Convolutional Neural Network-based Occupancy Map Accuracy Improvement for Video-based Point Cloud Compression

Wei Jia, Li Li, *Member, IEEE*, Anique Akhtar, Zhu Li, *Senior Member, IEEE*, Shan Liu, *Senior Member, IEEE*,

***Abstract*—In video-based point cloud compression (V-PCC), a dynamic point cloud is projected onto geometry and attribute videos patch by patch for compression. In addition to the geometry and attribute videos, an occupancy map video is compressed into a V-PCC bitstream to indicate whether a two-dimensional (2D) point in the projected geometry video corresponds to any point in three-dimensional (3D) space. The occupancy map video is usually downsampled before compression to obtain a tradeoff between the bitrate and the reconstructed point cloud quality. Due to the accuracy loss in the downsampling process, some noisy points are generated, which leads to severe objective and subjective quality degradation of the reconstructed point cloud. To improve the quality of the reconstructed point cloud, we propose using a convolutional neural network (CNN) to improve the accuracy of the occupancy map video. We mainly make the following contributions. First, we improve the accuracy of the occupancy map video by formulating the problem as a binary segmentation problem since the pixel values of the occupancy map video are either 0 or 1. Second, in addition to the downsampled occupancy map video, we introduce a reconstructed geometry video as the other input of the CNN to provide more useful information in order to indicate the occupancy map video. To the best of our knowledge, this is the first learning-based work to improve the performance of V-PCC. Compared to state-of-the-art schemes, our proposed CNN-based approach achieves much more accurate occupancy map videos and significant bitrate savings.**

***Index Terms*—Convolutional Neural Network, High Efficiency Video Coding, Occupancy Map, Segmentation, Video-based Point Cloud Compression.**

I. Introduction

Three-dimensional (3D) industry- and consumer-level scanning equipment, such as RGBD cameras [1], [2] and light detection and ranging (LIDAR) [2], [3], are becoming more common and less expensive than ever before. These sensing devices are capable of scanning and producing a massive amount of 3D data. Due to their ability to represent 3D data in a more immersive and realistic pattern, 3D visual representation approaches such as polygon meshes, light fields, and point clouds are becoming increasingly popular. Among these 3D volumetric digital representation formats, point clouds achieve a good tradeoff among ease of acquisition, realistic rendering, and facilitating data manipulation and processing. Therefore, point clouds are being adopted more frequently.

Point clouds lay a solid foundation for unprecedented visual technologies, including immersive virtual reality (VR), augmented reality (AR), and mixed reality (MR) [4]. These advanced technologies are useful in many applications [5], [6], including historic site [7] and art museum exploration, immersive real-time remote telecommunications [8], interactive games [9], and mobile navigation [10] [11]. However, point clouds are typically represented by an extremely large amount of data. Consequently, it is impossible to cache, stream, and render these large amounts of raw point cloud data. This barrier has created the necessity for efficient point cloud compression (PCC).

Recently, the Moving Pictures Experts Group (MPEG) initiated a standardization activity [12] on PCC. The diversity of point clouds in terms of density has led to the development of two technologies: video-based point cloud compression (V-PCC) [12] and geometry-based point cloud compression (G-PCC) [12]. In this paper, we mainly focus on some improvements based on V-PCC. In V-PCC, a point cloud is initially segmented into 3D patches. Then, these 3D patches are projected onto two-dimensional (2D) planes and packed into geometry and attribute videos. Afterwards, the empty space in the geometry and attribute videos is padded to keep the spatial continuity to improve the video compression efficiency. Finally, the geometry and attribute videos are compressed with high-efficiency video coding (HEVC) [13].

Due to the padding process and the loss caused by compression, it is difficult to determine whether one pixel in the reconstructed geometry video corresponds to a valid 3D point. To address this problem, in addition to the geometry and attribute videos, an occupancy map video is compressed into the V-PCC bitstream. The pixels in the occupancy map video are used to indicate whether the pixels in the geometry and attribute videos correspond to any points in 3D space. Ideally, the occupancy map video should be coded with the same resolution as the geometry and attribute videos (full-resolution), but this incurs a high bitrate cost. To save bitrates, the V-PCC encoder downscales the full-resolution occupancy map video to a half-resolution or quarter-resolution video before compression. The V-PCC decoder then upscales this downscaled occupancy map video back to the full resolution for reconstructing the 3D point cloud. Some noisy pixels are

W. Jia, Anique Akhtar and Z. Li are with the Department of Computer Science and Electrical Engineering, University of Missouri-Kansas City, MO 64110, USA. Professor Zhu Li is the corresponding author (e-mail: wj3wr@umsystem.edu; aniqueakhtar@mail.umkc.edu; zhu.li@ieee.org).

L. Li is with the Department of Electronic Engineering and Information Science, University of Science and Technology of China. (email: lil1@ustc.edu.cn).

S. Liu is with the Tencent Media Lab, 2747 Park Blvd, Palo Alto, CA 94306, USA. (email: shanl@tencent.com).

thus introduced in the boundary areas of the upsampled full-resolution occupancy map video. These 2D noisy pixels are reconstructed into 3D noisy points, which leads to serious quality degradation of the reconstructed point cloud.

Through V-PCC standardization, a variety of occupancy map refinement methods [14]–[22] have been proposed to improve the occupancy map accuracy. Among them, the methods in [21] and [22] have been adopted in the V-PCC encoder, although they are disabled by default. In [21], a patch border filter (PBF) was proposed to manipulate occupancy map and geometry videos to reduce the distance between the contours of patches. However, this method may still introduce some error pixels on the contours. In [22], an occupancy refinement (OR) method is proposed to iteratively refine the occupancy flags of blocks with fewer pixels to avoid introducing noisy ones in the occupancy map. However, this method can still insert some noisy pixels. These deficiencies all lead to degradations in the quality of 3D point cloud reconstructions. Therefore, these two methods have not been adopted as part of the V-PCC common test condition (CTC) [23], and there is still considerable space to develop a better method to improve the accuracy of occupancy map videos.

In this paper, we propose an occupancy-geometry-based convolutional neural network (OGCNN) to improve the accuracy of occupancy map videos in order to improve the quality of reconstructed 3D point clouds. To the best of our knowledge, this work is the first CNN-based solution for improving the efficiency of V-PCC. This work mainly makes the following technical contributions.

- We formulate the problem of occupancy map accuracy improvement as a binary segmentation problem. The binary cross entropy loss is adopted as the loss function to train the CNN.
- A reconstructed geometry video is introduced as the other input of the proposed CNN in addition to an occupancy map video. The geometry contains useful information that can help improve the accuracy of the occupancy map video.
- The proposed algorithm is implemented in the V-PCC reference software. Extensive experiments have been conducted to compare the algorithm in this paper with state-of-the-art (SOTA) algorithms to demonstrate the effectiveness of the proposed scheme.

We organize the remainder of this paper as follows. We review the related works on point cloud compression in Section II, followed by our motivation and observations on occupancy map video enhancement in Section III. We introduce the proposed CNN-based occupancy map accuracy improvement method in Section IV. In Section V, we comprehensively report and analyze the experimental results. A summary of this paper is presented in Section VI.

II. Related Work

This section briefly reviews the prior works on dynamic point cloud compression and accuracy improvements based on occupancy map videos in V-PCC.

A. Dynamic point cloud compression

There are roughly two types of compression methods, 3D-based approaches and 2D-based approaches, for dynamic point cloud compression. As indicated by its name, a 3D-based approach directly performs 3D motion estimation and motion compensation in 3D space. Kammerl *et al.* [24] proposed a lossy compression method for dynamic point cloud streaming that uses the colocated octree node of the reference frame to predict that of the current frame. This method, however, can only be applied to frames with small motions. Thanou *et al.* [25] formulated 3D motion estimation as a feature-matching problem between successive graphs after representing the time-varying geometry of these point cloud frames with a set of graphs. Nonetheless, the motion vectors of some objects in point cloud frames are not accurately estimated. Queiroz *et al.* [26] developed a simple coder that breaks the voxelized point cloud at each frame into blocks of voxels. The 3D translational motion estimation was performed block by block to find the corresponding block of the reference frame. In addition, Mekuria *et al.* [27] further introduced the iterative closest point (ICP) instead of a translational motion model to better characterize the motions in neighboring frames. These schemes can attenuate the deficiencies of 3D motion estimation and motion compensation to some extent. Nevertheless, without flexible block partitioning and more efficient motion estimation algorithms, the compression performance of dynamic point clouds is still incomparable with that of 2D-based methods.

The 2D-based methods that are dedicated to converting a 3D dynamic point cloud to 2D videos for compression through 2D video coding standards have been proven to be efficient. Budagavi *et al.* [28] proposed compressing projected 2D videos derived by sorting points in a 3D point cloud with HEVC. However, this work cannot exploit the mature interprediction, as the generated videos do not have high spatial and temporal correlations. To alleviate this drawback, He *et al.* [29] employed the cubic projection method to convert a 3D dynamic point cloud to 2D videos. Although this work promotes video coding performance, this algorithm leads to the loss of many points due to occlusion. Lasserre *et al.* [30] proposed combining an octree and a projection to decrease the number of occluded points. Mammou *et al.* [31] considered projecting a 3D dynamic point cloud onto 2D videos with a patch-based algorithm. Compared to other proposals, the patch-based algorithm [32] shows better compression efficiency. The MPEG immersive (MPEG-I) media working group adopts a patch-based algotithm as the base of the V-PCC standard. In addition, Li *et al.* [33] proposed using occupancy-map-based rate-distortion optimization and partitioning to improve the performance of V-PCC. Although V-PCC has been proven to be efficient due to its astonishing performance, the downsampled occupancy map video, which intrinsically guides the reconstruction of the geometry and texture information, leads to severe objective and subjective quality degradation of the reconstructed point cloud.

