The influences of different Ks on the performances are discussed in the experimental results.

2) Continuous Unoccupied Pixel Smoothing: The continuous unoccupied pixels are smoothed using the average values of the co-located continuous unoccupied pixels of all the views in both frame 0 and frame 1,

$$g_{i,j} = \sum_{k=0}^{N-1} (f_{0,k} + f_{1,k})/(2\,N), i \in 0, 1, j \in [0, N-1], \quad (1)$$

where N is the number of views of a plenoptic point cloud. i is the frame index. j is the view index. $f_{i,j}$ and $g_{i,j}$ are the values of continuous unoccupied pixels in the jth view of the ith frame before and after group smoothing. After the above smoothing scheme, in both the view and temporal directions, good predictions can be obtained for the continuous unoccupied pixels. A comparison of the residual frames of a specific view with and without the block-based group smoothing is shown in Fig. 8. It can be seen from the enlarged red and blue squares that the proposed block-based group smoothing scheme leads to smoother residues, and thus can significantly reduce the bit cost.



(a)

(b)

Fig. 8. Comparison of the residual frames with and without the block-based group smoothing. These frames are from the $8th$ view of the plenoptic point cloud "Loot" under the bitrate scenario r1 defined in the V-PCC common test condition [45]. (a) With block-based group smoothing. The variances of the red and blue squares are 0.12 and 0.31, respectively. (b) Without block-based group smoothing. The variances of the red and blue squares are 4.69 and 3.63 accordingly.

B. Combined Occupancy-Map-Based Rate Distortion Optimization and Four-Neighbor Average Residual Padding

The block-based group smoothing only deals with continuous unoccupied pixels. In this subsection, both continuous and isolated unoccupied pixels are handled by introducing a combined

occupancy-map-based RDO and four-neighbor average residual padding scheme.

In the default encoder of the MV-HEVC reference software, encoding parameters P are determined by the rate distortion (R-D) cost J:

$$\min_{P} J = \sum_{i=1}^{N} D_i(P) + \lambda R(P), \tag{2}$$

where $D_i(P)$ is the distortion of pixel i in the current block. $R(P)$ is the bit cost of the current block. N is the number of pixels in the current block. λ is the Lagrangian multiplier. In different stages of RDO processes, the distortion can be the sum of the absolute difference (SAD), the sum of the absolute transformed difference (SATD), or the sum of the squared difference (SSD).

As can be seen from (2), for a block containing both occupied and unoccupied pixels, the distortions of the occupied and unoccupied pixels are accumulated together with the same weights. This indicates that the default optimization target treats the distortions of the occupied and unoccupied pixels equally. However, different from the occupied pixels, the distortions of the unoccupied pixels have no influences on the reconstructed qualities of plenoptic point clouds. Therefore, the RDO scheme in the MV-HEVC reference software is unsuitable for the proposed multiview-video-based plenoptic point cloud compression framework.

In this paper, an occupancy-map-based mask is added to the RDO scheme to address this problem [46] [17]. The R-D cost of a block after adding the mask is calculated as

$$\min_{P} J = \sum_{i=1}^{N} D_i(P) \times M_i + \lambda R(P), \tag{3}$$

where M_i is 1 when pixel i is an occupied pixel, and M_i is 0 when pixel i is an unoccupied pixel. Using (3), only distortions of occupied pixels are considered when calculating the R-D cost of the current block. The proposed occupancy-map-based RDO is applied to intra prediction, inter prediction, and sample adaptive offset (SAO) in the MV-HEVC reference software.

In addition, even if distortions of unoccupied pixels are ignored purposely, unoccupied pixels, especially isolated ones, may still cost many bits. As it may be difficult to obtain good predictions for isolated unoccupied pixels, the bit cost of the residual block including both occupied and unoccupied pixels is still high. The residual block here is obtained by subtracting the prediction block from the original block. In our previous work [18], a block-based average residual padding scheme is proposed to pad the unoccupied pixels in a residual block. However, it may make residual blocks unsmooth especially for those with isolated unoccupied pixels. In this paper, we propose to iteratively pad the unoccupied pixels of a residual block using the average of the four-neighbor occupied or padded pixels, as shown in Fig. 9. The blocks with orange and the other colors represent the occupied and unoccupied pixels, respectively. The blue blocks are padded first according to the four-neighbor average of the orange blocks in the first iteration. The green blocks are then padded according to the four-neighbor average of the blue blocks in the second iteration. Using the above residual padding scheme, the residual block becomes smoother as the unoccupied pixels are padded with the average value of its four-neighbors and is efficient for the following transform process.

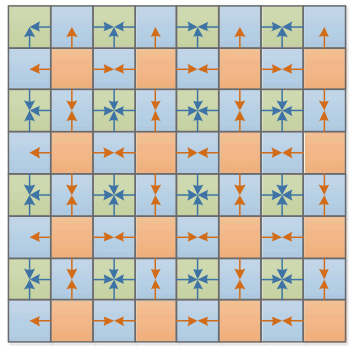


Fig. 9. Examples of padding unoccupied pixels of an 8×8 residual block. The orange blocks represent occupied pixels. The other blocks represent unoccupied pixels. The arrows indicate the padding process of the current unoccupied pixels.

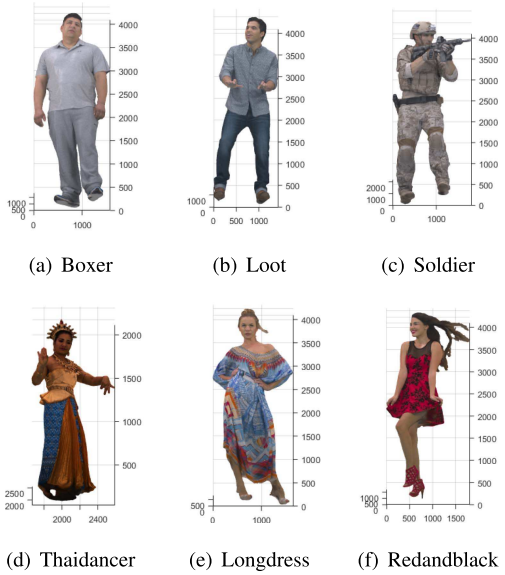
TABLE II
CHARACTERISTICS OF THE PLENOPTIC POINT CLOUDS

Name	Points	Cameras	Geometry/Color bit depth
Boxer	3496011	13	12/8
Loot	3021497	13	12/8
Soldier	4007891	13	12/8
Thaidancer	3130215	13	12/8
Longdress	3100469	12	12/8
Redandblack	2776067	12	12/8

V. EXPERIMENTAL RESULTS

The proposed algorithms are implemented in the V-PCC reference software TMC2-7.0 [47] and MV-HEVC reference software HTM-16.3 [48] to compare with the RAHT-KLT [13] and the V-PCC independently applied to each view direction [15]. All the static and dynamic plenoptic point clouds defined in [4] are tested to verify the effectiveness of the proposed algorithms. The characteristics of the tested static plenoptic point clouds are shown in Table II. Note that the static plenoptic point clouds are voxelized into 11 bits to guarantee a fair comparison with the RAHT-KLT. The tested dynamic point cloud is "Thaidancer" with 30 frames per second (fps). We test 32 frames to show the benefits of the proposed multiview-video-based framework. Some examples of the rendered views of the tested plenoptic point clouds are shown in Fig. 10.

