Multiview-Video-Based Compression of Plenoptic Point Clouds

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

***Abstract*—The plenoptic point cloud that has associated colors from various directions, is a more complete representation of a 3-D object than the general point cloud that has only one color. It is more realistic but also brings a larger volume of data that needs to be compressed. The state-of-the-art method to compress the plenoptic point cloud is the multiple attributes extension of the region-based adaptive hierarchical transform (RAHT). In addition to RAHT, the video-based point cloud compression (V-PCC) is also an efficient method to compress the point cloud. However, there are not any works using a video-based solution to compress the plenoptic point cloud yet. Therefore, in this paper, we provide a Multiview-video-based framework utilizing the high efficiency of the multiview video coding standard to compress the plenoptic point cloud efficiently. Under the proposed multiview-video-based framework, a plenoptic point cloud is projected to its bounding box to generate multiple attribute videos since it has multiple colors from various directions. Then the multiple attribute videos are compressed efficiently using Multiview High Efficiency Video Coding (MV-HEVC). To the best of our knowledge, this is the first work to compress the plenoptic point cloud using a video-based solution. To further improve the performance of the proposed framework, we propose two methods to reduce the bit cost of unoccupied pixels that are useless for the reconstructed quality of the plenoptic point cloud. First, we propose a block-based group padding scheme to unify the unoccupied pixels across the attribute direction to minimize the bit cost of the unoccupied pixels. Second, we propose ignoring the distortion of the unoccupied pixels during the rate distortion optimization in MV-HEVC. The proposed algorithms are implemented in the V-PCC and the MV-HEVC reference software. The experimental results show that the proposed algorithms can lead to significant bitrate savings compared with the state-of-the-art method.**

***Index Terms*—Multiview High Efficiency Video Coding, Plenoptic point cloud, Point cloud compression, Rate distortion optimization, Video-based point cloud compression**

I. Introduction

The general point cloud is a set of 3-D points that can be used to represent a 3-D surface. Each point is usually associated with one single color along with other attributes. The point cloud 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]. However, the general point cloud with only one single color is not realistic since the colors of the real world objects may vary significantly along with the change of the view angles. Recently, 8i captures several plenoptic point clouds with associated colors from various view angles [4]. The plenoptic point cloud is a more complete representation of a 3-D object than the general point cloud that has only one color.

The plenoptic point cloud is more realistic than the general point cloud. However, it also brings a much larger volume of data. For example, for a general point cloud captured by 8i using the camera plus depth sensors, it has around one million points [5]. With each point represented by 30 and 24 bits for the geometry and attribute, respectively, a general point cloud can be as large as $6M$ bytes without compression. For a plenoptic point cloud with 14 views, the size can be as large as $45M$ bytes. The plenoptic point clouds bring more burdens to the communication and storage than the general point clouds. Therefore, there is an urgent need to compress them more efficiently.

Both the geometry and attribute need to be signaled to compress a point cloud. Octree and its variations [6] [7] are typical methods to compress the geometry. Some methods also introduce plane [8] [9] or mapping [10] to improve the geometry compression efficiency. However, the main difference between the plenoptic point cloud and the general point cloud with one single color is the attribute. Therefore, we put more focuses on the introduction of the point cloud attribute compression in this paper.

The attribute compression methods of the general point cloud can be roughly divided into two groups. The first group is the 3-D-based method. The most common 3-D-based method is the transform-based method that utilizes the information in the geometry to generate a geometry-aware transform. The transform is then applied to the attribute to remove the correlations among the attributes. Among all the transforms, the region-based adaptive hierarchical transform (RAHT) [11] has been adopted as part of the point cloud compression standard [12] because it shows a good trade-off between compression efficiency and complexity. In addition, the RAHT has been further extended to support the plenoptic point cloud compression with multiple attributes [13]. However, the correlations among various attributes have not been substantially exploited with only the transform.

L. Li and Z. Li are with the Department of Computer Science and Electrical Engineering, University of Missouri-Kansas City, MO 64110, USA. L. Li is also 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. Professor Zhu Li is the corresponding author (e-mail: lil1@umkc.edu; lizhu@umkc.edu).

S. Liu is with Tencent America, 661 Bryant St, Palo Alto, CA 94301 (e-mail: shanl@tencent.com).

H. Li is 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: lihq@ustc.edu.cn).

The second group is the projection-based method. The most common projection-based method is the video-based method that projects the point cloud to 2-D videos and compresses them using 2-D video compression standard. A variety of methods have been proposed to project the point cloud to 2-D videos. Among them, the video-based point cloud compression (V-PCC) first projects the point cloud to its bounding box patch by patch [14]. The patches are then packed into a video that is compressed using High Efficiency Video Coding (HEVC) [15]. Due to the high efficiency of the 2-D video compression pipeline including prediction, transform, quantization, and entropy coding, the video-based method shows a good compression efficiency for the point cloud. The V-PCC has been adopted as part of the point cloud compression standard [12] because it shows a good balance between the number of points projected and the video compression efficiency. However, to the best of our knowledge, no methods have been proposed to compress the plenoptic point cloud using the video-based method to exploit the view correlations.

Therefore, in this paper, we propose a Multiview-video-based framework utilizing the high efficiency of the multiview video coding standard to compress the plenoptic point cloud efficiently. The proposed framework has the following key contributions.

- We propose projecting the plenoptic point cloud to its bounding box to generate multiple attribute videos since it has associated colors from various directions. Then the multiple attribute videos are compressed efficiently using Multiview High Efficiency Video Coding (MV-HEVC) [16]. To the best of our knowledge, this is the first work that compresses the plenoptic point cloud with a multiview-video-based framework.
- We propose a block-based group padding scheme to unify the unoccupied pixels across the view and temporal directions to reduce the bit cost of the unoccupied pixels. In this way, the unoccupied blocks will have zero residues after prediction, and thus fewer bits will be spent on the unoccupied pixels.
- We propose ignoring the distortions of the unoccupied pixels during the rate distortion optimization (RDO) in MV-HEVC to further reduce the bit cost of the unoccupied pixels. In addition, some occupied pixels will obtain a good prediction without considering the distortions of the unoccupied pixels. This also contributes to part of the performance improvements.

Part of the works has been proposed in our previous work [17]. In this work, we give a more comprehensive analysis and more experimental results on the proposed block-based group padding scheme. In addition, we propose optimizing the MV-HEVC encoder by ignoring the distortions of the unoccupied pixels for the plenoptic point cloud compression. The experimental results show that the proposed algorithm leads to much better performance compared with that of the previous work.

