IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 7, 2020, accepted August 31, 2020, date of publication September 18, 2020, date of current version September 29, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3024633

Visual Quality of Compressed Mesh and Point Cloud Sequences

KEMING CAO[1], (Student Member, IEEE), YI XU[2], AND PAMELA COSMAN[1], (Fellow, IEEE)
[1]Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA 92093, USA
[2]Owlii Inc., Beijing 100086, China

Corresponding author: Pamela Cosman (pcosman@eng.ucsd.edu)

This work was supported in part by the National Science Foundation (NSF) under Grant IIS-1522125, and in part by the Owlii Inc.

ABSTRACT With the development of immersive video, the delivery and storage of 3D content have become important research areas. While compression methods for meshes and point clouds, the two main representations for 3D content, are actively studied, there are few studies of their perceptual compression quality and none that consider observation distance. In this paper, we study the perceptual quality of compressed 3D sequences, for both point cloud compression and mesh-based compression. We explore the impact of bit rate and observation distance on perceptual quality. Evaluation of perceptual quality is carried out both by collecting viewer opinion scores of the compressed sequences separately, and with a side-by-side comparison. A functional model for mesh and point cloud compression quality is estimated to predict Mean Opinion Score (MOS) which yields high Pearson correlation and rank correlation scores with measured MOS.

INDEX TERMS Point cloud compression, mesh compression, compression quality model, video quality evaluation, perceptual quality assessment.

I. INTRODUCTION

Immersive video, as one rapidly growing multimedia form in daily life, is an important future direction of interaction with content and the real world, due to the richer experience it provides compared to traditional 2D media content, including innovative navigation and interactive functionalities [1], [2]. In immersive video, viewers can see details without constraint on the viewpoint, since immersion is guaranteed through 6 degrees of freedom. The advantages of immersive video lead to multimedia applications in many areas, such as teleconferencing [3], [4], sports [5] and education [6]. In this paper, we refer to this kind of volumetric video as 3D video. The development of depth sensors, such as Kinect from Microsoft and RealSense from Intel, led to an increase in 3D data processing applications. Also, research into autonomous vehicles has boosted the requirement to process 3D information to understand the surrounding world. However, along with the advantages over traditional 2D video, huge amounts of 3D data lead to storage problems and the need for compression.

The associate editor coordinating the review of this manuscript and approving it for publication was Shiqi Wang.

There are two main representations for 3D data, mesh and point cloud (PC). A mesh represents 3D content with faces (represented by edges and vertices) that define the shape, and texture information that defines the color across the surface of each face. Point clouds represent 3D content with a collection of points in 3D space; each point is associated with attributes such as color information. Visual quality of mesh and PC compression, including their comparative performance, is an important and relatively new area of study. Difficulties arise from the lack of promising objective metrics. There are few objective metrics that work on both mesh and PC representations, and most current objective metrics are poorly correlated with human perception [7]–[12]. Most current work on quality evaluation focuses on geometry distortion and ignores texture distortion, however, the overall visual quality is affected by both. Subjective experiments under varying conditions, as well as objective metrics correlated with perception, are needed. In this context, the contributions of this paper are:

- We compare PC sequence compression and mesh sequence compression which allows us to determine which representation is preferred depending on factors of sequence content, bit rate and observation distance. We use the evaluation method in [13] from MPEG,

in which a virtual viewing trajectory is chosen and the 3D sequence is rendered into 2D for subjective viewing.
- A model of subjective rating estimation is proposed which consists of two parts, bit rate quality factor (BQF) assessing the quality of compression based on bit rate, and observation distance correction factor (ODCF) making a correction over BQF with observation distance. Our model fits well with subjective ratings.

This paper is structured as follows. Sec. II introduces prior work on compression and quality evaluation of meshes and point clouds. Sec. III describes the subjective experiment. Experimental results and the estimation model are in Sec. IV. Sec. V provides compression suggestions based on the subjective test results, and conclusions are in Sec. VI.

II. RELATED WORK

A. 3D CONTENT COMPRESSION

For mesh compression, triangle fan-based compression (TFAN) [14] enumerates triangle connection cases to encode the connection information of a mesh so as to improve compression efficiency. Google's open-source compression tool, Draco, is based on the corner-table method [15] and achieves real-time compression and decompression. Similar connectivity pattern information is used to improve encoding efficiency for mesh compression in [16]. The whole mesh is divided into a few sub-meshes and compressed with a graph Fourier transform in [17] which approximates the connectivity of a sub-mesh with a sparse matrix.