For the static plenoptic point clouds, the all intra configuration defined in the V-PCC common test condition [45] is tested. For the dynamic plenoptic point clouds, the low delay configuration is tested, as shown in Fig. 3. The QPs of the I frames in the proposed multiview-video-based framework are set the same as the color QPs from the low bitrate (r1) [45] to the high bitrate (r5) [45] scenarios to verify the performance of the proposed algorithm in a large bitrate range. The detailed configurations of the MV-HEVC reference software are shown in Table III. Note that lossless geometry is used when comparing with the



(a) Boxer (b) Loot (c) Soldier

(d) Thaidancer (e) Longdress (f) Redandblack

Fig. 10. Examples of rendered views of the tested plenoptic point clouds.

TABLE III
CONFIGURATIONS OF THE MV-HEVC CODING STRUCTURE

Notation	Value
Number of layers	17
View encoding order	0, 8, 4, 2, 1, 3, 6, 5, 7, 12, 10, 9, 11
Frame encoding order	0, 1
Interview references	3
Temporal references	1
Intra QP	42/37/32/27/22 (r1-r5)

RAHT-KLT for a fair comparison while lossy geometry is used when comparing with the V-PCC [15].

The total color bits for the Y, C_B, and C_R components across all viewpoints are used as the bit cost. For the quality metric, we first calculate the Peak-Signal-to-Noise-Ratio (PSNR) of the Y, C_B, and C_R components using the MPEG "pc_error" software [49] to measure the quality of each viewpoint. Then the PSNRs of all the viewpoints are averaged to obtain the quality of a plenoptic point cloud. In addition to the PSNR for each component individually, the following formula is used to calculate the PSNR of YC_BC_R [50],

$$PSNR_{YC_BC_R} = \frac{6 \cdot PSNR_Y + PSNR_{C_B} + PSNR_{C_R}}{8}, \tag{4}$$

where $PSNR_Y$, $PSNR_{C_B}$, $PSNR_{C_R}$ are the PSNRs for the Y, C_B, and C_R components, respectively. Since various algorithms may generate different bitrates, the Bjontegaard Delta bitrate (BD-rate) [51] is employed for a fair comparison.

We first show the performance of the proposed multiview-video-based framework without the efficient unoccupied pixel compression algorithms compared with the RAHT-KLT and the V-PCC independently applied to each view direction. Then, the performances of the efficient unoccupied pixel compression algorithms are shown step by step. After that, some figures regarding the qualities of various views to show the quality variances across various views are provided. Finally, some representative R-D curves and subjective examples are shown to better illustrate the benefits of the proposed framework.

TABLE IV
BD-RATE REDUCTIONS OF THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK WITHOUT THE EFFICIENT UNOCCUPIED PIXEL COMPRESSION ALGORITHMS COMPARED WITH THE RAHT-KLT [13]

Name	RAHT-KLT		Multiview-video		Y
	Color bits	Y-PSNR	Color bits	Y-PSNR	BD-rate
Boxer	138869	33.27	167216	34.38	–31.9%
	534974	36.58	312592	36.18	
	1102667	38.51	609392	38.02	
	1740290	39.85	1269464	40.09	
	2506516	41.02	2729384	42.27	
Loot	122169	32.84	145448	33.64	–35.7%
	505156	36.47	305688	36.15	
	1036214	38.57	622192	38.69	
	1604170	39.97	1245248	41.07	
	2252251	41.16	2487784	43.28	
Soldier	279712	30.03	251280	30.90	–37.1%
	1193244	34.15	546656	33.32	
	2361547	36.60	1138136	35.79	
	3514995	38.24	2345992	38.15	
	4777745	39.63	4921416	40.55	
Thai	434126	28.46	277056	29.01	–34.5%
	1719585	33.63	577528	31.70	
	3058823	36.63	1133280	34.21	
	4292715	38.52	2302576	36.51	
	5599587	40.03	5243640	39.39	
Long	519371	28.01	280816	28.87	–52.1%
	2081546	33.01	539944	31.31	
	3770193	36.19	1027848	33.73	
	5245716	38.36	2043576	36.09	
	6701977	40.16	4536824	38.81	
Red	224020	31.82	167128	32.65	–46.4%
	903125	35.90	308184	34.62	
	1736193	38.43	570992	36.85	
	2510406	40.15	1104824	39.04	
	3313844	41.59	2217960	41.34	
Average	–	–	–	–	**–39.7%**

A. *Performance of the Proposed Multiview-Video-Based Framework*

Table IV shows the BD-rate reductions of the proposed multiview-video-based framework without the efficient unoccupied pixel compression algorithms compared with the RAHT-KLT. It can be seen that the proposed multiview-video-based framework leads to a BD-rate reduction of 39.7% compared with the RAHT-KLT on average. For the plenoptic point cloud "Longdress," the proposed framework achieves a BD-rate reduction as high as 52.1%. The experimental results demonstrate that the proposed multiview-video-based framework can compress the plenoptic point clouds more efficiently than the RAHT-KLT. The performance improvements are consistent for all tested plenoptic point clouds. Note that we only compare the R-D performance of the Y component since only the Y-PSNR is shown in [13]. The color bits are the total bits for the Y, C_B, and C_R components.

The benefits of the proposed framework mainly come from the following two aspects. First, the V-PCC outperforms the RAHT in terms of exploiting the spatial correlations. Second, as we mentioned in Section I, the wide distribution of colors across viewpoints may make the KLT less efficient for each specified point. However, various local blocks can choose different prediction modes depending on the contents. This is why the proposed framework can better exploit the view correlations to achieve a better R-D performance. In addition, the computational complexity of the RAHT-KLT can be quite high especially at the decoder since the KLT needs to be trained online.

TABLE V
BD-RATE REDUCTIONS OF THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK WITH THE EFFICIENT UNOCCUPIED PIXEL COMPRESSION ALGORITHMS COMPARED WITH THE V-PCC WITHOUT USING THE CORRELATIONS AMONG VARIOUS VIEWS [15]

Name	Y	C_B	C_R	YC_BC_R
Boxer	–65.3%	–57.7%	–58.2%	–63.7%
Loot	–69.3%	–67.6%	–67.7%	–68.9%
Soldier	–75.3%	–73.6%	–73.9%	–74.9%
Thaidancer	–82.7%	–82.9%	–82.6%	–82.7%
Longdress	–85.8%	–85.7%	–85.7%	–85.7%
Redandblack	–78.1%	–78.4%	–79.2%	–78.2%
Thaidancer (Dynamic)	–78.3%	–79.3%	–79.3%	–78.5%
Average (Static)	**–76.1%**	**–74.3%**	**–74.5%**	**–75.7%**
Enc.time (self)	100%			
Enc.time (child)	157%			
Dec.time (self)	100%			
Dec.time (child)	135%			

Table V shows the BD-rate reductions of the proposed multiview-video-based framework without the efficient unoccupied pixel compression algorithms compared with the V-PCC independently applied to each view direction [15]. It can be seen that the proposed algorithm leads to BD-rate reductions of 76.1%, 74.3%, and 74.5% for the static point clouds compared with the V-PCC for the Y, C_B, and C_R components, respectively. For the dynamic point cloud "Thaidancer," the proposed algorithm achieves BD-rate reductions of 78.3%, 79.3%, and 79.3% accordingly. The performance improvements for all tested plenoptic point clouds are significant and consistent. The experimental results show that the MV-HEVC can better exploit the correlations across different viewpoints compared with HEVC to significantly improve the R-D performance.