The rest of this paper is organized as follows. We will review the related works in Section II. The proposed multiview-video-based compression framework will be described in Section II-A in detail. The proposed methods to handle the unoccupied pixels will be introduced in Section IV Section V will show the experimental results and analysis. Section VI will conclude the paper.

II. Related Work

The main difference between the general point cloud and the plenoptic point cloud is the attribute. We will focus on the review of the point cloud attribute compression methods in this section. In subsection II-A and subsection II-B, we will review the 3-D-based method and projection-based method, respectively.

A. 3-D-based method

The 3-D-based method can be roughly divided into the transform-based method and prediction-based method. The essence of the transform-based method is utilizing the correlation between the geometry and attribute to derive a geometry-aware transform from the geometry to compress the attribute. Zhang *et al.* [18] first proposed using Graph Fourier Transform (GFT) to exploit the correlations in the geometry to compress the attribute. Shao *et al.* [19] further improved the GFT with optimized Laplacian sparsity to select a better GFT for each local block. In addition, Queiroz and Chou [20] proposed Gaussian Process Transform (GPT) that is the Karhunen-Loève transform of the Gaussian process to compress the attribute. However, these methods need to solve a complex Eigenproblem to derive the transform and will significantly increase the encoder and decoder complexity. To address this problem, Queiroz and Chou [11] introduced RAHT to compress the attribute to get a better balance between the complexity and the performance. RAHT is a wavelet transform [21] weighted by the cell occupancy so that it can substantially utilize the information in the geometry. Due to a good balance between the complexity and the performance, RAHT has been adopted as part of the point cloud compression standard [12] in the Moving Pictures Experts Group (MPEG). Sandri *et al.* [22] [13] extended the RAHT to RAHT-DCT or RAHT-KLT to further exploit the correlations among multiple attributes to compress the plenoptic point cloud. However, those correlations among various views cannot be fully utilized with only the transform. In addition, Gu *et al.* [23] proposed using a geometry-guided sparse representation to compress the attribute.

In addition to the transform-based method, the prediction is also a common way to decorrelate signals. Cohen *et al.* [24] introduced a 3-D intra prediction method using the neighboring reconstructed points to predict the current points. Shao *et al.* [25] proposed decomposing the point cloud into two slices and introduced intra prediction to predict them separately. These methods divide the point cloud into multiple blocks and perform the intra prediction block by block. In addition, Mammou *et al.* [14] introduced a layer structure and used the point cloud with coarse granularity to predict that with finer granularity. A lifting scheme is further proposed in [26] to improve its performance. This method shows good compression performance for the sparse point cloud compression, and

thus it has been adopted as part of the point cloud compression standard in MPEG [12]. During the standardization process, some works focus on deriving an optimal layer-based structure using the kd-tree [27] or the neighboring information [28].

B. Projection-based method

The projection-based method projects the point cloud to the 2-D domain to utilize the efficient image or video compression standard. Rufael *et al.* [29] proposed projecting the points into an image through a depth-first tree traversal. The image is then compressed using image compression standard such as Joint Photographic Experts Group (JPEG). Lasserre *et al.* [30] and Budagavi *et al.* [31] proposed sorting the points directly into a video using an octree or point position in a lossless manner. However, the videos generated are unsuitable for the current video compression framework since the spatial and temporal correlations are limited. To solve this problem, Schwarz *et al.* [32] and He *et al.* [10] proposed projecting the point cloud to video using cubes or cylinders and unfolding it into a 2-D video. The generated videos have high spatial and temporal correlations. However, many points are lost due to occlusion. Mammou *et al.* [14] proposed a patch-based method to project the point cloud to the cube patch by patch and organized all the patches into a video. This method won the MPEG call for proposal for the dynamic point cloud compression. In addition, it has also been demonstrated as an effective method to compress the static point cloud [33]. Moreover, Zhang *et al.* [34] proposed projecting the multiview images to a point cloud for compression. However, to the best of our knowledge, there are not any works using a video-based solution to exploit the correlations among various views to compress the plenoptic point cloud yet.

III. Multiview Video-based Plenoptic Point Cloud Compression Framework

To compress a plenoptic point cloud using a video-based solution, we first project the plenoptic point cloud to a 2-D video using the same way as in V-PCC [12]. The projection from a plenoptic point cloud to a 2-D video in the V-PCC can be roughly divided into three steps: patch generation, patch packing, and patch padding. A plenoptic point cloud is first divided into several patches by projecting it to its 3-D bounding box. Roughly speaking, each patch is generated by clustering the neighboring points with similar normals together. Compared with the face-based method in [32], the patch-based method has the following two benefits. First, more points will be projected so that the reconstructed point cloud will have better quality. Second, the points with similar normals clustered together will generate a geometry video with fewer variances, which will lead to less bit cost. In addition, each 3-D patch is projected two 2-D patches to further increase the number of projected points. The two 2-D patches will have small differences only when one point obstructed another point. Most pixels with the same values lead to strong "temporal correlations" between the two patches.

After patch generation, 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. For dynamic point cloud compression, some patches are packed in a temporally consistent manner to improve the compression efficiency [35]. In addition, patch rotation is supported to allow more flexible packing so as to improve the compression performance.

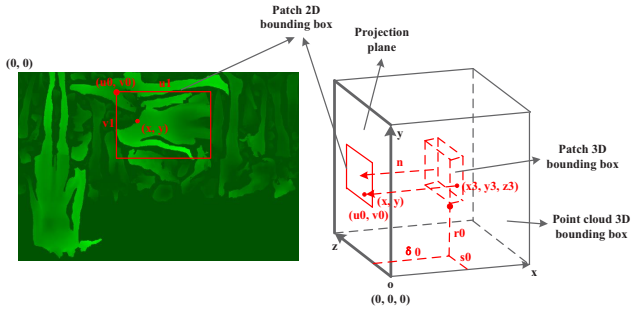

Fig. 1. 3-D to 2-D projection and packing processes.

A lot of empty space exists among various patches after patch generation. The patch packing aims to fill the empty space to make the generated frames more suitable for 2-D video compression. A variety of methods have been proposed for the padding of the geometry and attribute video. For the geometry, the padding is performed block by block in a raster scan order using the neighboring occupied pixels. For the attribute, a push-pull algorithm [36] and a harmonic background filling method [37] have been proposed to make the attribute be compressed more efficiently. The unoccupied pixels in the first and second frames will further go through a group padding to increase the temporal correlations [38].