For point clouds, since the points are not connected as they are in a mesh, the spatial organization of points should be built so as to encode the points efficiently. Research on PC compression has included many diverse approaches. PC reconstruction based on rank minimization theory is proposed in [18] to complete holes in dynamic PC sequences. Octrees are applied in [19], [20] to progressively compress point clouds. In [21], octrees and graph-based transforms are combined to compress geometry information in static PCs. Hierarchical clustering of points can generate Level of Detail (LoD) in [22], which describes different levels of complexity for a PC, and LoD is progressively compressed. A hierarchical sub-band transform that resembles an adaptive variation of a Haar wavelet is applied in [23] for color attribute compression for PCs. A motion-compensated approach to encoding dynamic PC sequences is proposed in [24]. An encoding scheme for building a 3D model is based on a set of low-frequency spherical harmonic basis functions in [25].

MPEG hosted a call for proposals [13] and picked three methods [26] as winners for three different categories: static models, dynamic sequences and dynamic acquisition. Test Model Category 2 (TMC2) from Apple Inc. [27] achieves the best subjective and objective quality under given target bit rates for the dynamic sequence category. Its core idea is to project points, both their geometry coordinates and attributes, to 2D and convert the 3D sequence to a 2D sequence. Then any existing video codec, such as FFmpeg or HM (Test Model for HEVC) could be used to compress the 2D sequence. In their framework, after the resolution of the 2D sequence is fixed as an initial parameter, any scale-up or scale-down of the 2D sequence will create outlier points when projected back to 3D space. To add spatial scalability to TMC2, [28] proposed to add a patch-aware averaging filter to remove outliers.

B. 3D CONTENT QUALITY ASSESSMENT

Subjective quality evaluation and computable quality evaluation of 3D representations are both active research topics. Subjective quality for 3D content is much less well studied than for 2D video. Prior work on subjective evaluation and objective metrics for PCs are summarized in [29]. Most recent work focused on evaluating subjective quality or Quality of Experience (QoE) for static 3D objects, including comparison with computable metrics. In [13], MPEG adopted a subjective experiment approach to evaluate the quality of compressed PCs, in addition to the conventional point-to-point and point-to-plane metrics. The impact of different noise levels on QoE of PCs was studied in [30]. Augmented reality head-mounted displays were used in subjective evaluation of PCs in [31]. In [32], different subjective methodologies were studied, such as Absolute Category Rating (ACR) and Double Stimulus Impairment Scale (DSIS). The authors concluded that they performed similarly for PC subjective experiments involving the quality of compression-like distortions. In [9], 20 subjects subjectively scored PC compressed videos from 1 (Bad) to 5 (Excellent), however, this research only considered different encoding configurations without comparing compression bit rates. Also, subjects were allowed to observe the sequence with free viewpoint which might lead to a variety of results since people may focus on different local details. The impact of different reduction methods for PCs on the QoE of rendered images was investigated in [33].

There is less research on subjective quality of 3D sequences. A subjective experiment on TMC2 compression for two PC sequences was carried out in [34]; the authors found that perceptual quality was more affected by texture distortion than geometry distortion. The impact of different rendering configurations on QoE in VR-based training was studied in [35]. Better immersion and faster interaction with 3D content was found to affect subjective quality in [36]. QoE of adaptive PC streaming was investigated in [37] with different network configurations.

One prior work compared the visual quality of mesh and PC representations. In [38], a comparison between colored mesh and PC concluded that mesh compression generates better visual quality at high bit rates while PC compression is better at low bit rates. Our work differs from [38] in that we consider multiple observation distances, and our compression pipelines allow PC scaling and mesh simplification, which can change the visual tradeoffs. In addition, we include a model for estimating quality ratings based on bit rate and observation distance.

Computable quality metrics for 2D and 3D sequences can be categorized as Full-Reference (FR) metrics, which have

access to the original sequence as well as the distorted one, Reduced-Reference (RR) metrics, which have access to the distorted sequence and to some key parameters extracted from the original sequence, and No-Reference (NR) metrics [39], which do not have access to the original reference sequence. NR metrics have been further subdivided into pixel-based (NR-P) metrics which use the distorted sequence to evaluate quality, and bit-stream-based (NR-B) quality predictors which do not make use of the distorted sequence directly, but rather use basic bit stream parameters such as the bit rate and packet loss rate to predict quality. Because of different properties of mesh and point cloud representations, different metrics are applied. A survey of perceptually-based computable metrics for visual impairment of 3D objects appears in [40].