In Table V, the Enc.time (self), Enc. time (child), Dec. time (self), and Dec. time (child) denote the V-PCC encoding time ratio, MV-HEVC encoding time ratio, V-PCC decoding time ratio, and MV-HEVC decoding time ratio of the proposed algorithm compared with the reference algorithm, respectively. The ratios are the average values of all tested static plenoptic point clouds. It can be seen that, for the V-PCC reference software, the proposed multiview-video-based framework without the efficient unoccupied pixel compression algorithms leads to the same encoding and decoding complexities compared with the V-PCC independently applied to each view direction since the changes to the V-PCC reference software are the same with respect to the two algorithms. However, the proposed multiview-video-based framework without the efficient unoccupied pixel compression algorithms costs 157% and 135% of the encoding and decoding time for the MV-HEVC reference software, respectively, as the inter-view prediction involves more complex motion estimation and motion compensation processes.

B. Performance of the Block-Based Group Smoothing

Table VI shows BD-rate reductions of the proposed multiview-video-based framework with the block-based group smoothing compared with that without it. The block size is set to 4 in the experimental results shown in Table VI. It can be seen that the proposed block-based group smoothing can lead

TABLE VI
BD-RATE REDUCTIONS OF THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK WITH THE BLOCK-BASED GROUP SMOOTHING COMPARED WITH THE FRAMEWORK WITHOUT IT

Name	Y	C_B	C_R	YC_BC_R
Boxer	–15.1%	–14.1%	–14.1%	–14.9%
Loot	–15.2%	–13.8%	–13.9%	–14.9%
Soldier	–9.3%	–8.2%	–8.1%	–9.0%
Thaidancer	–10.8%	–10.4%	–10.7%	–10.8%
Longdress	–9.1%	–9.0%	–9.2%	–9.1%
Redandblack	–13.5%	–14.4%	–14.5%	–13.7%
Thaidancer (Dynamic)	–12.3%	–12.4%	–12.2%	–12.3%
Average (Static)	**–12.2%**	**–11.6%**	**–11.8%**	**–12.1%**
Enc.time (self)	104%			
Enc.time (child)	86%			
Dec.time (self)	100%			
Dec.time (child)	95%			

TABLE VII
BD-RATE REDUCTIONS OF THE PROPOSED BLOCK-BASED GROUP SMOOTHING WITH DIFFERENT BLOCK SIZES

Name	Y BD-rate				
	K = 1	K = 2	K = 4	K = 8	K = 16
Boxer	160.0%	–14.2%	–15.1%	–14.9%	–14.0%
Loot	112.1%	–14.4%	–15.2%	–14.7%	–13.2%
Soldier	95.7%	–8.5%	–9.3%	–8.6%	–7.1%
Thaidancer	–10.7%	–10.9%	–10.8%	–10.2%	–9.6%
Longdress	0.9%	–9.4%	–9.1%	–8.7%	–8.1%
Redandblack	114.2%	–12.0%	–13.5%	–13.2%	–12.5%
Average	**78.7%**	**–11.6%**	**–12.2%**	**–11.7%**	**–10.7%**

to an average of 12.2%, 11.6%, and 11.8% performance improvements for the Y, C_B, and C_R components, respectively. The experimental results show that the proposed algorithm effectively reduces bit cost of the unoccupied pixels, and thus leads to significant BD-rate savings.

In terms of the complexity, the block-based group smoothing leads to a 4% encoder complexity increase for the V-PCC reference software due to the smoothing operations. For the MV-HEVC reference software, the block-based group smoothing decreases the encoding time by 14% as some unoccupied pixels choose skip mode in advance. In addition, the block-based group smoothing decreases the decoding complexity by 5% because large blocks are chosen for motion compensation.

To show the influences of different K values on the performances of the block-based group smoothing, the BD-rate reductions using various K values are shown for different static plenoptic point clouds in Table VII. It can be seen that the block size of 4 achieves the best performance among all the values on average. Although the block size of 2 is the best choice for the plenoptic point clouds "Thaidancer" and "Longdress," the performance differences between the block sizes of 4 and 2 for these two point clouds are small. Therefore, K can be set to 4 in practical applications. In addition, the block size 4 is the same as the minimum block size for the MV-HEVC intra prediction. Therefore, a 4×4 block including only unoccupied pixels can be an independent prediction unit or transform unit in the MV-HEVC encoder. In this way, the negative effects of the distortions of the unoccupied pixels on occupied pixels are minimized.

The BD-rate reductions of the block-based group smoothing are essentially determined by a trade-off between the number of

TABLE VIII
BD-RATE REDUCTIONS OF THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK WITH THE COMBINED OCCUPANCY-MAP-BASED RDO AND FOUR-NEIGHBOR AVERAGE RESIDUAL PADDING COMPARED WITH THE FRAMEWORK WITHOUT IT

Name	Y	C_B	C_R	YC_BC_R
Boxer	–24.1%	–10.3%	–8.8%	–21.3%
Loot	–25.0%	–16.0%	–15.0%	–23.1%
Soldier	–23.9%	–3.1%	–3.6%	–19.8%
Thaidancer	–16.3%	–14.7%	–14.2%	–15.9%
Longdress	–33.6%	–17.0%	–17.4%	–29.8%
Redandblack	–33.2%	–29.8%	–38.2%	–33.5%
Thaidancer (Dynamic)	–20.4%	–17.6%	–16.5%	–19.6%
Average (Static)	**–26.0%**	**–15.1%**	**–16.2%**	**–23.9%**
Enc.time (self)	101%			
Enc.time (child)	115%			
Dec.time (self)	100%			
Dec.time (child)	95%			

averaged continuous and isolated unoccupied pixels. The more the averaged continuous unoccupied pixels, the better the performance. The less the averaged isolated unoccupied pixels, the better the performance. When K is equal to 1, all continuous and isolated unoccupied pixels are averaged. The isolated unoccupied pixels destroy the spatial continuity of many blocks, and thus degrade the average R-D performance significantly. Along with the increase of K from 1 to 4, the number of averaged isolated and continuous unoccupied pixels decreases. However, the reduction of the isolated points is the major influence, and thus, the R-D performance is improved gradually. If K further increases, the number of averaged isolated unoccupied pixels remains approximately unchanged, while the number of averaged continuous unoccupied pixels continues to decrease. Therefore, the R-D performance becomes slightly worse.