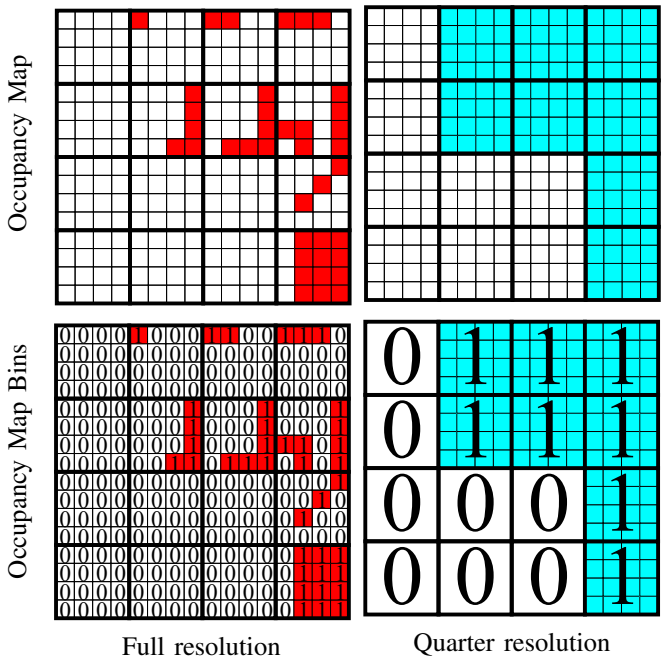


Fig. 1. Occupancy map comparison of the full resolution and quarter resolution with the same occupancy distribution. A grid represents a pixel in the occupancy map video. The bold border square is denoted as a 4×4 block. The red pixels indicate the occupied pixels in the full-resolution occupancy map video, while the blue pixels indicate the occupied pixels in the reconstructed full-resolution occupancy map video.

B. Recent advances in occupancy map video improvement

Through the V-PCC standardization process, many occupancy map refinement methods were proposed to improve occupancy map accuracy. Vosoughi *et al.* [14] proposed a scalable locally adaptive erosion filter that first classified the current pixel of the full-resolution decoded occupancy map into a set of intuitively well-defined classes. Then, different erosion patterns were applied to various classes in the neighborhood of the current pixel. Due to the coarse occupancy resolution, some noisy points are added to the reconstructed point cloud. Oh *et al.* [15] proposed a combination of upsampling and 2D filtering to remove the added points in the occupancy map video. To smooth the jaggy patch boundaries and reduce redundant points, Lee *et al.* [16] proposed an occupancy map refinement method with corner-based boundary estimation. This work primarily addressed the oblique lines. Cai *et al.* [17] proposed an adaptive occupancy map upsampling method for reconstructing a high-resolution occupancy map video. However, there is no guarantee that it can be as close as possible to the original full-resolution occupancy map video. Najaf-Zadeh *et al.* [18] proposed signaling a ternary occupancy map to the decoder if a boundary block in the occupancy map is allowed to be trimmed. Wang *et al.* [20] proposed shifting the position of the occupancy map bounding box during patch generation. However, it can only partially reduce the number of noisy points. These methods can partially solve the problem of inaccurate occupancy map videos. However, none of them are significant enough to be adopted by V-PCC.

There are some methods adopted by the V-PCC encoder during the V-PCC standardization process. Andrivon *et al.* [21] proposed a patch border filtering (PBF) method to manipulate the occupancy map and geometry videos to reduce the distance between contours of patches. However, this method can still insert some noisy pixels on the contours. Guede *et al.* [22] proposed a method to iteratively refine an occupancy map video. This method is proposed to modify the occupancy flags of the blocks with fewer pixels to avoid inserting error flags in an occupancy map video. However, this method may still introduce some noisy points while removing some real points. As a result, they are disabled in the V-PCC encoder by default and are not part of the V-PCC common test condition. Therefore, there is still considerable space to devise a better occupancy map video accuracy improvement method to boost the dynamic point cloud compression efficiency.

III. Motivation

In this section, we first give a clear definition of the resolution of an occupancy map video. Then, the influences of the occupancy map resolution on distortions and bitrates are introduced in detail.

A. Occupancy map video resolution

Ideally, an occupancy map video should be coded at full resolution to indicate exactly whether pixels in the geometry and attribute videos correspond to any points. Nevertheless, a full-resolution occupancy map would cost too many bits. To save bit cost, the V-PCC downscales the full-resolution occupancy map video by P times. Correspondingly, a $P \times P$ block b_p of the full-resolution occupancy map, consisting of P^2 pixels, is downscaled to a single pixel s_p in the downsampled video. When P equals 2 and 4, the downscaled video is called a half-resolution and quarter-resolution occupancy map video, respectively. The V-PCC then upscales the downsampled occupancy map video back to a full-resolution video. Correspondingly, s_p is upscaled to a $P \times P$ block $b_p^{'}$. The reconstructed full-resolution occupancy map video is finally used for reconstructing the geometry and attributes.

In the following, to further analyze the influence of the occupancy map video on the reconstructed quality of the geometry and attributes, we call the pixels indicating that there are corresponding points in 3D space occupied pixels, while we name the pixels indicating that there are no corresponding points in 3D space unoccupied pixels. Suppose a $P \times P$ block b_p in the original full-resolution occupancy map video includes X_o occupied pixels. If X_o is less than P^2, then b_p is partially occupied. Even though b_p is partially occupied, V-PCC marks its corresponding s_p as occupied to avoid losing points. The occupied pixel s_p indicates that all P^2 pixels in the reconstructed full-resolution occupancy map video are occupied. The downsampling and upsampling processes would increase $P^2 - X_o$ occupied pixels. Fig. 1 gives a typical example to compare the full-resolution occupancy map video with the quarter-resolution occupancy map video. The red pixels indicate the occupied pixels in the full-resolution occupancy map video, while the blue pixels indicate the occupied pixels in the reconstructed full-resolution occupancy map video. In

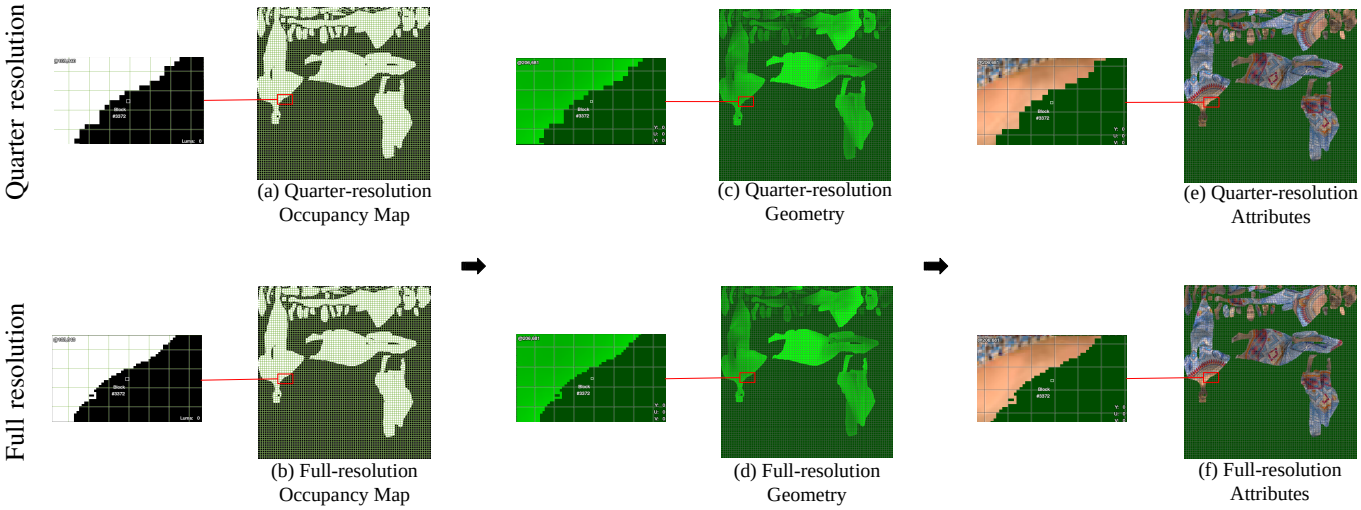


Fig. 2. Comparison of the occupancy maps, geometry, and attributes of the quarter-resolution and full-resolution videos. The reconstructions of the occupancy maps, geometry, and attributes in the first and second rows are derived from the configurations of the quarter-resolution and full-resolution videos, respectively. We can see from the enlarged areas that, compared to the full resolution, the edges of the body in the quarter-resolution occupancy map, geometry and attribute videos show a more serious zigzag artifact.