Fig. 1 shows the patch generation processes and a typical geometry frame generated using the 3-D to 2-D projection. In Fig. 1, a patch with 3-D start coordinate $(\delta 0, r0, s0)$ is projected to a 2-D bounding box $(u0, v0, u1, v1)$ with normal n. The size of the 3-D bounding box is the same as the 2-D bounding box. In this way, the 3-D coordinate $(x3, y3, z3)$ is projected to the 2-D coordinate (x, y) as follows,

$$\begin{cases} x = y3 - s0 + u0 \\ y = z3 - r0 + v0 \end{cases}. \tag{1}$$

Under the proposed framework, the geometry video is first generated, compressed, and reconstructed. Then the attribute video is generated using recoloring, compressed, and reconstructed.

Different from the general point cloud compression that generates only one attribute video, the plenoptic point cloud that has multiple attributes leads to a number of attribute videos after the above processes. The number of attribute videos is determined by the number of cameras from various directions. Fig. 2 gives a typical example of the attribute frames generated from multiple view angles. We can see that these frames are similar to each other despite some pixel differences. These "view correlations" need to be fully utilized to improve the compression efficiency. In addition to the "view correlations", each point cloud is projected to two frames that

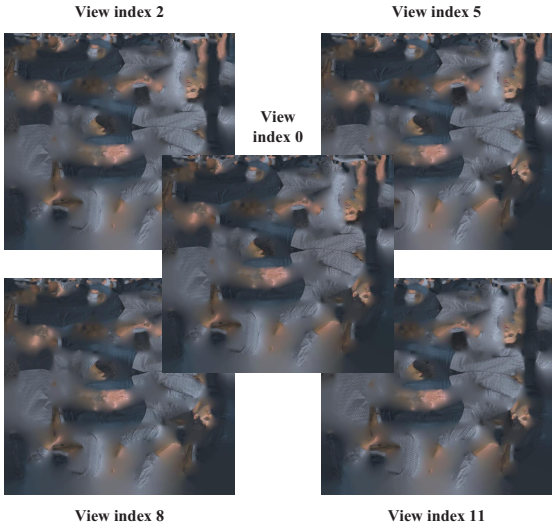

Fig. 2. Examples of the projected views from the plenoptic point cloud "Loot". These views are from the view index 0, 2, 5, 8, and 11, respectively.

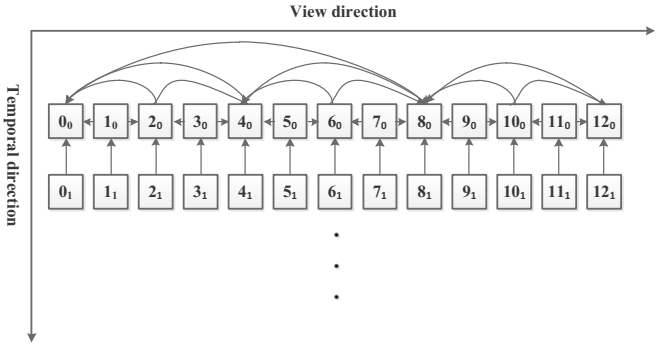

Fig. 3. Proposed multiview-video-based framework.

have strong "temporal correlations" as we have mentioned above.

To fully utilize both the temporal and view correlations, we propose using MV-HEVC to compress these frames efficiently. Using the plenoptic point cloud with 13 views as an example, Fig. 3 shows the proposed multiview-video-based framework. In Fig. 3, the squares with indices 0 to 12 represent the 13 views. The subscripts 0 and 1 indicate the frame index in the temporal direction. The arrows among various views and frames indicate the reference relationships. In the view direction, we use a hierarchical coding structure with GOP size 8. All the views are divided into 4 hierarchical levels. The even and odd views are coded as reference and non-reference frames, respectively. All the views are encoded with B frames using uni-directional and bi-directional prediction to fully utilize the correlations among various views. In the temporal direction, only frame 0 with the same view index is used as the reference of frame 1 since the "temporal correlation" is much higher than the "view correlation".

In addition to the reference relationships, the bit allocation of various views and frames are also important to the compression performance. We set the quantization parameters (QPs) of various views and frames as follows. First, the higher the hierarchical level in the view direction is, the larger the QP is. In the view direction, we set the QP of each level as $QP_I + level + 1$. Second, the second non-reference frame uses a higher QP than the first reference 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.

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 this paper, we mainly deal with the static plenoptic point cloud. The proposed framework can be easily extended to encode the dynamic plenoptic point cloud by extending the reference structure in the temporal direction. When the resolution of each view is too large to use the MV-HEVC structure, the 2-D hierarchical coding structure [39] can be used to compress them efficiently.

IV. Efficient unoccupied pixel compression

Under the proposed multiview-video-based plenoptic point cloud compression framework, the unoccupied pixels padded among various patches are useless for the quality of the reconstructed plenoptic point cloud. Since the number of the unoccupied pixels in the multiple attribute videos is much more than that in one attribute video, this problem becomes much more serious under the multiview video-based plenoptic point cloud compression framework. In this section, we design two methods to minimize the bit cost of the unoccupied pixels from two aspects: the V-PCC and MV-HEVC encoders.

A. Block-based group padding for unoccupied pixels

In the V-PCC for general point cloud compression, several padding methods [36] [37] are first proposed to minimize the bit cost of unoccupied pixels in the first frame. Then, all the unoccupied pixels are padded using the average of the first frame and the second frame to minimize the bit cost of the unoccupied pixels in the second frame [38]. This method exploits the temporal correlations of the unoccupied pixels in the first and second frames to improve compression efficiency. However, since the difference among various views in the view direction is much larger than that in the temporal direction, the isolated unoccupied pixels makes this method not working in the view direction.

Fig. 4 shows a typical example of a projected view with the unoccupied pixels set as black. We can see that the unoccupied pixels can be divided into two groups: the isolated unoccupied pixels indicated by the red square and the continuous unoccupied pixels indicated by the green rectangle. The continuous unoccupied pixels can be padded with the average value of all the unoccupied pixels in the same location across all the view directions. However, the padding of the isolated unoccupied

Fig. 4. Typical example of a projected view with the unoccupied pixels set as black from the plenoptic point cloud "Loot".