C. Performance of the Combined Occupancy-Map-Based RDO and Four-Neighbor Average Residual Padding

Table VIII shows BD-rate reductions of the proposed multiview-video-based framework with the combined occupancy-map-based RDO and four-neighbor average residual padding scheme compared with the framework without it. It can be seen that the combined occupancy-map-based RDO and four-neighbor average residual padding scheme achieves BD-rate reductions of 26.0%, 15.1%, and 16.2% for the Y, C_B, and C_R components, respectively. The combined occupancy-map-based RDO and four-neighbor average residual padding scheme leads to significant BD-rate reductions as we have not taken the distortions of the unoccupied pixels into consideration and smoothed the residual blocks. It leads to a better performance improvement compared with the block-based group smoothing as it considers the bit cost of both the continuous and isolated unoccupied pixels. Comparing Tables VIII and VI, it can be seen that the performance difference between the block-based group smoothing and the combined occupancy-map-based RDO and four-neighbor average residual padding scheme for the plenoptic point cloud "Thaidancer" is the smallest since the number of its isolated unoccupied pixels is the smallest. In terms of the complexity of the MV-HEVC reference software, the combined occupancy-map-based RDO and four-neighbor average residual padding scheme results in a 15% encoding time increase due to the padding of the residual blocks, while saving approximately 5% of decoding time due to the large block size for motion compensation.

TABLE IX
BD-RATE REDUCTIONS OF THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK WITH THE OCCUPANCY-MAP-BASED RDO COMPARED WITH THE FRAMEWORK WITHOUT IT

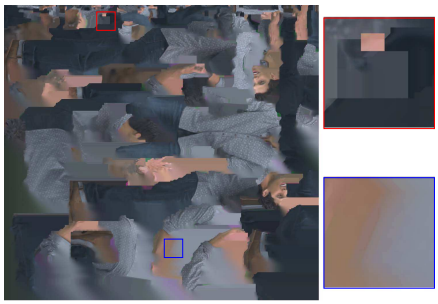
Name	Y	C_B	C_R	YC_BC_R
Boxer	–21.4%	–12.8%	–12.7%	–19.5%
Loot	–19.3%	–15.1%	–14.7%	–18.2%
Soldier	–16.2%	–5.8%	–7.5%	–13.9%
Thaidancer	–13.5%	–12.5%	–12.4%	–13.2%
Longdress	–19.9%	–11.8%	–12.2%	–18.0%
Redandblack	–22.5%	–23.1%	–26.2%	–23.0%
Thaidancer (Dynamic)	–17.1%	–15.5%	–14.4%	–16.6%
Average (Static)	**–18.8%**	**–13.5%**	**–14.3%**	**–17.6%**
Enc.time (self)	101%			
Enc.time (child)	102%			
Dec.time (self)	100%			
Dec.time (child)	95%			

TABLE X
BD-RATE REDUCTIONS OF THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK WITH THE COMBINED OCCUPANCY-MAP-BASED RDO AND BLOCK-BASED AVERAGE RESIDUAL PADDING SCHEME COMPARED WITH THE FRAMEWORK WITHOUT IT

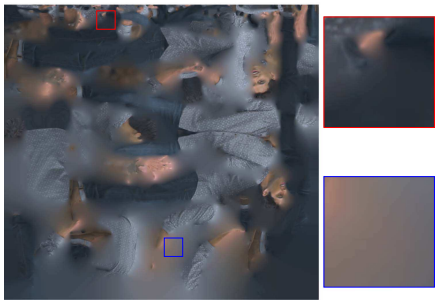
Name	Y	C_B	C_R	YC_BC_R
Boxer	–5.1%	–27.5%	–30.5%	–9.5%
Loot	–5.4%	–35.7%	–33.9%	–10.9%
Soldier	–4.7%	–25.2%	–23.3%	–6.9%
Thaidancer	–16.4%	–14.6%	–14.1%	–15.9%
Longdress	–5.1%	22.2%	26.9%	2.4%
Redandblack	–13.7%	–9.7%	3.9%	–10.5%
Thaidancer (Dynamic)	–21.0%	–18.5%	–17.6%	–20.2%
Average (Static)	**–8.4%**	**–15.1%**	**–11.8%**	**–8.6%**
Enc.time (self)	101%			
Enc.time (child)	107%			
Dec.time (self)	101%			
Dec.time (child)	95%			

Table IX shows BD-rate reductions of the proposed multiview-video-based framework with the occupancy-map-based RDO [17] compared with the framework without it. Note that the residual padding is disabled when showing the performance of occupancy-map-based RDO in Table IX. Comparing Tables IX and VIII, it can be seen that the residual padding yields an extra 7.2% BD-rate reduction for the Y component. The experimental results demonstrate that the proposed four-neighbor average residual padding algorithm can further reduce the bit cost of the unoccupied pixels.

Table X shows BD-rate reductions of the proposed multiview-video-based framework with the combined occupancy-map-based RDO and block-based average residual padding scheme [18] compared with the framework without it. The combined occupancy-map-based RDO and block-based average residual padding scheme leads to an average of 8.4% BD-rate reduction for the Y component, which is much less than 26.0% achieved by the combined occupancy-map-based RDO and four-neighbor average residual padding scheme. The block-based average residual padding algorithm may make the residual blocks with isolated unoccupied pixels unsmooth, and thus result in much less performance improvements compared with the proposed algorithm in this paper.



(a) With combined occupancy-map-based RDO and four-neighbor average residual padding



(b) Without combined occupancy-map-based RDO and four-neighbor average residual padding

Fig. 11. Comparison between the reconstructed multiview-video frame of the plenoptic point cloud in the $YC_B\ C_R$ color space with and without the combined occupancy-map-based RDO and four-neighbor average residual padding scheme. These frames are from the $8th$ view of the plenoptic point cloud "Loot" under the bitrate scenario r3 defined in the V-PCC common test condition [45]. The bit cost with and without the combined occupancy-map-based RDO and four-neighbor average residual padding scheme is 1241488 and 1515400, respectively.

To better show why the combined occupancy-map-based RDO and four-neighbor average residual padding scheme can achieve significant BD-rate reductions, we show the comparison between the reconstructed multiview-video frame of the plenoptic point cloud in the $YC_B\ C_R$ color space with and without the combined occupancy-map-based RDO and four-neighbor average residual padding scheme in Fig. 11. It can be seen that the unoccupied pixels are coded with much worse qualities when the combined occupancy-map-based RDO and residual padding scheme is used as much fewer bits are assigned to them. The combined occupancy-map-based RDO and four-neighbor average residual padding scheme devotes bits to the reconstruction of occupied pixels and avoids wasting bits on coding areas which contain unoccupied pixels.