TABLE I
V-PCC ANCHOR [34] PERFORMANCE COMPARISON OF THE QUARTER-RESOLUTION, HALF-RESOLUTION, AND FULL-RESOLUTION OCCUPANCY MAPS WITHIN THE FIRST 32 FRAMES

Class	Sequence	Quarter vs. Full Geom.BD-GeomRate		Quarter vs. Half Geom.BD-GeomRate	
		D1	D2	D1	D2
A	Loot	-50.3%	-45.5%	-21.9%	-14.9%
	Redandblack	-52.9%	-43.3%	-23.8%	-11.8%
	Soldier	-52.1%	-44.2%	-23.8%	-13.9%
	Queen	-61.1%	-52.7%	-25.4%	-15.3%
B	Longdress	-50.1%	-41.1%	-23.2%	-12.8%
	Class A	-54.1%	-46.4%	-23.7%	-14.0%
	Class B	-50.1%	-41.1%	-23.2%	-12.8%
Avg.	All	-53.3%	-45.3%	-23.6%	-13.8%

an extreme case, in the original full-resolution occupancy map, as shown in the top left subfigure of Fig. 1, only one pixel is occupied in a 4×4 block. However, in the quarter-resolution occupancy map, as shown in the top right subfigure of Fig. 1, all the pixels in the corresponding 4×4 block are considered occupied. In this way, 15 noisy pixels are generated in the restored full-resolution occupancy map.

B. The impact of the occupancy map resolution on the quality of the geometry and attributes

An increase in the number of noisy pixels in the full-resolution occupancy map video can lead to noisy pixels in the geometry and attribute videos. As illustrated in Fig. 2, the reconstructions of the occupancy maps, geometry, and attributes in the first and second rows are derived from the configurations of the quarter resolution and full resolution, respectively. We can see from the enlarged areas of the occupancy map videos ((a) and (b)) that the edge of the body shows a more severe block artifact in the quarter-resolution case than in the full-resolution case. Compared with the quarter resolution, the full resolution provides more accurate representations of the occupancy map. Moreover, the impact of the occupancy accuracy can be propagated into the geometry and attributes. We can see from the enlarged areas of the geometry ((c) and (d)) and attributes ((e) and (f)) that the block distortions are more severe in the quarter-resolution case than in the full-resolution case.

C. The impact of the occupancy map on the bitrates

As mentioned in Section III-A, an occupancy map with higher resolution may lead to smaller distortions. However, it also brings a much higher bitrate cost. According to our observations, the bit cost of the full-resolution occupancy map is approximately four times greater than that of the quarter-resolution map. As shown in Table I, we compare the BD-rates [35] of the point-to-point error (D1) and point-to-plane error (D2) [36] among the quarter-resolution, half-resolution and full-resolution occupancy maps in the V-PCC anchor version 11 [34]. Compared to the full-resolution occupancy map, the quarter-resolution occupancy map achieves a -53.3% BD-rate savings on D1 and a -45.3% BD-rate savings on D2. Compared to the half-resolution occupancy map, the quarter-resolution occupancy map achieves a -23.6% and -13.8% BD-rate savings on D1 and D2, respectively. The main reason for these results is that, compared to the half-resolution and full-resolution occupancy map videos, the quarter-resolution videos are downscaled two and four times, respectively; hence, they cost much fewer bits.

IV. PROPOSED ALGORITHM

In this section, we introduce the proposed OGCNN scheme in detail, including a detailed discussion on the design of the OGCNN, loss function, dataset, and training process.

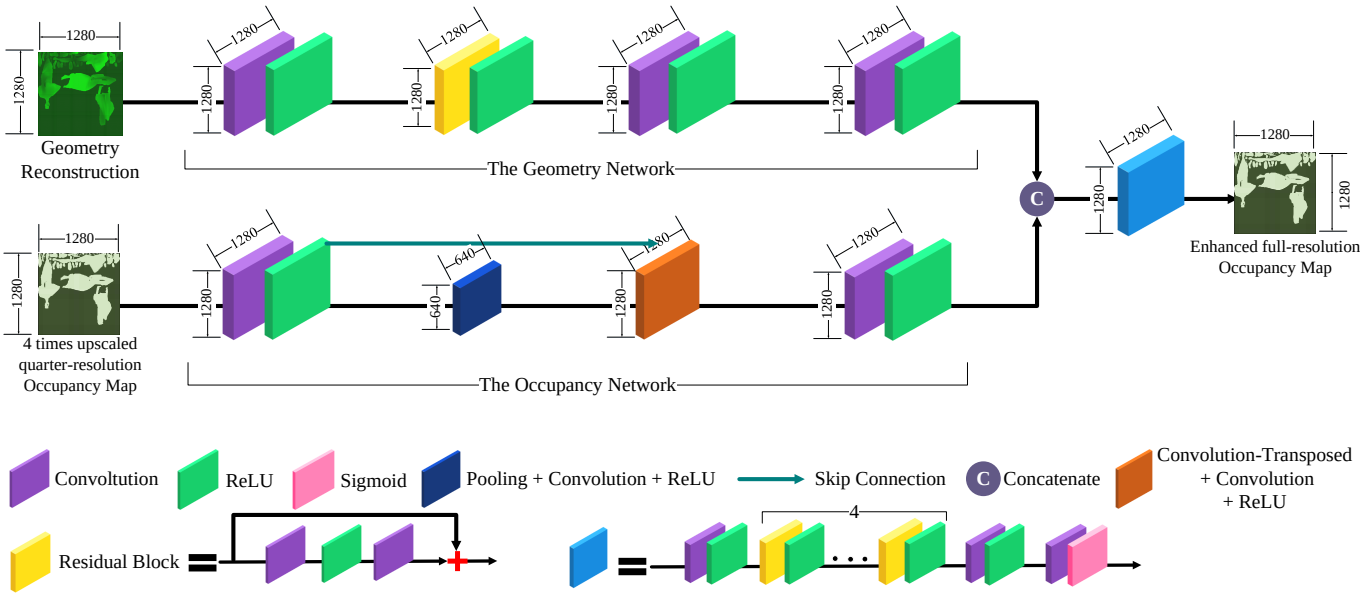


Fig. 3. The proposed OGCNN framework includes two subnetworks: the Occupancy Network and the Geometry Network. The Occupancy Network uses the quarter-resolution occupancy map video as input. It derives occupancy segmentation feature maps from the occupancy map. The Geometry Network uses the reconstructed geometry video as input and derives geometry segmentation feature maps from the geometry. The occupancy map and geometry segmentation features are then concatenated together and used as the input of the remaining convolutional layers.

A. Architecture of the proposed OGCNN

As shown in Table I, the quarter-resolution occupancy map video leads to a better performance compared with the half-resolution and full-resolution occupancy map videos. However, we also know that the higher the occupancy map video resolution is, the better the quality of the reconstructed geometry and attributes. Therefore, we use the quarter-resolution occupancy map video as the base and try to design an algorithm to improve its accuracy and to improve the reconstructed point cloud geometry and attribute quality. As CNNs have been demonstrated to be powerful in both low-level and high-level vision tasks [37], we propose using a CNN to make the accuracy of the quarter-resolution occupancy map as close as possible to that of a higher-resolution target. The higher-precision target can be the full-resolution or half-resolution occupancy map.

When we design the proposed architecture, we mainly consider the following two aspects to optimize its performance. First, as the occupancy map is a particular type of video that incorporates only binary values, we formulate the problem of improving the occupancy map precision as a binary segmentation problem. In other words, we try to devise a segmentation CNN that can discriminate the occupied (value 1) and unoccupied (value 0) statuses per pixel in the inputted occupancy map. Second, in addition to the quarter-precision occupancy map video, geometry reconstruction is introduced as the other input to provide the network with more useful information. In the V-PCC encoder, as the geometry values of the unoccupied pixels are padded from their neighbors [38], they can better reflect the real occupancy distribution than the binary occupancy map. For example, if the geometry value of a specific position is not the same as that of its neighbors, it is almost impossible for it to be an unoccupied pixel. However, we cannot obtain this information from the quarter-resolution occupancy map itself. Therefore, we consider the geometry reconstruction to be an important supplement to the quarter-resolution occupancy map.

Fig. 3 shows the overall architecture of the proposed OGCNN scheme with both the quarter-resolution occupancy map video and reconstructed geometry video as inputs. The scheme consists of two subnetworks: the Occupancy Network and the Geometry Network. The Occupancy Network uses the quarter-resolution occupancy map video as input. It derives occupancy segmentation feature maps from the occupancy map. The Geometry Network uses the reconstructed geometry video as input and derives geometry segmentation feature maps from the geometry. The occupancy map and geometry segmentation features are then concatenated together and used as the input of the remaining convolutional layers.

In addition to the dual inputs, as shown in Fig. 3, we develop different subnetworks for the quarter-resolution occupancy map video and the reconstructed geometry video. As mentioned above, the characteristics of the occupancy map and geometry reconstruction are different. The occupancy map is binary, while the information in the geometry is more sensitive. We design different subnetworks to optimize the features derived from the occupancy map and geometry. Detailed introductions of the two subnetworks are described in Section IV-B and Section IV-C, respectively.

Algorithm 1 shows the algorithm flow of the proposed OGCNN. We first extract the occupancy map and geometry reconstructions from the bitstream. Then, both of them are fed into the OGCNN to generate the occupancy map video with a higher accuracy. The occupancy map video with a

Algorithm 1 The flow of the OGCNN approach in V-PCC

Input: x is the Occupancy Network input, and the geometry reconstruction z is the Geometry Network input.
Output: The enhanced occupancy map $F_{out}(O(x), G(z))$.

if *Initialization succeeds* **then**
  Input the occupancy map x into the Occupancy Network;
  Extract the geometry reconstruction z as the Geometry Network input;
  Compute the Occupancy Network segmentation features $O(x)$;
  Compute the Geometry network segmentation features $G(z)$;
  Concatenate the segmentation features of $O(x)$ and $G(z)$;
  Obtain the enhanced occupancy map $F_{out}(O(x), G(z))$;
end

TABLE II
OCCUPANCY NETWORK PARAMETERS OF THE CONV AND TRANSPOSED CONV LAYERS

Layer	Conv1	Conv2	Transposed Conv1	Conv3	Conv4
Kernel Size	3×3	3×3	2×2	3×3	3×3
Feature Map Number	4	8	4	4	4
Stride	1	1	2	1	1
Padding	1	1	0	1	1

higher accuracy is finally used in loop for reconstructing the geometry, attributes, and point cloud.