Among FR metrics of mesh quality, root mean square error (RMS) and Hausdorff distance (HD) were adopted as straightforward metrics, but they were found to be poorly correlated with human perception [7], [8]. Curvature was computed on different scales of a distorted mesh and its reference in [7], based on which a correspondence was found to generate a mapping between meshes. Then, inspired by the work of [41], a structural similarity index on 3D, named Mesh Structural Distortion Measure (MSDM) was implemented on all scales. A final score was computed as a weighted combination considering all scales. Similarly, based on curvature, [42] took visual masking and saturation effects into consideration to correct scores directly from curvature. Quality for watermarked meshes was explored in [43] by measuring the difference of surface roughness between watermarked and original meshes. Measuring distance between curvature tensors of two triangle meshes under comparison was proposed in [44]. A RR method was developed in [45] that considered distributions of extracted parameters from dihedral angles of distorted and reference meshes. The Kullback-Leibler divergence between the distributions was calculated as a perceptual distance. In [46], a local roughness measure was derived from Gaussian curvature. A NR method proposed in [47] fed distributions of dihedral angles into a trained support vector regression to predict quality scores. With the development of neural networks, [48] took mean curvature as input to a regression neural network to predict a quality score without a reference mesh. All of this prior work focused on meshes without color.

For computable metrics of compressed PCs, similar to [7], [8], simple point-to-point and point-to-plane FR objective metrics, such as RMS and HD, showed no correlation with perceptual quality [9]–[12]. The conclusion that point-to-point and point-to-plane metrics are limited in predicting subjective quality ratings especially for TMC2 was also verified in [49], [50]. In [12], point-to-point and point-to-plane metrics were studied for a PC de-noising algorithm; they concluded that the point-to-plane metric is more correlated with perceptual quality than is a point-to-point metric for PC with no noise. In [51], the normal vectors used for the point-to-plane metric were averaged within a small local region to avoid the error of estimated normal from geometric distortions. In [52], a plane-to-plane FR metric measured angular similarity through the intersection angle of normal vectors between two corresponding points. While the aforementioned work also focused only on geometric distortion, two studies considered color attributes as well [53], [54]. In [53], the authors rendered a 3D point cloud onto a 2D plane, then applied traditional image FR quality metrics to measure the 2D image quality as input to their prediction model for perceptual quality of the 3D PC. Different objective metrics were proposed in [54] based on geometry, normal vectors, curvature and color separately, and the color-based metric was found to best match perceptual quality.

The current work differs from this prior work in several ways. We provide a comparison of PC and mesh compression across many different bit rates and across three different observation distances, in order to explore the effect that both rate and observation distance have on perceived quality. Unlike most prior work, we consider 3D content with color. We include the possibility of reducing the number of triangles or number of points. Lastly, we develop a simple NR quality predictor which can predict subjective quality scores for both mesh and PC compression as functions of bit rate and observation distance. A list of terms and symbols used in this paper is provided in Table 1.

TABLE 1. Definitions of Terms and Symbols.

Terms & Symbols	Definitions
Mesh Compression:	
N_t	Number of triangles per frame
L	Atlas image size after scaling
q	Quantization parameters for atlas image sequences
β	Quantization step size for vertex coordinates
γ	Quantization step size for vertex-atlas mapping
PC Compression:	
s	Down-scaling factor for PC sequence
l_x, l_y	Size of projected image
$QP_{geometry}$	Quantization parameter for geometry
$QP_{texture}$	Quantization parameter for texture
MOS Estimation Model:	
r	Bit rate
r_{max}	Maximum bit rate (25Mbps in our case)
d	Observation distance
BQF	Bitrate Quality Factor, defines general trend of quality when bitrate changes
ODCF	Observation Distance Correction Factor, equals ratio of quality score at distance d to quality score at maximum observation distance, for a given bitrate
w	Parameter that controls BQF
$slope()$	Slope function of ODCF
$b()$	Intercept function of ODCF
m, p, t	Parameters of slope function
ρ	Pearson Correlation
MSE	Mean Squared Error for fitting evaluation
Choice of Number of Triangles and Scale Factor:	
k, a, c	Parameters involved in number of triangles and scale factor recommendation

III. SUBJECTIVE EXPERIMENT

Here we design a subjective test to compare the quality under compression of PC and mesh representations of 3D content.

A. TEST SEQUENCES

We use four 3D dynamic sequences: *Basketball*, *Dancer*, *Model* and *Exercise*. Each sequence has 300 frames. In Fig. 1, we show example frames to illustrate sequence content. *Exercise* contains a man wearing solid-color clothing; he does exercises slowly, with a large movement range. In *Dancer*, a man in solid-color clothing does an urban dance with fast movement and large movement range. The *Model* sequence shows a woman wearing a patterned dress with a swirly skirt; this sequence contains more complex texture than the others. Lastly, *Basketball* involves two objects, a basketball and a man; he plays with the basketball with a moderate motion speed. Both the ball and the man's shirt have some color variation.