D. Performance of the Combination of the Block-Based Group Smoothing and the Combined Occupancy-Map-Based RDO and Four-Neighbor Average Residual Padding

Table XI shows BD-rate reductions of the proposed multiview-video-based framework with the proposed block-based group smoothing and the combined occupancy-map-based RDO and four-neighbor average residual padding compared with the framework without them. It can be seen that the proposed combination achieves a BD-rate reduction of 26.1% for the Y component on average. Comparing Tables XI and VIII,

TABLE XI
BD-RATE REDUCTIONS OF THE PROPOSED MULTIVIEW-VIDEO FRAMEWORK COMBINING THE PROPOSED BLOCK-BASED GROUP SMOOTHING AND THE COMBINED OCCUPANCY-MAP-BASED RDO AND FOUR-NEIGHBOR AVERAGE RESIDUAL PADDING COMPARED WITH THE FRAMEWORK WITHOUT THEM

Name	Y	C_B	C_R	YC_BC_R
Boxer	–24.1%	–10.2%	–8.4%	–21.2%
Loot	–25.0%	–16.9%	–15.9%	–23.3%
Soldier	–24.0%	–2.9%	–3.8%	–19.9%
Thaidancer	–16.3%	–14.8%	–14.2%	–15.9%
Longdress	–33.6%	–17.0%	–17.5%	–29.9%
Redandblack	–33.3%	–30.0%	–38.3%	–33.5%
Thaidancer (Dynamic)	–20.5%	–17.9%	–16.9%	–19.7%
Average (Static)	**–26.1%**	**–15.3%**	**–16.3%**	**–23.9%**
Enc.time (self)	104%			
Enc.time (child)	116%			
Dec.time (self)	100%			
Dec.time (child)	95%			

it can be seen that the proposed combination achieves similar BD-rate reductions compared with the combined occupancy-map-based RDO and four-neighbor average residual padding scheme. For the static plenoptic point clouds "Soldier" and "RedandBlack," the proposed combination achieves a BD-rate reduction of 0.1% for the Y component.

Since the combined occupancy-map-based RDO and four-neighbor average residual padding scheme handles both the continuous and isolated unoccupied pixels, it covers the performance provided by the proposed block-based group smoothing, which can only deal with the continuous unoccupied pixels. Note that these two algorithms actually have slightly different use cases. The combined occupancy-map-based RDO and four-neighbor average residual padding scheme makes changes to the MV-HEVC encoder while the block-based group smoothing only changes the V-PCC encoder. If the users want to reuse the MV-HEVC encoder or other video encoders in the market, they can choose the block-based group smoothing. If the users want to optimize the performance and can make changes to the MV-HEVC encoder, they can select the combined occupancy-map-based RDO and four-neighbor average residual padding scheme.

E. R-D Curves

1) Comparison With V-PCC: Fig. 12 shows representative R-D curves to compare the proposed framework with the V-PCC independently applied to each view direction [15] to illustrate its benefits. It can be seen that the V-PCC leads to the worst performance since no inter view correlations are utilized. The multiview-video-based framework outperforms the V-PCC significantly through considering the inter view correlations. The proposed block-based group smoothing can lead to a better R-D performance by reducing the bit cost of the continuous unoccupied pixels. The combined occupancy-map-based RDO and four-neighbor average residual padding scheme is able to further achieve some R-D performance improvements by considering the bit cost of both the continuous and isolated unoccupied pixels. In addition, it can be seen that the R-D performance improvements of the proposed efficient unoccupied pixels compression algorithms are larger at high bitrate scenarios. As the blocks with

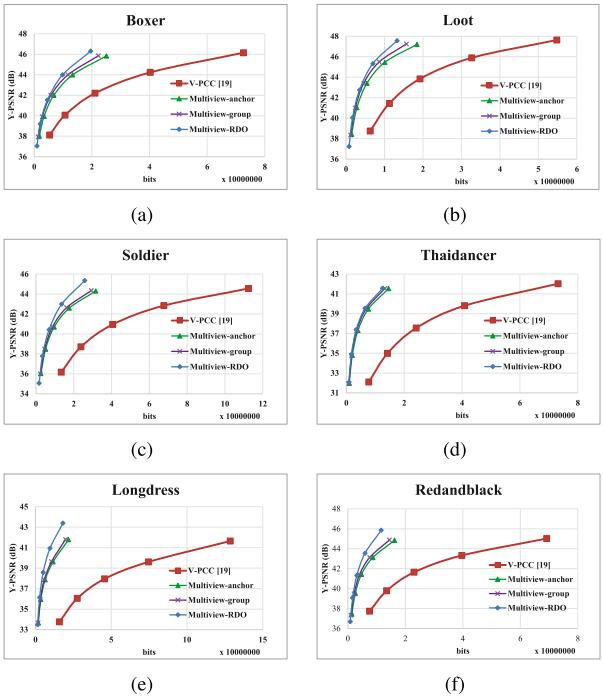


Fig. 12. Representative R-D curves of the plenoptic point clouds compared with V-PCC [15].

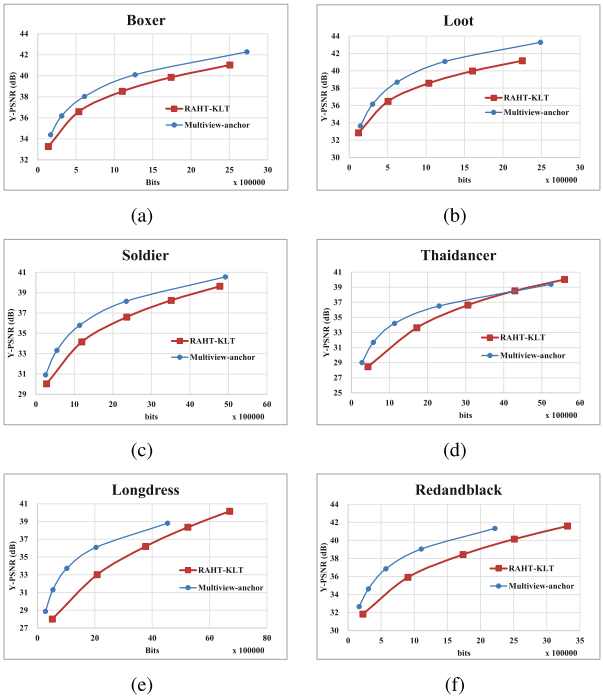


Fig. 13. Representative R-D curves of the plenoptic point clouds compared with the RAHT-KLT [13].

unoccupied pixels are smoother than those with occupied pixels, many of them may select skip mode and only cost a small number of bits in the low bitrate range even without the efficient unoccupied pixels compression algorithms. That is why the proposed algorithms lead to smaller R-D gains in low bitrate scenarios.

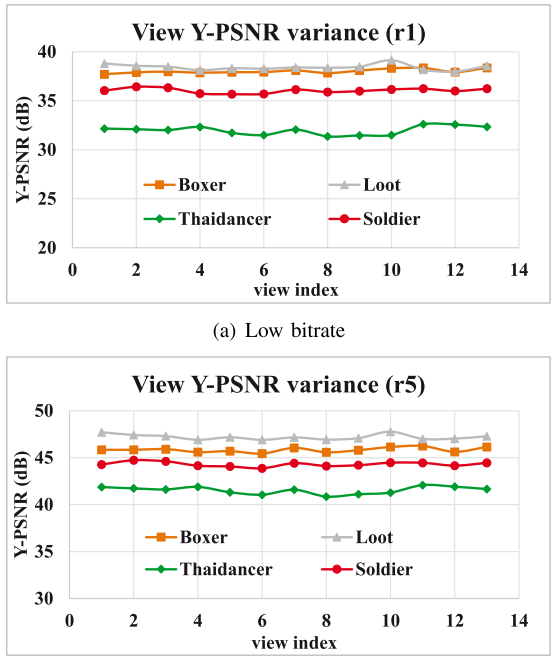


Fig. 14. Y-PSNR variances of different views for various plenoptic point clouds.