B. Design of the Occupancy Network

The Occupancy Network uses the unsampled quarter-resolution occupancy map video as the input. It adopts the classic autoencoder architecture [39] [40] with a skip connection concatenating the encoder and decoder [41]. The Occupancy Network contains a downsampling and upsampling pair to segment the occupancy map. In this way, the Occupancy Network can collect the global information as much as possible.

The lower branch of Fig. 3 shows the detailed architecture of our proposed Occupancy Network. We adopt the max pooling plus convolutional layer and transposed convolutional layer [42] to perform downsampling and upsampling, respectively. At the encoder, downsampling reduces the occupancy map redundancy and keeps the most distinctive features for segmentation. At the decoder, upsampling increases the spatial resolution of the features to the target resolution for accurate segmentation. However, the downsampling-upsampling process may lead to a loss of global information. To provide accurate global information for segmentation, a skip connection, which concatenates the features in the encoder and decoder, is added to the network structure.

Table II shows the detailed configurations of the Occupancy Network. For the convolutional layers, we set the kernel size to 3×3, the stride to 1, the padding size to 1, and the feature map number to 4 or 8. For the transposed convolutional layers, we set the kernel size to 2×2, the stride to 2, the padding size to 0, and the feature map number to 4. We use the rectified linear unit (ReLU) as the activation function.

C. Design of the Geometry Network

As analyzed in Section IV-A above, we consider the reconstructed geometry video as the other input of the proposed OGCNN to improve the precision of the occupancy map video. Accordingly, we develop a specific Geometry Network to derive distinctive features. In the Geometry Network, the residual block [43] is employed to derive the geometry features for segmentation. The residual block also has the benefit of preventing the vanishing of the gradient.

The upper branch of Fig. 3 describes the detailed structure of the proposed Geometry Network. The Geometry Network includes a residual block and three convolutional layers. Considering the complexity, we only use a total of five convolutional layers to derive the geometry features. For each convolutional layer, we set the kernel size to 3×3, the padding size to 1, the stride to 1, and the feature map number to 4.

D. Loss function

To train our proposed segmentation network effectively, we adopt the binary cross-entropy loss [44] to supervise the training of the proposed OGCNN.

$$L(\Theta) = \frac{1}{N}\sum_{i=1}^{N}(\log \Upsilon\left((O_i, G_i)|\Theta\right) \cdot X_i - \log(1 - \Upsilon\left((O_i, G_i)|\Theta\right)) \cdot (1 - X_i)) \quad (1)$$

where Θ encapsulates the whole parameter set of the OGCNN, including the weights and bias, and $\Upsilon\left(Y_i|\Theta\right)$ denotes the OGCNN module. X_i denotes the labels of a half-resolution or full-resolution occupancy map, where i indexes each label. O_i and G_i are the corresponding dual inputs of the upsampled quarter-resolution occupancy map and the reconstructed geometry video, respectively. N is the number of samples. Under the supervision of the binary cross-entropy loss, the output of the occupancy map video is close to that of the target half-resolution or full-resolution occupancy map video.

E. Dataset and training

Dataset. There are currently no widely used datasets to train the proposed OGCNNN for improving V-PCC. The only dataset we can have access to is the dynamic point cloud dataset provided by $8i$ and defined in the V-PCC CTC [23]. We divide the five dynamic point clouds from $8i$ into training, validating, and testing datasets. More specifically, we use the dynamic point cloud called Soldier for training and validation. We use the other four dynamic point clouds, Loot, Redandblack, Queen, and Longdress, for testing. With Soldier, we first derive 300 frames of the quarter-resolution occupancy map video and reconstructed geometry video, both of which have spatial resolutions of 1280×1280, from the V-PCC reference software. Among them, 224 and 76 frames are used for training and validation, respectively. Then, we generate the same number of full-resolution and half-resolution occupancy

TABLE III
TRAINING PARAMETERS

Parameters	Value
Batch size	16
Total Epochs	60
Base Learning Rate	$1e^{-4}$
γ Adjusting Coefficient	0.1
Adjusting Epoch Intervals	50
Weight Decay	$1e^{-4}$
Momentum	0.9

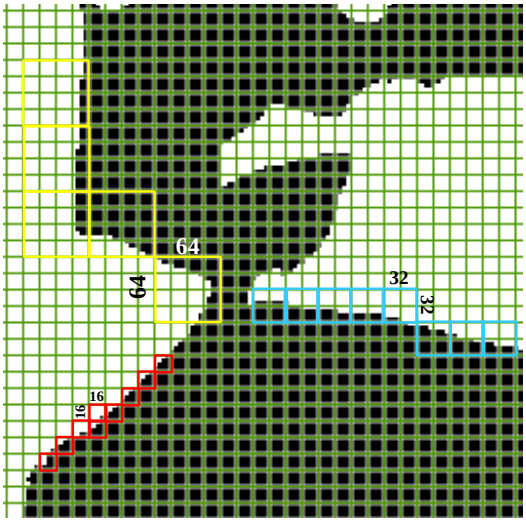


Fig. 4. Occupancy map boundary blocks. A boundary block is an $N \times N$ square that consists of both occupied and unoccupied pixels. The white grids represent the occupied pixels, while the black grids represent the unoccupied pixels. The red, blue, yellow squares indicate the 16×16, 32×32, and 64×64 boundary blocks, respectively.

map videos as labels. Finally, we extract 64×64 blocks from the Luma component of the occupancy map videos and the reconstructed geometry videos and use them for training the proposed OGCNN. In total, there are $89,600$ pairs of 64×64 inputs and labels for training and $30,400$ pairs for validation.

Training. Table III shows the detailed parameters of the training process. The batch size and total number of epochs are set as 16 and 60, respectively. For training, we set the base learning rate to $1e^{-4}$. After 50 epochs, we decrease the learning rate by multiplying by 0.1. We adopt the adaptive moment estimation (Adam) [45] algorithm as the gradient optimizer. The momentum and weight decay are set to 0.9 and $1e^{-4}$, respectively.

V. EXPERIMENTAL RESULTS

A. Experimental settings and metrics

To test the performance of the proposed OGCNN, we implement the proposed OGCNN in version 11 of the V-PCC reference software [34] to compare it with the V-PCC version 11 anchor, PBF [21], and OR [22]. Two OGCNNs are trained depending on whether we use the full-resolution occupancy map video or the half-resolution occupancy map video as the label. The OGCNN trained with the full-resolution occupancy map video as the label is named the Full OGCNN. The OGCNN trained with the half-resolution occupancy map is called the Half OGCNN. We test the performance of the proposed algorithms in both the all intra and random access cases, as defined in the V-PCC CTC [23]. We test the five rate points from a low bitrate $r1$ to a high bitrate $r5$, as defined in the V-PCC CTC [23]. As the dynamic point cloud Soldier is used in the training process, we use the other four dynamic point clouds from $8i$ to show the performance of the proposed OGCNN. To save some encoding time, we only test the first 32 frames of each point cloud, which are a good representation of all frames.

To evaluate the geometry distortions, we use the point-to-point error (D1) and point-to-plane error (D2) as the metrics [23]. Both D1 and D2 are calculated in a symmetrical way with both the original point cloud and reconstructed point cloud as the anchors. The one with a larger distortion is used as the final distortion. O and R denote the original point cloud and its reconstruction. For each point $r \in R$, we identify its corresponding point $o \in O$ by searching the nearest neighbor with a KD-tree in O. Then, D1 $d'_{R,O}$ from R to O is calculated as follows:

$$d'_{R,O} = \frac{1}{N_R} \sum_{\forall r \in R} ||D(r,o)||_2^2 \tag{2}$$

where N_R is the number of points in point cloud R. $D(r,o)$ is the error vector connecting r to o. D1 $d'_{O,R}$ from O to R can be computed in a similar manner.

Similarly, $d''_{R,O}$ denotes D2 from R to O, which is calculated as

$$d''_{R,O} = \frac{1}{N_R} \sum_{\forall r \in R} (D(r,o) \cdot V_r)^2 \tag{3}$$

where V_r is the normal vector on point r. D2 $d''_{O,R}$ from O to R can be computed in a similar manner.

The attribute distortion also employs the symmetric computation method. The attribute distortion [23] $d_{R,O}$ from R to O uses the mean square error (MSE)

$$d_{R,O} = \frac{1}{N_R} \sum_{\forall r \in R} ||y(o) - x(r)||_2^2 \tag{4}$$

where $y(o)$ and $x(r)$ are the attribute values of the original and reconstruction point cloud points, respectively. The attribute distortion $d_{O,R}$ from O to R can be computed in a similar manner.