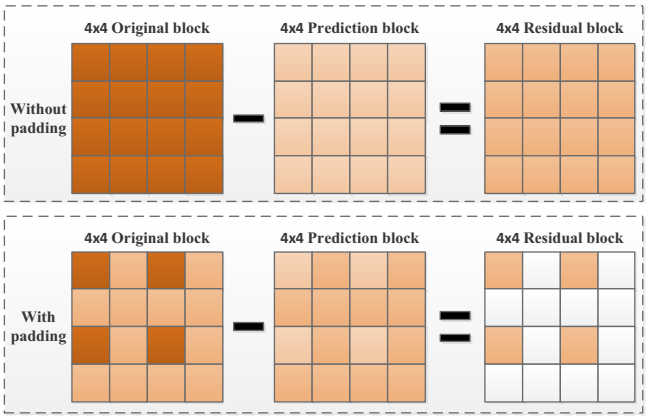

Fig. 5. Influence of the padding of isolated unoccupied pixels.

pixels may destroy the spatial continuity of a block containing both occupied and unoccupied pixels as shown in Fig. 5. We can see that the difference between the original block and the prediction block are larger but more smooth without padding. After padding, the current block and its prediction block will have the same pixel values for the unoccupied pixels but different values for the occupied pixels. In this way, the residual block will have some singular values and it will be unsuitable for the following transform, quantization, and entropy coding processes.

In this paper, we propose a block-based group padding algorithm to filter out the isolated unoccupied pixels. For each pixel, we find the $K \times K$ block with the pixel as the center pixel. Only when all the pixels in the $K \times K$ block are unoccupied, the pixel will be padded across the view direction. In this way, we will not pad the isolated unoccupied pixels and still keep the spatial continuity of a block containing both occupied and unoccupied pixels. The continuous unoccupied pixels will be padded to reduce the bit cost of the unoccupied pixels to improve the compression efficiency. Note that the proposed block-based group padding algorithm will become pixel-based group padding when K is equal to 1. We will discuss the influences of the block size K in the experimental results.

After the detection of the to-be-padded unoccupied pixels, we will pad them using the average value of all the views in both frame 0 and frame 1,

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

where N is number of views of the plenoptic point cloud, i is the frame index, j is the view index, and $f_{i,j}$ is the value of each unoccupied pixel. After the above padding scheme, in both the view and temporal directions, we can obtain very good predictions for the continuous unoccupied pixels. Therefore, the bitrate of the continuous unoccupied pixels can be significantly reduced.

B. Occupancy-based rate distortion optimization

The block-based group padding algorithm can only deal with the continuous unoccupied pixels. In this subsection, we propose handling both the continuous and isolated unoccupied pixels by introducing a occupancy-map-based RDO scheme.

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

$$\min_P J = \sum_{i=1}^{N} D_i(P) + \lambda R(P), \quad (3)$$

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 determining the optimization target. In different stages of the RDO process, 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 we can see from (3), for a block containing both occupied and unoccupied pixels, the distortions of the occupied and unoccupied pixels are accumulated together. This indicates that the default optimization target treats the distortions of the occupied and unoccupied pixels equally. However, the distortions of the unoccupied pixels have no influence on the reconstructed quality of the plenoptic point cloud. Therefore, the default RDO scheme is unsuitable for the multiview-video-based plenoptic point cloud compression framework.

In this paper, we add an occupancy-map-based mask to the distortion in the RDO scheme to solve this problem. We have developed a occupancy-map-based RDO method for the video-based general point cloud compression in our previous work [40]. As the problems of unoccupied pixels are more serious for the plenoptic point cloud compression, we extend the algorithm here to see the performance of the occupancy-map-based RDO scheme in the view direction. The R-D cost of a block after adding the mask is calculated as

$$\min_P J = \sum_{i=1}^{N} D_i(P) \times M_i + \lambda R(P), \quad (4)$$

where M_i is 1 when pixel i is occupied, and M_i is 0 when pixel i is unoccupied. Using this equation, we will only consider the distortions of the occupied pixels when calculating the R-D cost. In the following, we will describe how we fit (4) into the intra prediction, inter prediction, and Sample Adaptive Offset (SAO) in MV-HEVC.

1) Intra prediction: In the MV-HEVC encoder, the intra mode decision can be roughly divided into 3 steps: rough intra mode decision, precise intra mode decision, and residue quadtree decision. The rough intra mode decision generates several best intra mode candidates using the SATD between the original and prediction blocks plus the λ times the bits of the intra prediction direction R_{dir} as the R-D cost,

$$\min_P J = \sum_{i=1}^{N} SATD_i(P) + \lambda R_{dir}(P). \quad (5)$$

During this process, the residue bits are not taken into consideration. If we add the mask to the SATD calculation, we can find a good prediction for the occupied pixels. However, we may find very bad predictions for the unoccupied pixels that will finally lead to a waste of many bits. Therefore, the proposed occupancy-map-based RDO is not applied to the rough intra mode decision process. For the precise mode decision and residue quadtree decision, since the full RDO employing the SSD between the original and reconstruction signals plus the λ times the total bits as the RD cost is used, (4) is applied to the SSD to ignore the distortions of the unoccupied pixels.

2) Inter prediction: In the MV-HEVC encoder, the inter modes can be divided into merge 2Nx2N/skip mode and the other inter modes including inter 2Nx2N and the other partitions. For the merge 2Nx2N/skip mode, as all the merge candidates will go through the full RDO, (4) is applied to ignore the distortions of unoccupied pixels. The other inter modes will first go through the integer and fractional ME to find the MV. In addition, for the partitions except for 2Nx2N, we will calculate the R-D cost of the merge mode to compare with that of the fractional ME. In all these processes, the R-D cost is calculated using

$$\min_P J = \sum_{i=1}^{N} SAD_i(P)/SATD_i(P) + \lambda R_{motion}. \quad (6)$$

As the residue bits are not taken into consideration, we have not applied (4) in the RDO process as explained in the intra prediction subsection. Therefore, for the other inter modes except for merge 2Nx2N/skip mode, the occupancy-map-based RDO is only used to calculate the final R-D cost of inter partitions to compare with the other modes in the full RDO process.

3) Sample adaptive offset: The SAO process can be roughly divided into two steps. First, it calculates the average offsets between the reconstructed and original pixels for different types of edge offset (EO) and band offset (BO). Second, the offsets are determined using RDO on whether they should be added to the reconstructed pixel values of a coding tree unit (CTU) or not. The statistics of the offsets are the key steps to determine the performance of SAO. In the original MV-HEVC encoder, the statistics of the offsets come from all the unoccupied and occupied pixels in the frame. However, the unoccupied pixels are encoded with severe distortions in the occupancy-map-based RDO. Therefore, the average offsets will be determined by the distortions of the unoccupied pixels if the original method is applied.