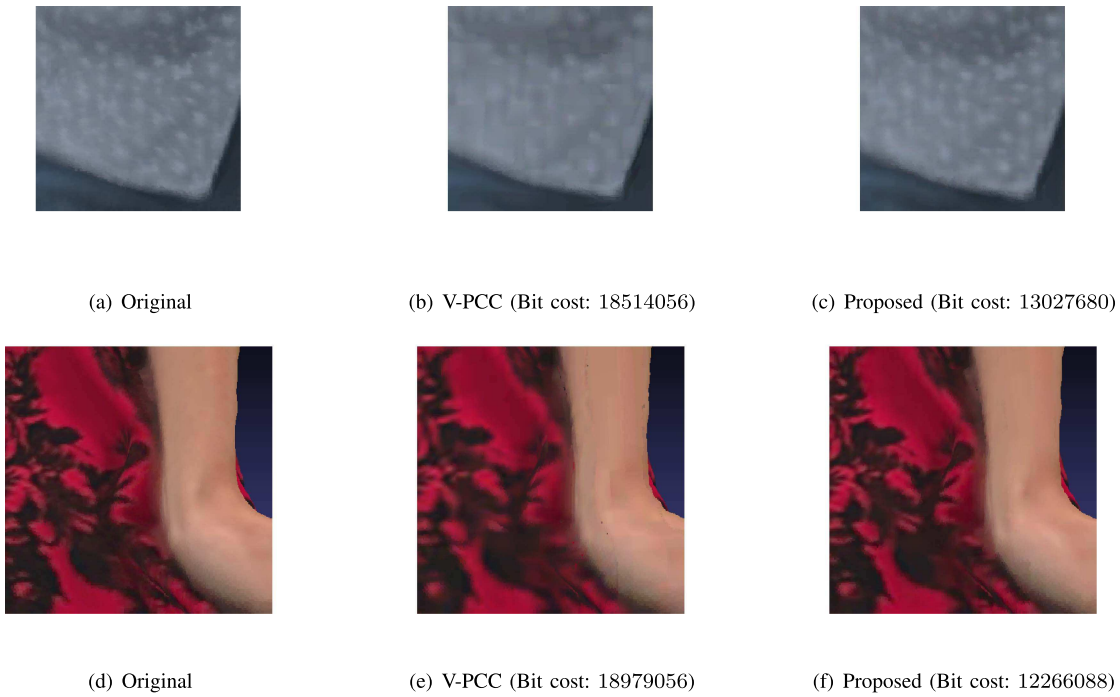
2) Comparison With the RAHT-KLT: Fig. 13 shows representative R-D curves to compare the proposed framework with the RAHT-KLT [13] to illustrate its benefits. It can be seen that the proposed multiview-video-based framework without efficient unoccupied pixel compression algorithms outperforms the RAHT-KLT especially in the medium bitrate case. The proposed algorithm achieves significant BD-rate reductions compared with the RAHT-KLT for most plenoptic point clouds under all bitrate scenarios. However, it can also be seen that the proposed algorithm suffers some performance losses in the high bitrate case for the plenoptic point cloud "Thaidancer".

F. Quality Variance Across Various Views

In addition to the average R-D performance, Fig. 14 shows the Y-PSNR variances of different views for various plenoptic point clouds to explain the influence of hierarchical bit allocation on the quality fluctuation. From Fig. 14, it can be seen that the PSNR differences across various views are within 1 dB for various plenoptic point clouds in both low and high bitrate scenarios. The experimental results demonstrate that the proposed multiview-video-based framework does not result in serious quality variances across various views.

G. Subjective Quality

Fig. 15 shows the subjective quality comparisons between the proposed multiview-video-based framework and the V-PCC [15]. From left to right, sub-figures show original point clouds without compression, reconstructed point clouds from the V-PCC, and reconstructed point clouds from the proposed



(a) Original (b) V-PCC (Bit cost: 18514056) (c) Proposed (Bit cost: 13027680)

(d) Original (e) V-PCC (Bit cost: 18979056) (f) Proposed (Bit cost: 12266088)

Fig. 15. Subjective quality comparison between the proposed multiview-video-based framework and the V-PCC. Sub-figures from (a) to (c) are cropped from the point cloud "Loot" and sub-figures from (d) to (f) are cropped from the point cloud "RedandBlack".

multiview-video-based framework with the efficient unoccupied pixel compression algorithms. Sub-figures from (a) to (c) are cropped from the point cloud "Loot" and sub-figures from (d) to (f) are cropped from the point cloud "Redandblack". From sub-figure (b), we can see that the texture of the shirt of the man is blurred. In addition, from sub-figure (e), we can see that there are some serious color artifacts at the boundary between the clothes and the hand of the woman. However, those artifacts are not shown in sub-figures (c) and (f) generated from the proposed algorithm. The experimental results show that the proposed multiview-video-based framework can achieve significant subjective quality improvements compared with the V-PCC.

VI. Conclusion

In this paper, we propose a multiview-video-based framework to compress plenoptic point cloud colors efficiently. First, we propose projecting a plenoptic point cloud to its bounding box using a projection process similar to that of the moving pictures experts group (MPEG) video-based point cloud compression (V-PCC) [11] to generate a geometry video and multiple color videos from various viewpoints. The multiple color videos are then proposed to be compressed using the Multiview extension of High Efficiency Video Coding (MV-HEVC). Second, we propose a block-based group smoothing algorithm to unify the unoccupied pixels in the view direction to reduce the bit cost of the continuous unoccupied pixels. Third, we propose a combined occupancy-map-based RDO and four-neighbor average residual padding scheme to further reduce the bit cost of the unoccupied pixels. The proposed framework is implemented in the MEPG V-PCC and the MV-HEVC reference software. The experimental results show that the proposed multiview-video-based framework with and without the efficient unoccupied pixel compression algorithms significantly outperforms the combination of the region-based adaptive hierarchical transform and the Karhunen-Loève transform (RAHT-KLT) and the V-PCC independently applied to each view direction.

References

[1] M.-L. Champel, R. Doré, and N. Mollet, "Key factors for a high-quality VR experience," in *Proc. Softw. Process. Improvement Example Opt. Eng. Appl.*, 2017, vol. 10396, pp. 183–194.

[2] G. Bruder, F. Steinicke, and A. Nüchter, "Poster: Immersive point cloud virtual environments," in *Proc. IEEE Symp. 3D User Interfaces (3DUI)*, 2014, pp. 161–162.

[3] C. Tulvan, R. Mekuria, Z. Li, and S. Laserre, "Use cases for point cloud compression," ISO/IEC JTC1/SC29/WG11 MPEG2015/N16331, Geneva, CH, Switzerland, Jun. 2016.