To better show the performance of the proposed OGCNN in improving occupancy map accuracy, we provide a new quality metric to measure the occupancy map accuracy. As the occupancy map accuracy is meaningful only at the boundary block between the patches and the empty space, we first give a clear definition of the boundary block. As shown in Fig. 4, a boundary block is an $N \times N$ square that consists of both occupied and unoccupied pixels. In Fig. 4, the white grids represent the occupied pixels, while the black grids represent the unoccupied pixels. The red, blue, yellow squares indicate the 16×16, 32×32, and 64×64 boundary blocks, respectively. Then, our proposed occupancy accuracy α_N is defined as

$$\alpha_N = \frac{\sum_{i=1}^{\xi} \Phi_N(i)}{\Psi_N}, \tag{5}$$

TABLE IV
PERFORMANCE COMPARISON OF THE FULL-RESOLUTION OGCNN, HALF-RESOLUTION OGCNN AND QUARTER-RESOLUTION V-PCC [34] UNDER THE ALL INTRA CASE

Class	Sequence	Full OGCNN vs. V-PCC [34]					Half OGCNN vs. V-PCC [34]				
		Geom.BD-TotalRate		Attr.BD-TotalRate			Geom.BD-TotalRate		Attr.BD-TotalRate		
		D1	D2	Luma	Cb	Cr	D1	D2	Luma	Cb	Cr
A	Loot	6.3%	**-16.9**%	2.9%	1.2%	4.6%	−0.2%	**-13.5**%	0.8%	0.4%	2.4%
	Redandblack	16.2%	**-20.4**%	3.4%	−0.2%	2.2%	4.1%	**-15.2**%	1.1%	−0.2%	0.5%
	Queen	18.4%	**-26.1**%	23.9%	24.3%	48.1%	−1.5%	**-18.5**%	5.0%	7.3%	14.4%
B	Longdress	22.0%	**-18.5**%	3.2%	0.9%	2.0%	7.7%	**-14.2**%	0.9%	0.1%	0.6%
	Class A	13.6%	**-21.1**%	10.1%	8.4%	18.3%	0.8%	**-15.7**%	2.3%	2.5%	5.8%
	Class B	22.0%	**-18.5**%	3.2%	0.9%	2.0%	7.7%	**-14.2**%	0.9%	0.1%	0.6%
Avg.	All	15.7%	**-20.5**%	8.4%	6.5%	14.2%	2.5%	**-15.3**%	1.9%	1.9%	4.5%

TABLE V
OCCUPANCY ACCURACY COMPARISON OF THE V-PCC ANCHOR AND OGCNNS ON BOUNDARY BLOCKS

	$N = 16$			$N = 32$			$N = 64$		
	V-PCC Anchor	Half OGCNN	Full OGCNN	V-PCC Anchor	Half OGCNN	Full OGCNN	V-PCC Anchor	Half OGCNN	Full OGCNN
Loot	89.27%	94.61%	96.24%	94.03%	97.00%	97.89%	96.08%	98.03%	98.61%
RedandBlack	88.27%	93.52%	95.38%	92.70%	95.96%	97.10%	94.59%	97.00%	97.84%
Queen	87.35%	92.26%	94.09%	91.86%	95.01%	96.16%	93.58%	96.06%	96.95%
Longdress	88.96%	93.69%	95.06%	93.42%	96.23%	97.00%	95.53%	97.43%	97.95%
Avg. All	88.47%	93.52%	95.19%	93.01%	96.05%	97.04%	94.94%	97.13%	97.84%

where N is the boundary block size and ξ is the total number of boundary blocks. $\Phi_N(i)$ indicates the number of correctly identified pixels in the ith boundary block between the reconstructed occupancy map video and the label. Ψ_N is the total number of pixels in all ξ boundary blocks. As indicated by (5), when we measure the occupancy map accuracy, we restrict the statistical area to the boundary blocks to avoid counting large amounts of successive occupied or unoccupied pixels, as they are identical in the reconstructed and original occupancy map videos. Therefore, our proposed occupancy accuracy measure can better reflect the benefits of the proposed algorithms for improving the occupancy map accuracy.

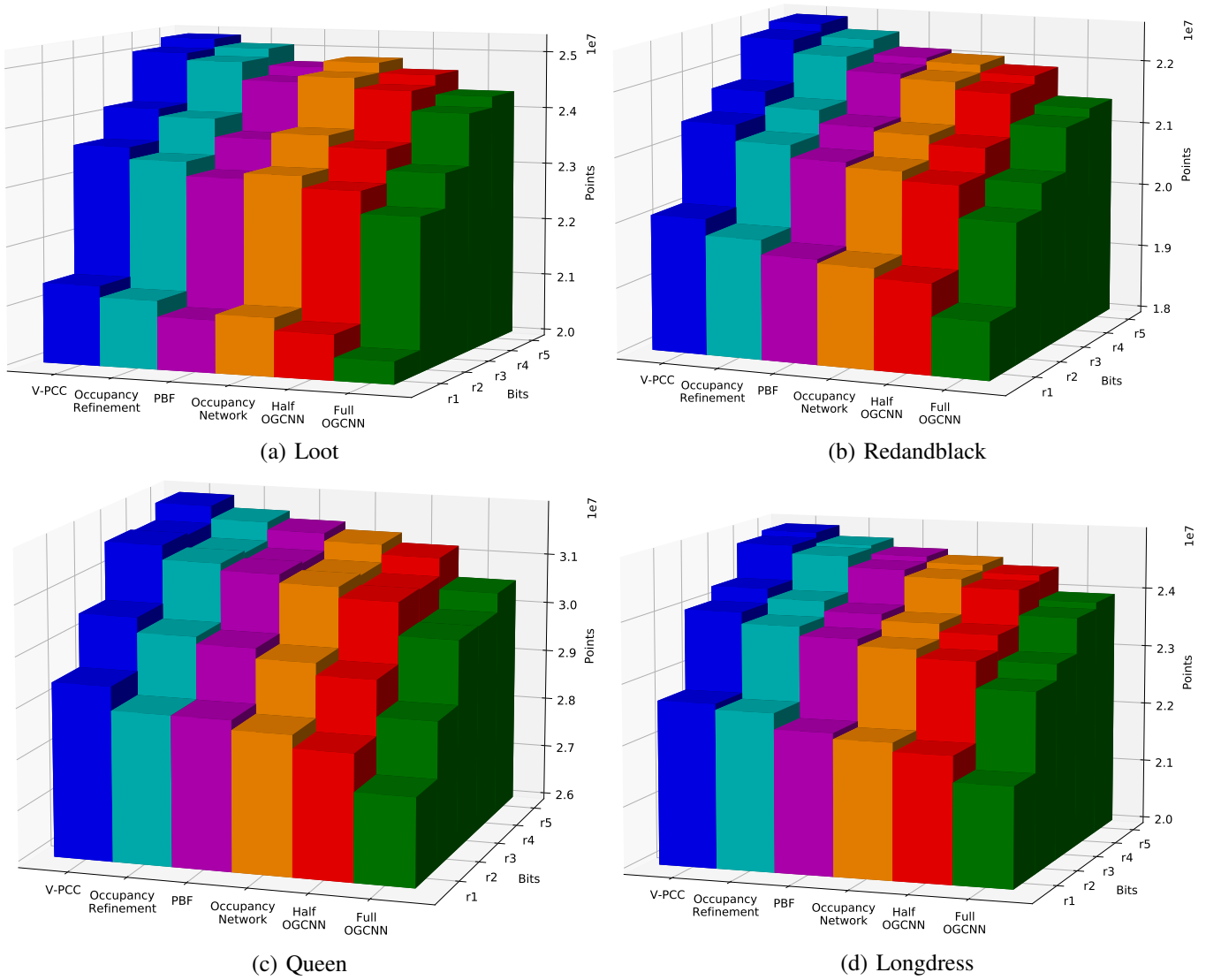
B. Performances of the proposed OGCNN algorithm under the all intra case

Table IV shows the BD-rate comparison results of the proposed OGCNN and the quarter-resolution V-PCC anchor under the all intra case. We can see that the proposed half and Full OGCNNs achieve an average of 15.3% and 20.5% BD-rate savings when D2 is used as the quality metric. The performance improvements are consistent for all tested dynamic point clouds, as the proposed OGCNN achieves over 10% rate-distortion (R-D) performance improvements for all dynamic point clouds. The peak difference reaches 18.5% and 26.1% for the dynamic point cloud Queen. The experimental results demonstrate the effectiveness of the proposed OGCNN.

In addition, we can see from Table IV that both the half and Full OGCNNs lead to some performance losses on the geometry if measured by D1 and the attributes. As stated in Section IV-A, the OGCNN aims to remove some noisy points. Therefore, the numbers of points N_R of the proposed half and Full OGCNNs are less than that of the V-PCC anchor. According to (2) and (4), the smaller the number of points N_R is, the larger D1 and the attribute distortion are since they are the average of all points. That is why the proposed OGCNN suffers some performance losses in terms of geometry if measured by D1 or the attributes. In addition, as explained by [36], D1 has the disadvantage of ignoring the fact that point clouds represent surfaces of objects.

To better show the performance of the proposed OGCNNs for improving occupancy map accuracy, we compare the occupancy accuracies on the boundary blocks between the OGCNNs and the V-PCC anchor in Table V. In Table V, N represents the boundary block size. We test three configurations with N set to 16, 32 and 64 for evaluation. We can see that both the Full OGCNN and Half OGCNN perform much better than the V-PCC anchor. For example, when N equals 16, the Full OGCNN and the Half OGCNN improve the occupancy map accuracy by 6.72% and 5.05% compared with the V-PCC anchor, respectively. These results further demonstrate that the proposed OGCNN can lead to a better occupancy map video than the V-PCC anchor.

To measure the complexities of the proposed algorithm, we use the same environment to test both the V-PCC anchor and the proposed algorithm. More specifically, the CPU configuration is an Intel(R) Core i5-8400 CPU @ 2.80 GHz, and the GPU configuration is a GTX 1080ti. In the all intra case, both the full and Half OGCNNs lead to almost the same encoding time compared with the V-PCC anchor. In addition, the decoding time is increased by 2% on average. The time complexities of the proposed algorithms are similar to that of the the V-PCC anchor.