The offsets derived using the above steps have two disadvantages. First, the occupied CTUs will not have suitable offsets and their R-D performance will be degraded. Second, many bits will be wasted to encode the offsets as they will only be used by the unoccupied CTUs. In this work, we use a simple method to accumulate only the differences of the occupied pixels to calculate the offsets to address the problems. These offsets will be suitable for the occupied CTUs while unsuitable for the unoccupied CTUs. Therefore, the distortions of the occupied CTUs will be reduced while the unoccupied CTUs will not choose SAO to avoid wasting bits.

TABLE II
CHARACTERISTICS OF THE PLENOPTIC POINT CLOUDS

Name	Points	Cameras	Geometry/Attribute 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 [41] and MV-HEVC [42] reference software to compare with the state-of-the-art plenoptic point cloud compression algorithm using RAHT-KLT [13] and the V-PCC without considering the inter correlations [12]. We test all the static plenoptic point clouds proposed in [4]. The characteristics of the static plenoptic point clouds are shown in Table II. Since all the plenoptic point clouds are static, we only test the all intra configuration defined in the V-PCC common test condition [43]. The QPs of the I frames in the proposed multiview-video-based framework are set the same as the attribute QP from the low bitrate (r1) to the high bitrate (r5) to verify the performance of the proposed algorithm in a large bitrate range. The average Peak Signal to Noise Ratio (PSNR) of all the views is used as the quality metric. Since various algorithms may generate different bitrates, the Bjontegaard Delta bitrate (BD-rate) [44] is employed as the performance metric for a fair comparison.

We will first show the performance of the proposed multiview-video-based framework compared with the RAHT-KLT and the V-PCC without considering the inter correlations. Then we will show the performances of the efficient unoccupied pixel compression algorithms in detail. Finally, some examples of the R-D curves will be shown to better illustrate the experimental results.

A. Performance comparison with the state-of-the-art method

Table III shows the performance comparison between the proposed multiview-video-based framework and the state-of-

TABLE III
COMPARISON BETWEEN THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK AND RAHT-KLT [13]

Name	RAHT-KLT		Multiview-video		Y
	Color bits	Y-PSNR	Color bits	Y-PSNR	BD-rate
Boxer	534974	36.58	800592	37.42	1.0%
	1102667	38.51	1469168	39.37	
	2506516	41.02	2646496	41.41	
	4144398	42.77	5030344	43.40	
	7624336	45.08	9660168	45.22	
Loot	505156	36.47	639904	37.71	–31.2%
	1036214	38.57	1214984	40.35	
	2252251	41.16	2183896	42.74	
	3576056	42.91	4010464	44.80	
	6210303	45.21	7266152	46.51	
Soldier	1193244	34.15	1088120	35.37	–26.9%
	2361547	36.60	2077560	37.87	
	3514995	38.24	3741072	40.13	
	7227865	41.62	6740544	42.07	
	11973133	44.15	12000880	43.78	
Thai	434126	28.46	515368	31.29	–42.7%
	1719585	33.63	959752	34.09	
	3058823	36.63	1744408	36.67	
	4292715	38.52	3242152	38.86	
	5599587	40.03	6150648	40.89	
Long	519371	28.01	942000	33.03	–35.4%
	2081546	33.01	1639144	35.37	
	3770193	36.19	2798792	37.40	
	5245716	38.36	4972816	39.13	
	9214122	42.67	9176816	41.17	
Red	224020	31.82	744512	36.61	–16.9%
	903125	35.90	1305184	38.80	
	1736193	38.43	2203496	40.73	
	3313844	41.59	3970184	42.44	
	6081458	45.08	7318392	44.05	
Average	–	–	–	–	–27.0%

the-art plenoptic point cloud compression method RAHT-KLT. We can see that the proposed multiview-video-based framework can lead to 27.0% bitrate savings compared with the RAHT-KLT on average. For the plenoptic point cloud Thaidancer, the proposed framework brings as high as 42.7% bitrate savings. The experimental results demonstrate that the proposed multiview-video-based framework can compress the plenoptic point cloud more efficiently compared with RAHT-KLT. For various rate points, the proposed multiview-video-based framework shows significant bitrate savings compared with the RAHT-KLT in low and medium bitrate cases. While in high bitrate case, the proposed framework leads to similar or even slightly worse performance. This is in accordance with the results shown in [33] that the video-based solution shows very good performance in low and medium bitrate cases. 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, U, and V components.

Table IV shows the performance comparison between the proposed multiview-video framework and the V-PCC solution without considering the correlations among various views. We can see that the proposed algorithm leads to an average of 74.4%, 76.8%, and 77.7% bitrate savings compared with the V-PCC for the Y, Cb, Cr components, respectively. The performance improvements for all the tested plenoptic point clouds are significant and consistent. The experimental results show that the proposed algorithm can fully exploit the correlations among various views so as to improve the performance significantly. The proposed algorithm will lead to some encoding and decoding complexities increase as the inter-view prediction involves more complex mode decision and motion estimation operations compared with intra prediction.

TABLE IV
PERFORMANCE COMPARISON BETWEEN THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK AND THE V-PCC SOLUTION WITHOUT USING THE CORRELATIONS AMONG VARIOUS VIEWS

Name	Y	Cb	Cr
Boxer	–62.4%	–67.1%	–69.2%
Loot	–67.1%	–71.8%	–73.3%
Soldier	–73.6%	–75.1%	–76.1%
Thaidancer	–82.6%	–83.5%	–83.2%
Longdress	–86.5%	–86.6%	–86.5%
Redandblack	–78.1%	–78.1%	–79.1%
Average	–74.4%	–76.8%	–77.7%

TABLE V
PERFORMANCE COMPARISON BETWEEN THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK WITH AND WITHOUT THE BLOCK-BASED GROUP PADDING METHOD

Name	Y	Cb	Cr
Boxer	–18.7%	–13.8%	–16.5%
Loot	–16.5%	–15.7%	–15.0%
Soldier	–9.6%	–7.7%	–7.4%
Thaidancer	–13.3%	–12.2%	–12.6%
Longdress	–8.1%	–8.2%	–8.2%
Redandblack	–13.6%	–13.6%	–14.1%
Average	–13.3%	–11.5%	–13.6%

B. Performance of the efficient unoccupied pixel compression algorithms

1) Block-based padding: The performance comparison between the proposed multiview-video-based framework with and without the block-based group padding method is shown in Table V. The block size is set as 4 in the experimental results shown in Table V. We can see that the proposed block-based padding can bring 13.3%, 11.5%, and 13.6% performance improvements for the Y, Cb, and Cr components, respectively. The experimental results show that the proposed algorithm can bring significant bitrate savings by reducing the bit cost of the unoccupied pixels. Note that with the proposed block-based group padding method, no plenoptic point clouds will suffer performance losses any more compared with the RAHT-KLT including the Boxer.