[4] M. Krivokuća, P. A. Chou, and P. Savill, "8i Voxelized surface light field (8iVSLF) dataset," ISO/IEC JTC1/SC29/WG11 m42914, Ljubljana, Slovenia, Jul. 2018.

[5] R. Schnabel and R. Klein, "Octree-based point-cloud compression," *SPBG*, vol. 6, pp. 111–120, 2006.

[6] P. de Oliveira Rente, C. Brites, J. Ascenso, and F. Pereira, "Graph-based static 3D point clouds geometry coding," *IEEE Trans. Multimedia*, vol. 21, no. 2, pp. 284–299, Feb. 2019.

[7] B. Kathariya, L. Li, Z. Li, J. Alvarez, and J. Chen, "Scalable point cloud geometry coding with binary tree embedded quadtree," in *Proc. IEEE Int. Conf. Multimedia Expo.*, 2018, pp. 1–6.

[8] S. Park and S. Lee, "Multiscale representation and compression of 3-D point data," *IEEE Trans. Multimedia*, vol. 11, no. 1, pp. 177–183, Jan. 2009.

[9] L. He, W. Zhu, and Y. Xu, "Best-effort projection based attribute compression for 3D point cloud," in *Proc. 23rd Asia-Pacific Conf. Commun.*, 2017, pp. 1–6.

[10] M. Krivokuća, P. A. Chou, and M. Koroteev, "A volumetric approach to point cloud compression-Part II: Geometry compression," *IEEE Trans. Image Process.*, vol. 29, pp. 2217–2229, 2020.

[11] S. Schwarz *et al.*, "Emerging MPEG standards for point cloud compression," *IEEE Trans. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 1, pp. 133–148, Mar. 2019.
[12] K. Mammou, A. M. Tourapis, D. Singer, and Y. Su, "Video-based and hierarchical approaches point cloud compression," Document ISO/IEC JTC1/SC29/WG11 m41649, Macau, China, Oct. 2017.
[13] G. Sandri, R. L. de Queiroz, and P. A. Chou, "Compression of plenoptic point clouds," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1419–1427, Mar. 2019.
[14] H. Liu, H. Yuan, Q. Liu, J. Hou, and J. Liu, "A comprehensive study and comparison of core technologies for MPEG 3-D point cloud compression," *IEEE Trans. Broadcast.*, vol. 66, no. 3, pp. 701–717, Sep. 2020.
[15] D. Naik, M. Pesonen, and S. Schwarz, "[V-PCC] On surface light field support for TMC2," Document ISO/IEC JTC1/SC29/WG11 MPEG2019/M46057, Marrakesh, MA, Morocco, Jan. 2019.
[16] S. Rhyu, Y. Oh, and J. Woo, "PCC CE2.13 Report on texture and depth padding improvement," Document ISO/IEC JTC1/SC29/WG11 m43667, Ljubljana, Slovenia, Jul. 2018.
[17] L. Li, Z. Li, S. Liu, and H. Li, "Occupancy-map-based rate distortion optimization and partition for video-based point cloud compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 326–338, Jan. 2021.
[18] L. Li, Z. Li, S. Liu, and H. Li, "Efficient projected frame padding for video-based point cloud compression," *IEEE Trans. Multimedia*, vol. 23, pp. 2806–2819, 2021.
[19] M. M. Hannuksela, Y. Yan, X. Huang, and H. Li, "Overview of the multiview high efficiency video coding (MV-HEVC) standard," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2015, pp. 2154–2158.
[20] C. Zhang, D. Florncio, and C. Loop, "Point cloud attribute compression with graph transform," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2014, pp. 2066–2070.
[21] Y. Shao, Z. Zhang, Z. Li, K. Fan, and G. Li, "Attribute compression of 3D point clouds using Laplacian sparsity optimized graph transform," in *Proc. IEEE Vis. Commun. Image Process.*, 2017, pp. 1–4.
[22] R. L. de Queiroz and P. A. Chou, "Transform coding for point clouds using a Gaussian process model," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3507–3517, Jul. 2017.
[23] R. L. de Queiroz and P. A. Chou, "Compression of 3D point clouds using a region-adaptive hierarchical transform," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3947–3956, Aug. 2016.
[24] R. S. Krivokuća and B. J. Falkowski, "The haar wavelet transform: Its status and achievements," *Comput. Elect. Eng.*, vol. 29, no. 1, pp. 25–44, 2003.
[25] P. A. Chou, M. Koroteev, and M. Krivokuća, "A volumetric approach to point cloud compression-Part I: Attribute compression," *IEEE Trans. Image Process.*, vol. 29, pp. 2203–2216, 2020.
[26] S. Gu, J. Hou, H. Zeng, H. Yuan, and K. Ma, "3D point cloud attribute compression using geometry-guided sparse representation," *IEEE Trans. Image Process.*, vol. 29, pp. 796–808, 2020.
[27] R. A. Cohen, D. Tian, and A. Vetro, "Point cloud attribute compression using 3-D intra prediction and shape-adaptive transforms," in *Proc. Data Compression Conf.*, Mar. 2016, pp. 141–150.
[28] Y. Shao, Q. Zhang, G. Li, Z. Li, and L. Li, "Hybrid point cloud attribute compression using slice-based layered structure and block-based intra prediction," in *Proc. 26th ACM Int. Conf. Multimedia*, 2018, pp. 1199–1207.
[29] K. Mammou *et al.*, "Lifting scheme for lossy attribute encoding in TMC1," Document ISO/IEC JTC1/SC29/WG11 m42640, San Diego, CA, USA, Apr. 2018.
[30] V. Zakharchenko, B. Kathariya, and J. Chen, "[G-PCC] [CE13.15] Response on level of detail generation using binary tree for lifting transform," Document ISO/IEC JTC1/SC29/WG11 m45966, Marrakech, MA, Morocco, Jan. 2019.
[31] K. Mammou, A. Tourapis, and J. Kim, "[G-PCC New Proposal] Efficient Low-Complexity LOD Generation.," ISO/IEC JTC1/SC29/WG11 m46188, Marrakech, MA, Morocco,, Jan. 2019.
[32] P. N. Hong and C. W. Ahn, "Unsupervised learning for stereo matching using single-view videos," *IEEE Access*, vol. 8, pp. 73 804–73 815, 2020.
[33] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.
[34] R. Mekuria, K. Blom, and P. Cesar, "Design, implementation, and evaluation of a point cloud codec for tele-immersive video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 4, pp. 828–842, Apr. 2017.
[35] M. Budagavi, E. Faramarzi, T. Ho, H. Najaf-Zadeh, and I. Sinharoy, "Samsungs response to CfP for point cloud compression (Category 2)," Document ISO/IEC JTC1/SC29/WG11 m41808, Macau, China, Oct. 2017.
[36] S. Schwarz *et al.*, "Nokias response to CfP for point cloud compression (Category 2)," Document ISO/IEC JTC1/SC29/WG11 m41779, Macau, China, Oct. 2017.
[37] S. Lasserre, J. Llach, C. Guede, and J. Ricard, "Technicolors response to the CfP for point cloud compression," Document ISO/IEC JTC1/SC29/WG11 m41822, Macau, China, Oct. 2017.
[38] M. Goncalves, L. Agostini, D. Palomino, M. Porto, and G. Correa, "Encoding efficiency and computational cost assessment of state-of-the-art point cloud codecs," in *Proc. IEEE Int. Conf. Image Process.*, 2019, pp. 3726–3730.
[39] G. Sandri, R. De Queiroz, and P. A. Chou, "Compression of plenoptic point clouds using the region-adaptive hierarchical transform," in *Proc. 25th IEEE Int. Conf. Image Process.*, 2018, pp. 1153–1157.
[40] M. Krivokuća and C. Guillemot, "Colour compression of plenoptic point clouds using raht-klt with prior colour clustering and specular/diffuse component separation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 1978–1982.
[41] X. Zhang *et al.*, "Surface light field compression using a point cloud codec," *IEEE Trans. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 1, pp. 163–176, Mar. 2019.
[42] D. Graziosi, "[V-PCC] TMC2 optimal texture packing" Document ISO/IEC JTC1/SC29/WG11 m43681, Ljubljana, SI, Slovenia Jul. 2018.
[43] D. Graziosi, "V-PCC new proposal (Related to CE2.12): Harmonic background filling," Document ISO/IEC JTC1/SC29/WG11 m46212, Marrakesh, MA, Morocco, Jan. 2019.
[44] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B pictures and MCTF," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2006, pp. 1929–1932.
[45] S. Schwarz, G. Martin-Cocher, D. Flynn, and M. Budagavi, "Common test conditions for point cloud compression," Document ISO/IEC JTC1/SC29/WG11 w17766, Ljubljana, Slovenia, Jul. 2018.
[46] L. Li, Z. Li, S. Liu, and H. Li, "Occupancy-map-based rate distortion optimization for video-based point cloud compression," in *Proc. IEEE Int. Conf. Image Process.*, 2019, pp. 3167–3171.
[47] Point cloud compression category 2 reference software TMC2-7.0, 2022. [Online]. Available: http://mpegx.int-evry.fr/software/MPEG/PCC/TM/mpeg-pcc-tmc2.git
[48] Multiview high efficiency video coding test model, HTM-16.3, 2022. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-16.3
[49] MPEG pc_error software, pc_error, 2022. [Online]. Available: http://mpegx.int-evry.fr/software/MPEG/PCC/mpeg-pcc-dmetric
[50] J. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards-including high efficiency video coding (HEVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1669–1684, Dec. 2012.
[51] G. Bjontegaard, "Calculation of average PSNR differences between RD-Curves," Document VCEG-M33, Austin, Texas, USA, Apr. 2001.