(a) Loot (b) Redandblack

(c) Queen (d) Longdress

Fig. 5. Comparison of the numbers of points N_R in the reconstructed 3D point clouds of the V-PCC [34], OR [22], PBF [21], Occupancy Network and OGCNN. The Y axis is the bitrate, which gradually increases from low bitrate $r1$ to high bitrate $r5$. We can see that for all dynamic point clouds, the number of points N_R in our proposed Half OGCNN and Full OGCNN are less than those in the V-PCC anchor, SOTAs, and the Occupancy Network.

TABLE VI
PERFORMANCE COMPARISON OF THE HALF OGCNN AND V-PCC ANCHOR [34] UNDER RANDOM ACCESS

Sequence	Geom.BD-TotalRate		Attr.BD-TotalRate		
	D1	D2	Luma	Cb	Cr
A.Loot	−1.1%	**-12.0**%	1.6%	0.5%	5.4%
A.Red&black	4.2%	**-14.1**%	1.1%	−0.1%	0.5%
A.Queen	−0.6%	**-18.8**%	7.0%	9.5%	17.6%
B.Longdress	9.2%	**-14.8**%	1.2%	1.1%	1.6%
Class A	0.8%	**-15.0**%	3.2%	3.3%	7.8%
Class B	9.2%	**-14.8**%	1.2%	1.1%	1.6%
Avg. All	2.9%	**-14.9**%	2.7%	2.7%	6.3%

C. Performance of the proposed OGCNN algorithm under the random access case

In the random access case, as shown in Table VI, we can see that compared to the V-PCC anchor, the proposed Half OGCNN can achieve an average 14.9% R-D performance improvement when D2 is used as the quality metric. The peak difference between the OGCNN and the V-PCC anchor reaches 18.8% on the dynamic point cloud Queen. This result demonstrates that, in addition to the all intra case, the OGCNN can bring significant benefits to the random access case. As explained in Section V-B, compared to the anchor, the Half OGCNN also suffers a few performance losses in terms of the attributes.

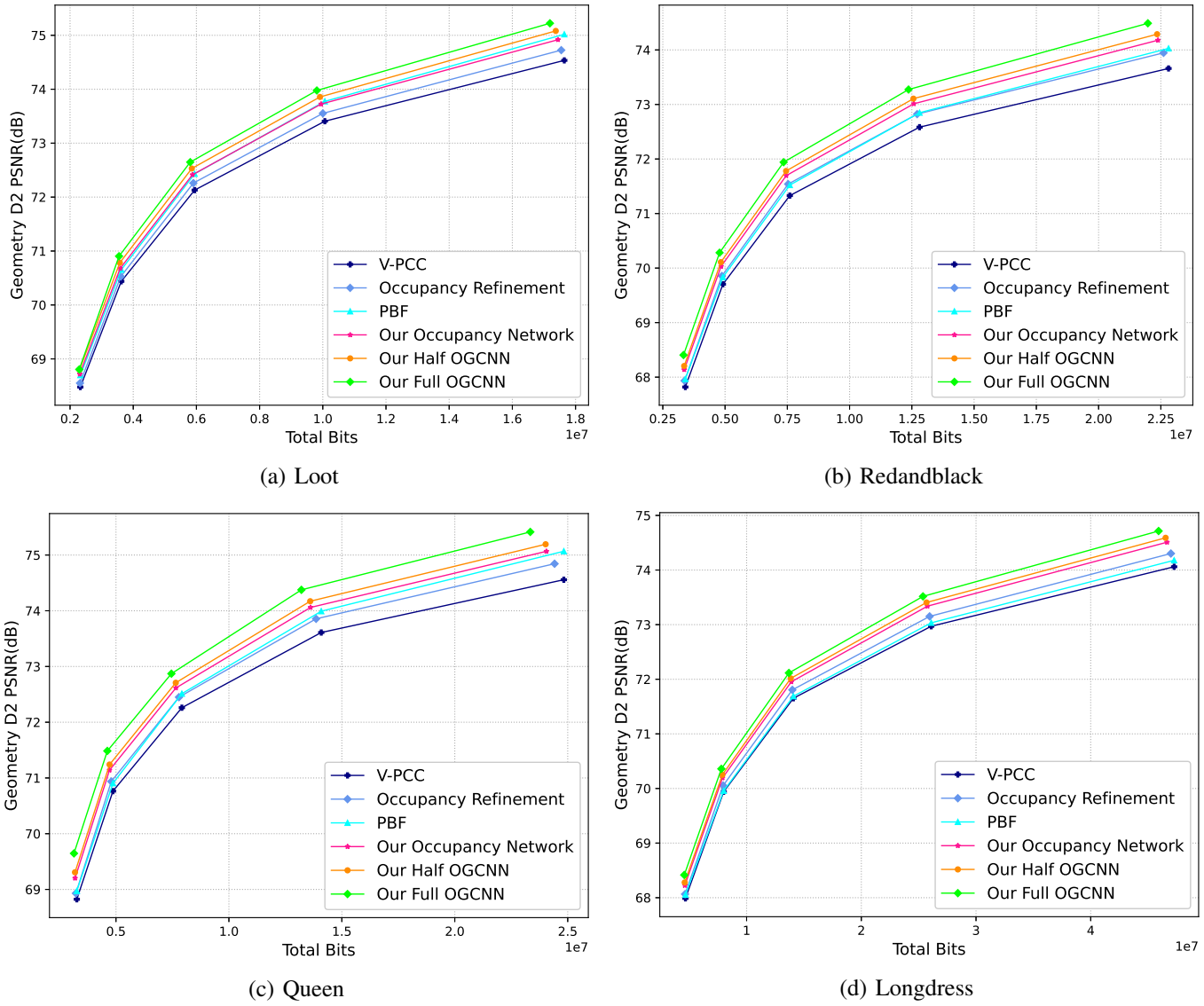


Fig. 6. Geometry R-D curve comparison of the V-PCC [34], OR [22], PBF [21], Occupancy Network and OGCNN for the all intra case. We can see that the D2 PSNRs of the proposed OGCNN at all five rate points are higher than those of the V-PCC anchor, SOTAs and Occupancy Network.

TABLE VII
PERFORMANCE COMPARISON OF THE HALF OGCNN, OR [22], AND PBF [21] UNDER THE ALL INTRA CASE

Sequence	Half OGCNN vs. PBF [21] Geom.BD-TotalRate		Half OGCNN vs. OR [22] Geom.BD-TotalRate	
	D1	D2	D1	D2
A.Loot	0.9%	**-4.5%**	1.2%	**-9.5%**
A.Red&black	3.3%	**-9.5%**	5.7%	**-8.9%**
A.Queen	4.2%	**-10.5%**	6.5%	**-11.3%**
B.Longdress	4.6%	**-12.8%**	7.4%	**-8.8%**
Class A	2.8%	**-8.1%**	4.5%	**-9.9%**
Class B	4.6%	**-12.8%**	7.4%	**-8.8%**
Avg. All	3.3%	**-9.3%**	5.2%	**-9.6%**

D. Comparison of the OGCNN and SOTAs

Table VII shows the BD-rate comparison of the Half OGCNN, PBF [21], and OR [22]. The Half OGCNN performs better than the PBF and OR by an average of 9.3% and 9.6% when D2 is used as the quality metric, respectively. These performance results demonstrate that the proposed OGCNN significantly outperforms the SOTAs. In addition, the time complexities of the proposed algorithms are comparable to that of the SOTAs.