To show the influences of K on the performance of the block-based group padding, we give the R-D performances of various values of K for different plenoptic point clouds in Table VI. We can see that the block size 4 brings the best performance among all the values on average. However, we can see that the performance varies for different plenoptic point clouds. The block size 4 is the best for most plenoptic point clouds. The block sizes 1, 2, and 8 are the best choices for the plenoptic point cloud Thaidancer, Longdress, and Boxer, respectively. The denser the plenoptic point cloud is, the smaller the block size should be and vice versa.

The average performance is essentially determined by a trade-off between the number of averaged continuous and

TABLE VI
PERFORMANCE OF THE PROPOSED BLOCK-BASED GROUP PADDING METHODS WITH DIFFERENT BLOCK SIZES

Name	Y BD-rate				
	K = 1	K = 2	K = 4	K = 8	K = 16
Boxer	105.2%	–17.9%	–18.7%	–18.8%	–18.0%
Loot	62.2%	–16.0%	–16.5%	–16.1%	–15.3%
Soldier	60.1%	–9.2%	–9.6%	–9.2%	–8.1%
Thaidancer	–13.4%	–13.3%	–13.3%	–12.8%	–11.9%
Longdress	–1.6%	–8.3%	–8.1%	–7.7%	–7.2%
Redandblack	64.4%	–12.6%	–13.6%	–13.6%	–12.7%
Average	42.5%	–13.0%	–13.3%	–12.9%	–12.1%

TABLE VII
PERFORMANCE COMPARISON BETWEEN THE PROPOSED MULTIVIEW-VIDEO-BASED FRAMEWORK WITH AND WITHOUT THE OCCUPANCY-MAP-BASED RDO

Name	Y	Cb	Cr
Boxer	–26.4%	1.6%	2.4%
Loot	–21.0%	–1.1%	3.5%
Soldier	–16.7%	13.2%	13.6%
Thaidancer	–15.8%	–14.8%	–14.8%
Longdress	–17.6%	–7.6%	–7.9%
Redandblack	–23.1%	–15.4%	–18.6%
Average	–19.5%	–1.7%	–0.5%

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 equals to 1, all the continuous and isolated unoccupied pixels will be averaged. The isolated unoccupied pixels destroy the spatial continuity of many blocks, and thus degrade the compression performance significantly. Along with the increase of K from 1 to 4, both the averaged isolated and continuous pixels become less. However, the reduction of the isolated points is the major influence and thus the R-D performance improved gradually. Then if the K further increases, the number of isolated unoccupied pixels keeps almost unchanged while the number of averaged unoccupied pixels keeps decreasing. Therefore, the R-D performance becomes slightly worse.

2) Occupancy-map-based RDO: The performance comparison between the proposed multiview-video-based framework with and without the occupancy-map-based RDO is shown in Table VII. We can see that the proposed occupancy-map-based RDO brings an average of 19.5%, 1.7%, and 0.5% bitrate savings for the Y, Cb, and Cr components, respectively. The proposed occupancy-map-based RDO leads to significant bitrate savings as we have not taken the distortions of the unoccupied pixels into consideration. The proposed algorithm leads to a better performance improvement compared with the block-based group padding as the occupancy-map-based RDO considers both the continuous and isolated unoccupied pixels into consideration. The block-based group padding can only reduce the bit cost of the continuous unoccupied pixels.

To better show why the proposed occupancy-map-based RDO can bring significant bitrate savings, we show the comparison between the reconstructed YUVs with and without the occupancy-map-based RDO in Fig. 6. We can see that the continuous unoccupied pixels become less smooth when the proposed occupancy-map-based RDO is used. This will lead to worse quality for the unoccupied pixels but not have influences on the occupied pixels. The bits saving from the worse quality of the unoccupied pixels is exactly the reason why the occupancy-map-based RDO algorithm can bring significant R-D performance improvements.

(a) With RDO

(b) Without RDO

Fig. 6. Comparison between the reconstructed YUVs with and without the occupancy-map-based RDO.

TABLE VIII
PERFORMANCE OF THE PROPOSED MULTIVIEW-VIDEO FRAMEWORK COMBINING THE PROPOSED BLOCK-BASED GROUP PADDING AND THE OCCUPANCY-MAP-BASED RDO

Name	Y	Cb	Cr
Boxer	–21.4%	7.4%	9.0%
Loot	–18.7%	0.4%	5.3%
Soldier	–14.4%	16.5%	15.9%
Thaidancer	–14.8%	–13.5%	–13.8%
Longdress	–16.2%	–6.5%	–6.4%
Redandblack	–18.6%	–11.2%	–14.7%
Average	–17.1%	0.9%	2.0%