Li Li (Member, IEEE) received the B.S. and Ph.D. degrees in electronic engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2011 and 2016, respectively. He is currently a Professor with the Department of Electronic Engineering and Information Science, USTC. From 2016 to 2020, he was a Visiting Assistant Professor with the University of Missouri-Kansas City, Kansas City, MO, USA.

His research interests include image/video coding and processing. He was the recipient of the Best 10% Paper Award at the 2016 IEEE Visual Communications and Image Processing (VCIP) and the 2019 IEEE International Conference on Image Processing (ICIP).

Zhu Li (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Northwestern University, Evanston, IL, USA, in 2004. He is currently an Associate Professor with the Department of CSEE, University of Missouri, Kansas City, MO, USA, directs the NSF I/UCRC Center for Big Learning, UMKC. He was the AFRL Summer Faculty with U.S. Air Force Academy, UAV Research Center, 2016, 2017, and 2018. He was a Senior Staff Researcher/Senior Manager with Samsung Research America's Multimedia Core Standards Research Lab in Dallas, from 2012 to 2015, Senior Staff Researcher with FutureWei, from 2010 to 2012, an Assistant Professor with the Department of Computing, The HongKong Polytechnic University, HongKong, from 2008 to 2010, and a Principal Staff Research Engineer with the Multimedia Research Lab (MRL), Motorola Labs, Schaumburg, IL, USA, from 2000 to 2008. He has 46 issued or pending patents, more than 100 publications in book chapters, journals, conference proceedings and standards contributions in his research field, which include image/video analysis, compression, and communication and associated optimization, and machine learning problems.

He is an Associate Editor-in-Chief (AEiC) for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, 2020-, and was / is an Associated Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING 2019-, IEEE TRANSACTIONS ON MULTIMEDIA during 2015–2019, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY during 2016–2019. He was the recipient of the Best Paper Award from IEEE Int'l Conf on Multimedia & Expo (ICME) at Toronto, 2006, and Best Paper Award from IEEE Int'l Conf on Image Processing (ICIP) at San Antonio, 2007.

Shan Liu (Senior Member, IEEE) received the B.Eng. degree in electronics engineering from Tsinghua University, Beijing, China, and the M.S. and Ph.D. degrees in electrical engineering from the University of Southern California, Los Angeles, CA, USA.

She is currently a Tencent Distinguished Scientist and General Manager of Tencent Media Lab. She was formerly the Director of Media Technology Division, MediaTek, USA. She was also formerly with MERL, Sony, and IBM. Dr. Liu has been actively contributing to international standards since the last decade. She has numerous proposed technologies adopted into various standards, such as HEVC, VVC, OMAF, DASH, and PCC, and was the Co-Editor of HEVC SCC and the emerging VVC. At the same time, technologies and products developed under her leadership have reached more than 10 million DAU. Dr. Liu holds more than 150 granted U.S. and global patents and has authored or coauthored more than 80 peer reviewed technical articles. She was in the committee of Industrial Relationship of IEEE Signal Processing Society during 2014–2015. She is currently on the Editorial Board of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY during 2018–2021 and is the Vice Chair of IEEE Data Compression Standards Committee in 2019. She was also the VP of Industrial Relations and Development of Asia-Pacific Signal and Information Processing Association during 2016–2017 and was named APSIPA Industrial Distinguished Leader in 2018.

Houqiang Li (Fellow, IEEE) received the B.S., M.Eng., and Ph.D. degrees in electronic engineering from the University of Science and Technology of China, Hefei, China, in 1992, 1997, and 2000, respectively. He is currently a Professor with the Department of Electronic Engineering and Information Science, University of Science and Technology of China.

He has authored or coauthored more than 200 papers in journals and conferences. His research interests include multimedia search, image/video analysis, video coding and communication. He was the winner of National Science Funds (NSFC) for Distinguished Young Scientists, the Distinguished Professor of Changjiang Scholars Program of China, and the Leading Scientist of Ten Thousand Talent Program of China. From 2010 to 2013, he was an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. He was the TPC Co-Chair of VCIP 2010, and he was the General Co-Chair of ICME 2021. He was the recipient of the National Technological Invention Award of China (second class) in 2019 and the National Natural Science Award of China (second class) in 2015. He was the recipient of the Best Paper Award for VCIP 2012, Best Paper Award for ICIMCS 2012, and Best Paper Award for ACM MUM in 2011.