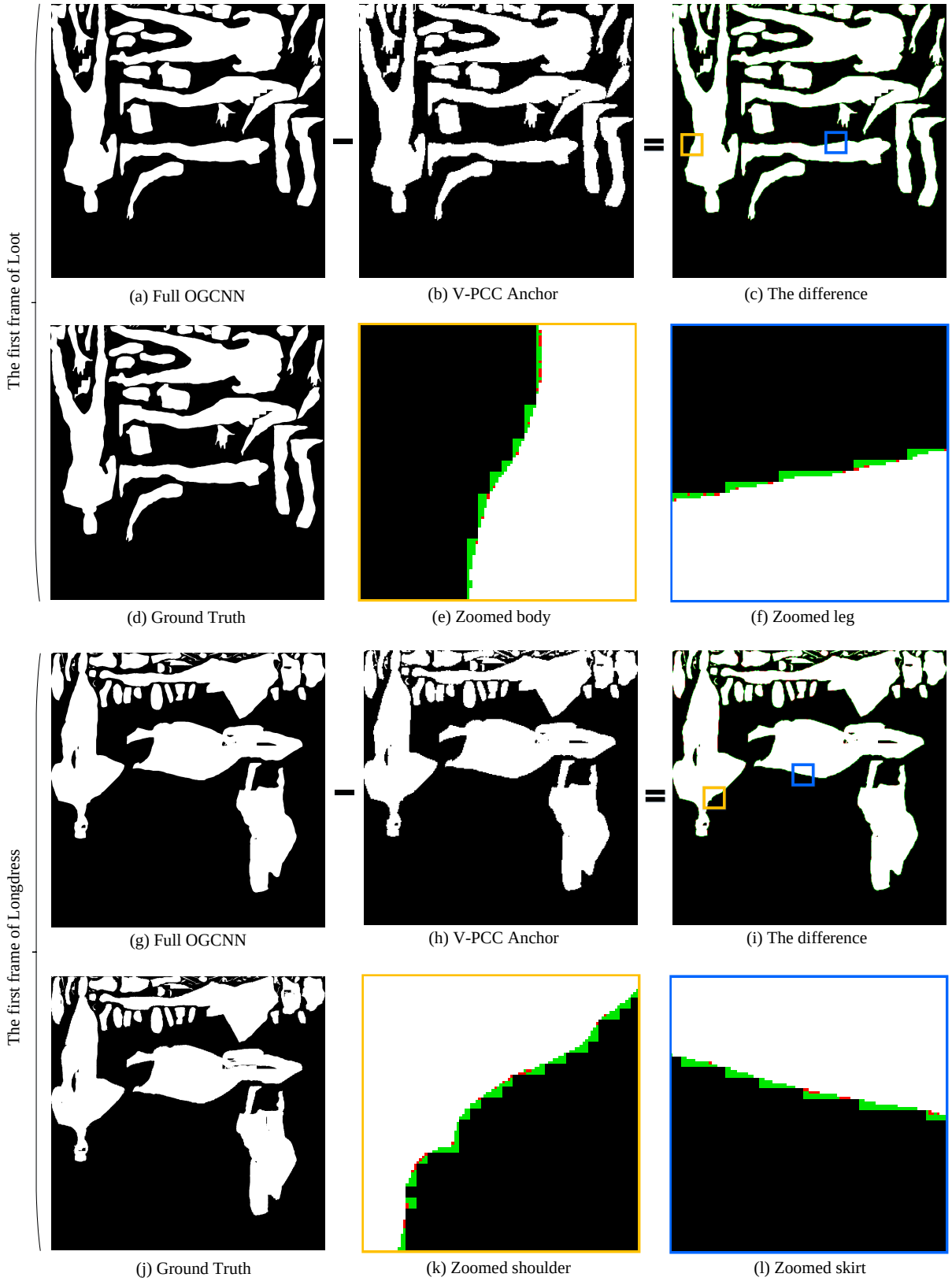


Fig. 7. 2D occupancy map comparison of the ground truth, V-PCC anchor [34] and proposed Full OGCNN. For Loot, (a), (b), (c) and (d) are the occupancy map reconstructions of the Full OGCNN and the V-PCC anchor, the difference between the two, and the ground truth, respectively. (e) and (f) are the enlarged areas of the gold and blue blocks in (c). For Longdress, the same order is followed. In (c) and (i), the green pixels denote the unoccupied pixels of the V-PCC anchor correctly removed by the Full OGCNN. The red pixels denote the occupied pixels of the V-PCC anchor wrongly removed by the Full OGCNN. We can see from (c) and (i) that the number of green pixels is much greater than the number of red pixels. (For a better visual comparison, please zoom in on the subfigures.)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2021.3079698, IEEE Transactions on Multimedia

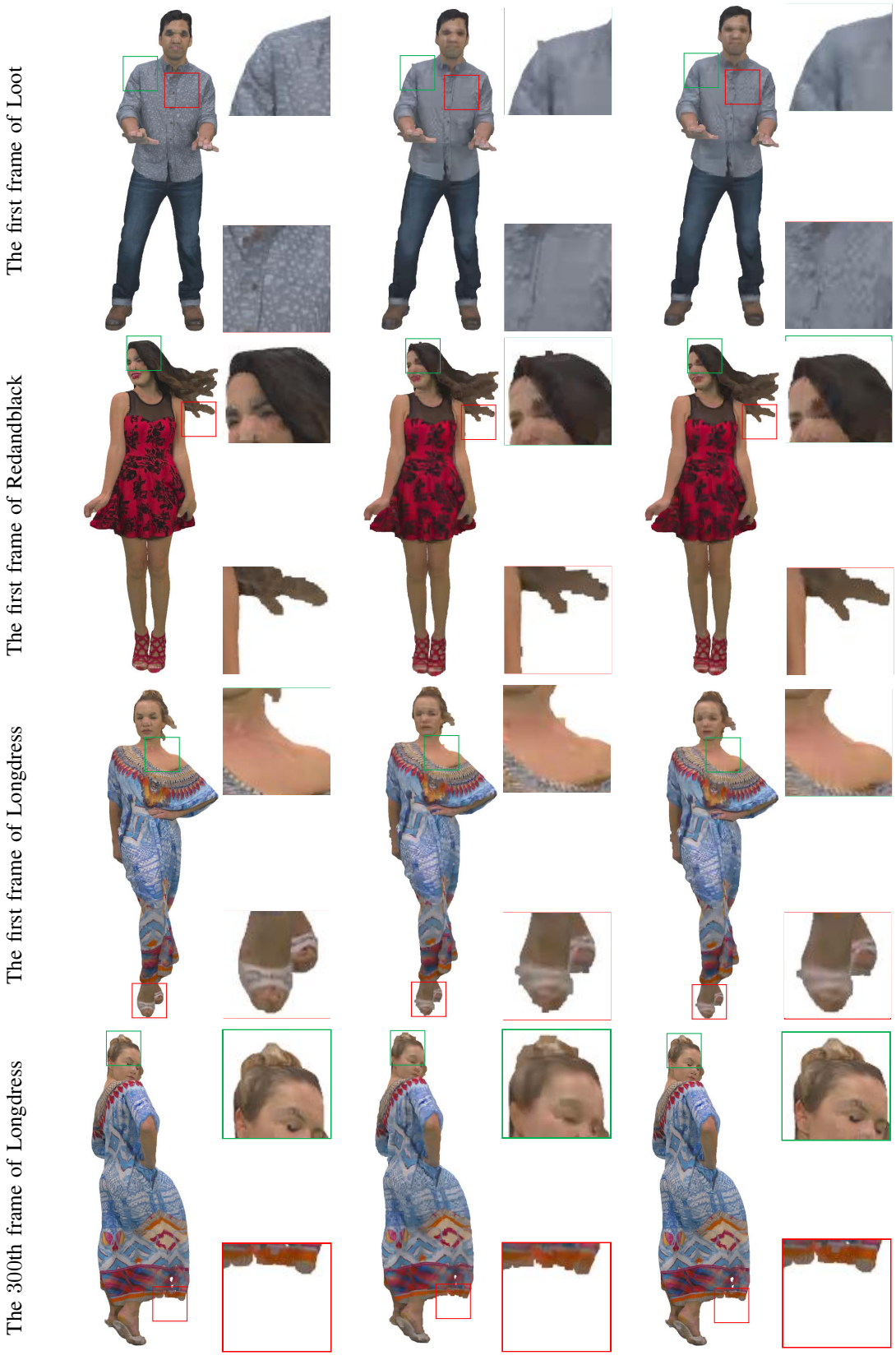


Fig. 8. 3D visual comparison of the original point clouds, the point clouds reconstructed by the V-PCC anchor and the proposed Half OGCNN. The zoomed figures are derived from the first frame of Loot, the first frame of RedandBlack, the first frame of Longdress, and the 300th frame of Longdress. From these frames, we can clearly see from the red and green rectangles that there are many noisy points in the reconstructions of the V-PCC anchor. However, the reconstructions of the Half OGCNN are much smoother and closer to the original point clouds.

E. Ablation analysis of introducing the geometry

TABLE VIII
PERFORMANCE COMPARISON OF THE HALF OGCNN AND OCCUPANCY NETWORK UNDER THE ALL INTRA CASE

Sequence	Geom.BD-TotalRate		Attr.BD-TotalRate		
	D1	D2	Luma	Cb	Cr
A.Loot	1.4%	**-4.1**%	0.4%	0.5%	1.0%
A.Red&black	0.2%	**-2.8**%	0.2%	0.3%	0.2%
A.Queen	1.8%	**-3.6**%	0.8%	3.0%	5.8%
B.Longdress	2.0%	**-2.6**%	0.5%	0.0%	0.2%
Class A	1.1%	**-3.5**%	0.5%	1.2%	2.3%
Class B	2.0%	**-2.6**%	0.5%	0.0%	0.2%
Avg. All	1.4%	**-3.3**%	0.5%	0.9%	1.8%

To evaluate the effect of introducing the geometry as an additional input, we compare the proposed Half OGCNN with the Occupancy Network illustrated in Fig. 3. The Occupancy Network uses only the quarter-resolution occupancy map video as input. Note that to ensure fairness, the Half OGCNN and the Occupancy Network use the same network configurations. Table VIII shows the comparison of the Half OGCNN with and without the Geometry Network. Compared to the Occupancy Network, the Half OGCNN saves an average of 3.3% BD-rate when D2 is used as the quality metric, while suffering a few performance losses of attributes. These performance results demonstrate that the geometry, as an additional input, can lead to clear benefits.

F. Number of points

To further demonstrate that the proposed OGCNN can reduce the number of noisy points, we count the numbers of points N_Rs in the reconstructed 3D point clouds under different algorithms in Fig. 5. The Y axis is the bitrate, which gradually increases from low bitrate $r1$ to high bitrate $r5$. We can see that for all dynamic point clouds, the number of points N_R of our proposed Half OGCNN and Full OGCNN are less than those of the V-PCC anchor, SOTAs, and Occupancy Network. These statistics fully demonstrate that the proposed OGCNN removes noisy points to improve the R-D performance.

G. Rate-Distortion Curves

Fig. 6 shows some representative geometry R-D curves from the all intra case. We can see that the D2 PSNRs of the proposed OGCNN at all five rate points are higher than those of the V-PCC anchor, SOTAs and Occupancy Network. These experimental results demonstrate that the proposed OGCNN is significantly superior to the V-PCC anchor, SOTAs, and Occupancy Network.