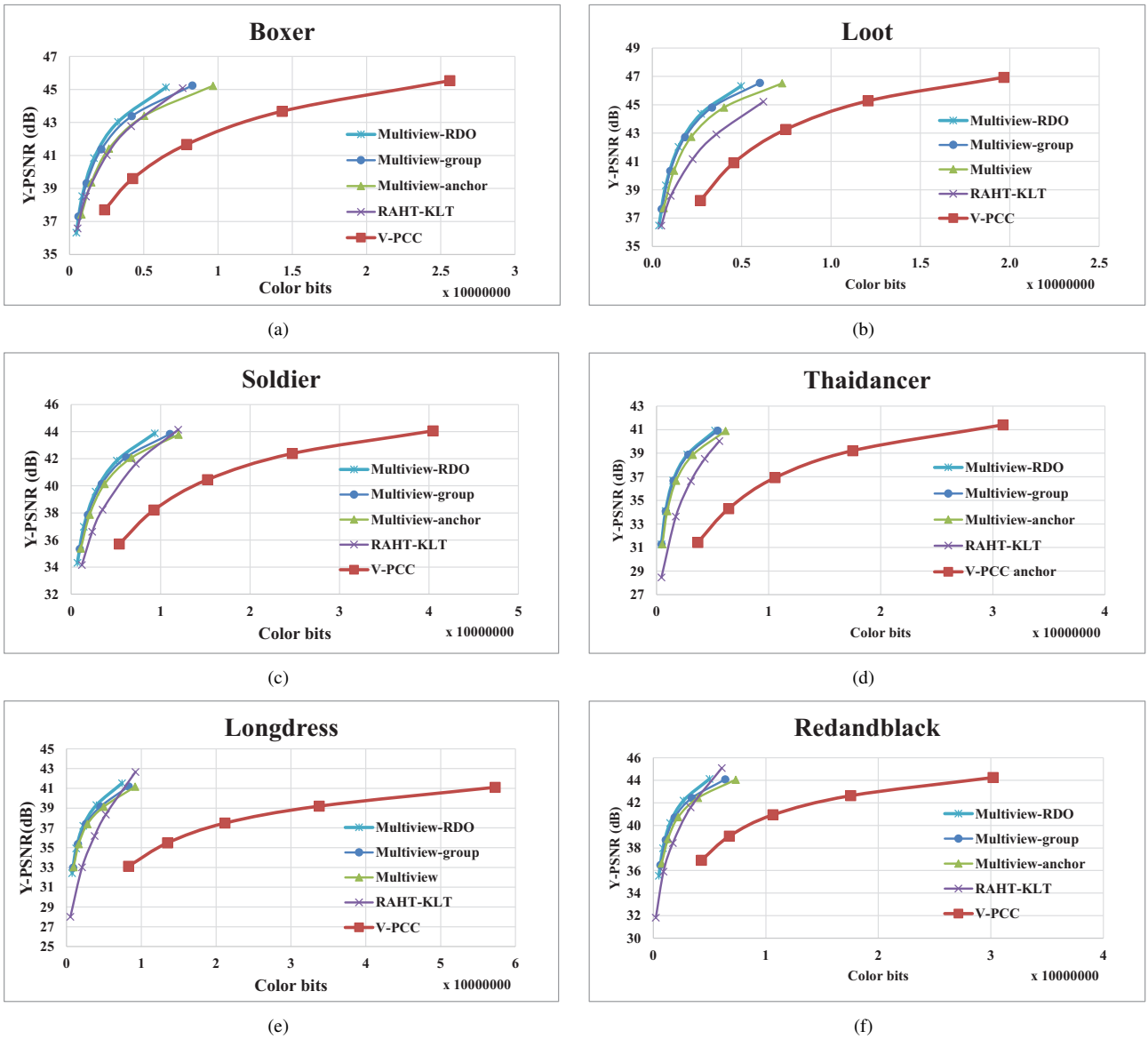

Fig. 7. R-D curves of the plenoptic point clouds.

3) Combination of the block-based group padding and the occupancy-map-based RDO: Table VIII shows the performance of the proposed multiview-video-based framework combining the proposed block-based group padding and the occupancy-map-based RDO. We can see that the proposed combination leads to 17.1% performance improvement for the Y component on average. Comparing Table VIII and Table VII, we can see that the proposed combination leads to some performance losses compared with the occupancy-map-based RDO. The reason is that the block-based padding may lead to spatial discontinuities between neighboring spatial blocks with and without padding and make the occupancy-map-based RDO not working well. Therefore, only one unoccupied pixel compression method is proposed to be used. When the users do not want to change the MV-HEVC encoder, they can choose the block-based padding method. When the users want to optimize the compression performance, they can choose the occupancy-map-based RDO algorithm.

C. R-D curves

To give a better illustration of the R-D performance of the tested plenoptic point clouds, we show some R-D curves of the proposed algorithms in Fig. 7. We can see that the V-PCC always shows the worst performance since no inter view correlations are utilized. Compared with the RAHT-KLT, the proposed multiview-video framework brings significant bitrate savings in low and medium bitrate cases. In high bitrate case, the multiview-video-based framework shows a better R-D performance for some plenoptic point clouds while shows a slightly worse R-D performance for the other plenoptic point clouds. The proposed block-based group padding can lead to better R-D performance by considering the bit cost of the continuous unoccupied pixels. The proposed occupancy-map-based RDO is able to further bring some R-D performance improvements by taking the bit cost of both the continuous and isolated unoccupied pixels into consideration.

VI. Conclusion

In this paper, we propose a multiview-video-based framework to compress the plenoptic point cloud efficiently. First, we propose projecting the plenoptic point cloud to its bounding box to generate a geometry video and multiple attribute videos from various directions. The multiple attribute videos are then proposed to be compressed using multiview-High Efficiency Video Coding (MV-HEVC). Second, we propose a block-based group padding method to unify the unoccupied pixels in the view direction to reduce the bit cost of the unoccupied pixels. Third, we propose an occupancy-map-based RDO method in MV-HEVC to further reduce the bit cost of the unoccupied pixels. The proposed framework is implemented in the video-based point cloud compression and the MV-HEVC reference software. The experimental results show that the proposed method significantly outperforms the state-of-the-art plenoptic point cloud compression method.

References

[1] M.-L. Champel, R. Doré, and N. Mollet, "Key Factors for a High-Quality VR Experience," in *2017 SPIE Optical Engineering and applications*, vol. 10396, 2017.

[2] G. Bruder, F. Steinicke, and A. Nüchter, "Poster: Immersive Point Cloud Virtual Environments," in *2014 IEEE Symposium on 3D User Interfaces (3DUI)*, 2014, pp. 161–162.

[3] C. Tulvan, R. Mekuria, Z. Li, and S. Laserre, "Use Cases for Point Cloud Compression (PCC)," ISO/IEC JTC1/SC29/WG11 MPEG2015/N16331, Geneva, CH, 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, Jul. 2018.

[5] E. d'Eon, B. Harrison, T. Myers, and P. A. Chou, "Input to Ad Hoc Groups on MPEG Point Cloud Compression and JPEG PLENO," Document ISO/IEC JTC1/SC29/WG11 m40059, Geneva, Switzerland, Jan. 2017.

[6] J. Peng and C.-C. J. Kuo, "Geometry-Guided Progressive Lossless 3D Mesh Coding with Octree (OT) Decomposition," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 609–616, 2005.

[7] R. Schnabel and R. Klein, "Octree-based Point-Cloud Compression," *Spbg*, vol. 6, pp. 111–120, 2006.

[8] B. Kathariya, L. Li, Z. Li, J. Alvarez, and J. Chen, "Scalable Point Cloud Geometry Coding with Binary Tree Embedded Quadtree," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2018, pp. 1–6.

[9] P. A. Chou, M. Krivokuca, G. Cernigliaro, and E. dEon, "Point Cloud Compression using a Blockable Geometry Representation and Region Adaptive Hierarchical Transform," ISO/IEC JTC1/SC29/WG11 m41645, Macau, China, Oct. 2017.

[10] L. He, W. Zhu, and Y. Xu, "Best-Effort Projection based Attribute Compression for 3D Point Cloud," in *2017 23rd Asia-Pacific Conference on Communications (APCC)*, Dec 2017, pp. 1–6.

[11] R. L. de Queiroz and P. A. Chou, "Compression of 3D Point Clouds Using a Region-Adaptive Hierarchical Transform," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3947–3956, Aug 2016.

[12] S. Schwarz, M. Preda, V. Baroncini, M. Budagavi, P. Cesar, P. A. Chou, R. A. Cohen, M. Krivokua, S. Lasserre, Z. Li, J. Llach, K. Mammou, R. Mekuria, O. Nakagami, E. Siahaan, A. Tabatabai, A. M. Tourapis, and V. Zakharchenko, "Emerging MPEG Standards for Point Cloud Compression," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 9, no. 1, pp. 133–148, Mar. 2019.

[13] G. Sandri, R. L. de Queiroz, and P. A. Chou, "Compression of Plenoptic Point Clouds," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1419–1427, March 2019.

[14] 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.

[15] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.

[16] M. M. Hannuksela, Y. Yan, X. Huang, and H. Li, "Overview of the Multiview High Efficiency Video Coding (MV-HEVC) Standard," in *2015 IEEE International Conference on Image Processing (ICIP)*, Sep. 2015, pp. 2154–2158.

[17] L. Li, Z. Li, S. Liu, and H. Li, "Video-based compression for plenoptic point clouds," in *arxiv 1911.01355*, 2019.

[18] C. Zhang, D. Florncio, and C. Loop, "Point Cloud Attribute Compression with Graph Transform," in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct 2014, pp. 2066–2070.

[19] Y. Shao, Z. Zhang, Z. Li, K. Fan, and G. Li, "Attribute Compression of 3D Point Clouds Using Laplacian Sparsity Optimized Graph Transform," in *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE, 2017, pp. 1–4.

[20] R. L. de Queiroz and P. A. Chou, "Transform Coding for Point Clouds Using a Gaussian Process Model," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3507–3517, July 2017.

[21] R. S. Stanković and B. J. Falkowski, "The Haar Wavelet Transform: its Status and Achievements," *Computers & Electrical Engineering*, vol. 29, no. 1, pp. 25–44, 2003.

[22] G. Sandri, R. De Queiroz, and P. A. Chou, "Compression of Plenoptic Point Clouds Using the Region-Adaptive Hierarchical Transform," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct 2018, pp. 1153–1157.

[23] S. Gu, J. Hou, H. Zeng, H. Yuan, and K. Ma, "3D Point Cloud Attribute Compression Using Geometry-Guided Sparse Representation," *IEEE Transactions on Image Processing*, vol. 29, pp. 796–808, 2020.

[24] R. A. Cohen, D. Tian, and A. Vetro, "Point Cloud Attribute Compression Using 3-D Intra Prediction and Shape-Adaptive Transforms," in *2016 Data Compression Conference (DCC)*, March 2016, pp. 141–150.

[25] 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 *Proceedings of the 26th ACM International Conference on Multimedia*, ser. MM '18. New York, NY, USA: ACM, 2018, pp. 1199–1207.

[26] K. Mammou, A. Tourapis, J. Kim, F. Robinet, V. Valentin, and Y. Su, "Lifting Scheme for Lossy Attribute Encoding in TMC1," Document ISO/IEC JTC1/SC29/WG11 m42640, San Diego, CA, US, Apr. 2018.

[27] 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, Jan. 2019.

[28] K. Mammou, A. Tourapis, and J. Kim, "[G-PCC][New Proposal] Efficient Low-Complexity LOD Generation," ISO/IEC JTC1/SC29/WG11 m46188, Marrakech, MA, Jan. 2019.

[29] R. Mekuria, K. Blom, and P. Cesar, "Design, Implementation, and Evaluation of a Point Cloud Codec for Tele-Immersive Video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 4, pp. 828–842, April 2017.

[30] 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.

[31] 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.

[32] S. Schwarz, M. M. Hannuksela, V. Fakour-Sevom, N. Sheiki-Pour, V. Malamalvadakital, and A. Aminlou, "Nokias Response to CfP for Point Cloud Compression (Category 2)," Document ISO/IEC JTC1/SC29/WG11 m41779, Macau, China, Oct. 2017.

[33] 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 *2019 IEEE International Conference on Image Processing (ICIP)*, Sep. 2019, pp. 3726–3730.

[34] X. Zhang, P. A. Chou, M. Sun, M. Tang, S. Wang, S. Ma, and W. Gao, "Surface Light Field Compression Using a Point Cloud Codec," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 9, no. 1, pp. 163–176, March 2019.

[35] D. Zhang, V. Zakharchenko, and J. Chen, "TMC2 Improved Temporally Consistent Patch Packing (Temporally Consistent Multiframe Patch Packing - TCMPP)," m42718, San Diego, USA, Apr. 2018.

[36] D. Graziosi, "[V-PCC] TMC2 Optimal Texture Packing," Document ISO/IEC JTC1/SC29/WG11 m43681, Ljubljana, SI, Jul. 2018.

[37] ——, "V-PCC New Proposal (related to CE2.12): Harmonic Background Filling," Document ISO/IEC JTC1/SC29/WG11 m46212, Marrakesh, MA, Jan. 2019.

[38] 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, SI, Jul. 2018.
[39] L. Li, Z. Li, B. Li, D. Liu, and H. Li, "Pseudo Sequence Based 2-D Hierarchical Coding Structure for Light-Field Image Compression," in *2017 Data Compression Conference (DCC)*, April 2017, pp. 131–140.
[40] L. Li, Z. Li, S. Liu, and H. Li, "Occupancy-Map-Based Rate Distortion Optimization for Video-Based Point Cloud Compression," in *2019 IEEE International Conference on Image Processing (ICIP)*, Sep. 2019, pp. 3167–3171.
[41] Point Cloud Compression Category 2 reference software, TMC2-7.0. [Online]. Available: http://mpegx.intevry.fr/software/MPEG/PCC/TM/mpeg-pcc-tmc2.git
[42] Multiview High Efficiency Video Coding test model, HTM-16.3. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-16.3/
[43] 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.
[44] G. Bjontegaard, "Calculation of Average PSNR Differences between RD-Curves," Document VCEG-M33, Austin, Texas, USA, April 2001.