H. Visual results of the 2D occupancy maps

Fig. 7 shows the 2D occupancy map video comparison of the ground truth, V-PCC anchor, and proposed Full OGCNN. The reconstructed occupancy map videos are derived from the first frames of Loot and Longdress. For Loot, (a), (b), (c) and (d) are the occupancy map reconstructions of the Full OGCNN and the V-PCC anchor, the difference between the two, and the ground truth, respectively. (e) and (f) are the enlarged areas of the gold and blue blocks in (c). For Longdress, the same order is followed. In (c) and (i), the green pixels denote the unoccupied pixels of the V-PCC anchor correctly removed by the Full OGCNN. The red pixels denote the occupied pixels of the V-PCC anchor wrongly removed by the Full OGCNN. We can see from (c) and (i) that the number of green pixels is much greater than the number of red pixels. The 2D occupancy map results demonstrate that the proposed OGCNN can remove many noisy points and very few original points.

I. Visual results of the 3D point clouds

Fig. 8 shows a visual comparison of the original point clouds and the point clouds reconstructed by the V-PCC anchor and the proposed Half OGCNN. The zoomed figures are derived from the first frame of Loot, the first frame of RedandBlack, the first frame of Longdress, and the 300th frame of Longdress. From these frames, we can clearly see from the red and green rectangles that there are many noisy points in the reconstructions of the V-PCC anchor. However, the reconstructions of the Half OGCNN are much smoother and closer to the original point clouds. The visual results demonstrate that the proposed OGCNN can achieve a much better subjective quality.

VI. CONCLUSION

In this paper, we first point out that the accuracy of the occupancy map video is important to the quality of reconstructed point clouds under video-based point cloud compression (V-PCC). Then, we propose an occupancy-geometry-based convolutional neural network (OGCNN) to improve the occupancy map accuracy. We formulate the problem of improving occupancy map accuracy as a binary segmentation problem. In addition to the quarter-resolution occupancy map video, we use the reconstructed geometry video as the other input. The experimental results show that our proposed OGCNN approach presents clear accuracy improvements in the occupancy map video and leads to significant BD-rate savings compared to the state-of-the-art schemes. To the best of our knowledge, this is the first CNN-based work on improving the performance of V-PCC. We will consider more CNN-based algorithms to improve the performance of V-PCC in the future.

REFERENCES

[1] K. Guo, F. Xu, T. Yu, X. Liu, Q. Dai, and Y. Liu, "Real-time geometry, albedo, and motion reconstruction using a single rgb-d camera," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, p. 1, 2017.

[2] E. dEon, B. Harrison, T. Myers, and P. Chou, "Input to ad hoc groups on mpeg point cloud compression and jpeg pleno," *Document ISO/IEC JTC1/SC29/WG11 m40059, Geneva, Switzerland*, 2017.

[3] E. S. Jang, M. Preda, K. Mammou, A. M. Tourapis, J. Kim, D. B. Graziosi, S. Rhyu, and M. Budagavi, "Video-based point-cloud-compression standard in mpeg: from evidence collection to committee draft [standards in a nutshell]," *IEEE Signal Processing Magazine*, vol. 36, no. 3, pp. 118–123, 2019.

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

[5] C. Tulvan, R. Mekuria, Z. Li, and S. Laserre, "Use cases for point cloud compression (pcc)," 2016.

[6] J. Chen, C. Lin, P. Hsu, and C. Chen, "Point cloud encoding for 3d building model retrieval," *IEEE Transactions on Multimedia*, vol. 16, no. 2, pp. 337–345, 2014.

[7] Culture 3D Cloud. [Online]. Available: http://c3dc.fr/

[8] H. Fuchs, A. State, and J.-C. Bazin, "Immersive 3d telepresence," *Computer*, vol. 47, no. 7, pp. 46–52, 2014.

[9] D. Sportillo, A. Paljic, M. Boukhris, P. Fuchs, L. Ojeda, and V. Roussarie, "An immersive virtual reality system for semi-autonomous driving simulation: a comparison between realistic and 6-dof controller-based interaction," in *Proceedings of the 9th International Conference on Computer and Automation Engineering*, 2017, pp. 6–10.

[10] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1907–1915.

[11] Mobile Mapping System. [Online]. Available: http://www.mitsubishielectric.com/bu/mms/index.html

[12] S. Schwarz, M. Preda, V. Baroncini, M. Budagavi, P. Cesar, P. A. Chou, R. A. Cohen, M. Krivokuća, S. Lasserre, Z. Li *et al.*, "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, 2018.

[13] G. J. Sullivan, J.-R. Ohm, W.-J. 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, 2012.

[14] A. Vosoughi, S. Yea, and S. Liu, "New proposal on occupancy map recovery using scalable locally adaptive erosion filter," *Document ISO/IEC JTC1/SC29/WG11 MPEG2019/ m46347, Marrakesh, Morocco*, 2019.

[15] R. J. Youngho Oh and M. Budagavi, "Improved point cloud compression through filtering of occupancy map," *Document ISO/IEC JTC1/SC29/WG11 MPEG2019/ m46370, Marrakesh, Morocco*, 2019.

[16] Y.-H. Lee, J.-L. Lin, Y.-C. Chang, C.-C. Ju, Y.-T. Tsai, C.-C. Lin, C.-L. Lin, Y. Oh, R. Joshi, and M. Budagavi, "New proposal on occupancy map refinement using corner-based boundary estimation," *Document ISO/IEC JTC1/SC29/WG11 MPEG2019/ m46389, Marrakesh, Morocco*, 2019.

[17] K. Cai, D. Zhang, V. Zakharcchenko, and J. Chen, "Adaptive occupancy map up-sampling," *Document ISO/IEC JTC1/SC29/WG11 MPEG2019/ m46455, Marrakesh, Morocco*, 2019.

[18] H. Najaf-Zadeh, M. Budagavi, R. Joshi, and Y. Oh, "Constrained occupancy map trimming using a ternary occupancy map," *Document ISO/IEC JTC1/SC29/WG11 MPEG2019/ m47593, Geneva, CH*, 2019.

[19] L. Li, Z. Li, S. Liu, and H. Li, "Efficient projected frame padding for video-based point cloud compression," *IEEE Transactions Multimedia*, 2020.

[20] S.-P. Wang, Y.-T. Tsai, C.-C. Lin, C.-L. Lin, Y.-H. Lee, J.-L. Lin, Y.-C. Chang, and C.-C. Ju, "Bounding box shifting for occupancy map generation," *Document ISO/IEC JTC1/SC29/WG11 MPEG2019/ m47766, Geneva, CH*, 2019.

[21] P. Andrivon, J. Ricard, C. Guede, O. Nakagami, D. Graziosi, and A. Tabatabai, "Patch border filtering specification in V-PCC," Document ISO/IEC JTC1/SC29/WG11 m51501, Geneva, CH, Oct. 2019.

[22] C. Guede, J. Ricard, J. Llach, J.-C. Chevet, Y. Olivier, and D. Gendron, "Improve point cloud compression through occupancy map refinement," *Document ISO/IEC JTC1/SC29/WG11 MPEG2018/ m44779, Macao, China*, 2018.

[23] 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*, 2018.

[24] J. Kammerl, N. Blodow, R. B. Rusu, S. Gedikli, M. Beetz, and E. Steinbach, "Real-time compression of point cloud streams," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 778–785.

[25] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based compression of dynamic 3d point cloud sequences," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1765–1778, 2016.

[26] R. L. de Queiroz and P. A. Chou, "Motion-compensated compression of dynamic voxelized point clouds," *IEEE Transactions on Image Processing*, vol. 26, no. 8, pp. 3886–3895, 2017.

[27] 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, 2016.

[28] 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*, 2017.

[29] 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)*. IEEE, 2017, pp. 1–6.

[30] S. Lasserre, J. Llach, C. Guede, and J. Ricard, "Technicolor's response to the cfpp for point cloud compression," *Document ISO/IEC JTC1/SC29/WG11 m41822, Macau, China*, 2017.

[31] 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*, 2017.

[32] M. Preda, "Report on pcc cfp answers," *Document ISO/IEC JTC1/SC29/WG11 w17251, Macau, China*, 2017.

[33] L. Li, Z. Li, S. Liu, and H. Li, "Occupancy-map-based rate distortion optimization and partition for video-based point cloud compression," *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.

[34] Point Cloud Compression Category 2 Reference Software TMC2-11.0. [Online]. Available: http://mpegx.int-evry.fr/software/MPEG/PCC/TM/mpeg-pcc-tmc2

[35] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," Document VCEG-M33, Austin, Texas, USA, April 2001.

[36] D. Tian, H. Ochimizu, C. Feng, R. Cohen, and A. Vetro, "Geometric distortion metrics for point cloud compression," in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 3460–3464.

[37] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Computational intelligence and neuroscience*, vol. 2018, 2018.

[38] M. Committee, "V-PCC Codec Description," Document ISO/IEC JTC1/SC29/WG11 w19526, Virtual, Italy, Sep. 2020.

[39] X. Chen, D. P. Kingma, T. Salimans, Y. Duan, P. Dhariwal, J. Schulman, I. Sutskever, and P. Abbeel, "Variational lossy autoencoder," *arXiv preprint arXiv:1611.02731*, 2016.

[40] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[41] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[42] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks." in *Cvpr*, vol. 10, 2010, p. 7.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[44] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of operations research*, vol. 134, no. 1, pp. 19–67, 2005.

[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.