Fig. 2(a) shows the camera setup for data acquisition, consisting of 75 stationary cameras, of which 25 are color and 50 infrared. Fig. 2(b) shows the data acquisition pipeline. The whole pipeline is similar to [55]. The 50 infrared cameras lead to 25 depth images which are aligned with the 25 color images for each frame in the 'Generating Correspondence' module. Those 25 image pairs for each frame constitute the raw data of each sequence.

For the mesh version of each sequence, starting from the raw data, all image pairs are fed into a foreground segmentation algorithm. The foreground segmentation algorithm is adopted from [55]; it considers a confidence map for an RGB image, IR image and Shape from Silhouette jointly in order to generate a good segmentation. After foreground segmentation, a triangle mesh is reconstructed with a given rough number of triangles. Because the captured content is dynamic, a mesh tracking method, non-rigid Iterative Closest Point [56], is applied to ensure topology consistency in each group of frames. The reconstructed mesh has vertices with coordinates ranging from 0 to 2048. At the last step, a texture atlas image, containing the color information of size 2048×2048, is generated.

For the PC version of each sequence, we first generate a mesh with 100k triangles per frame as the source mesh. The number of triangles is high enough to ensure good quality. Then, we sample across the mesh surface with an interval of 0.5 which leads to a source PC with approximately 2.5 million points per frame.

B. SEQUENCE PREPARATION

Each test sequence is encoded at five target bit rates: 3Mbps, 6Mbps, 9Mbps, 15Mbps, and 25Mbps. Those target bit rates are chosen to cover a large range which will satisfy many potential applications. We render the 3D content as depicted in Fig. 3. Using OpenGL in a Linux OS, we obtain rendered 2D images of the 3D sequences out of which we generate video sequences that are shown to subjects. The observation

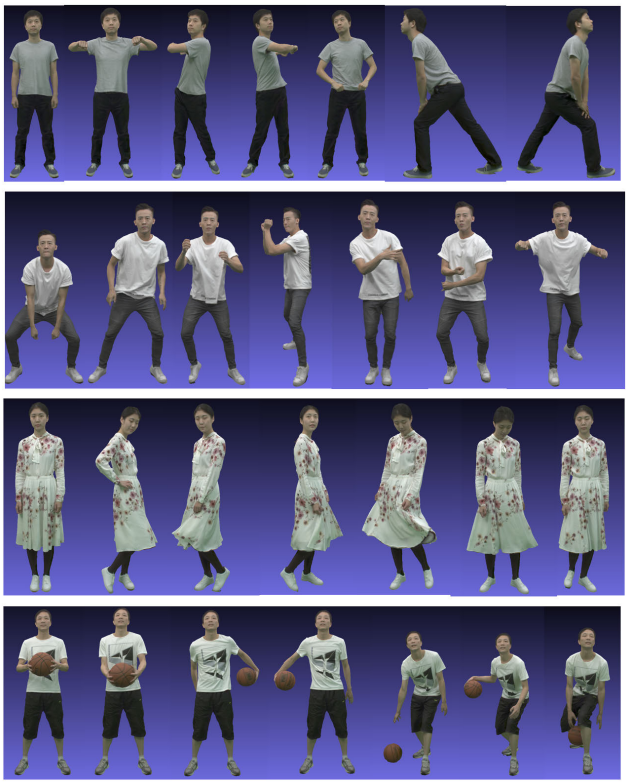

FIGURE 1. Example frames from sequences. From top to bottom, the sequences are *Exercise*, *Dancer*, *Model* and *Basketball*.

distance, which is the distance between the sequence centroid and the virtual rendering camera, takes the values 1.5m (close), 3.0m (middle) and 5.0m (far) where m represents meters. The observation distances chosen represent a significant range of possible observation distances. Observing at 1.5m or closer generally does not allow the whole body to be seen, while observing at 5m or farther means the body appears rather small. By projecting each point or face to the image plane, OpenGL simulates what the 3D content will look like from a camera positioned at the specified observation distance from the center of the object. The virtual viewing trajectory makes a full 360 degree turn around the main figure, while making a small variation in height (somewhat higher and lower than the midpoint). The average distance between the virtual rendering camera and the figure is roughly the observation distance (actual distance could be a little closer or further, less than 5% difference). These virtual viewing trajectories are chosen to present a large set of angles and details to be examined by the subjects. During the rendering process, the size of rendered videos is fixed to 2048×2048.

The observation distance is defined as the distance between the sequence centroid and the virtual camera. In a real application, observation distance would typically be an input from the user, who chooses the distance from which to view the content. Once an observation distance (and angle) are chosen by the user, that determines the rendering parameters which are needed to achieve that view, but it does not determine the

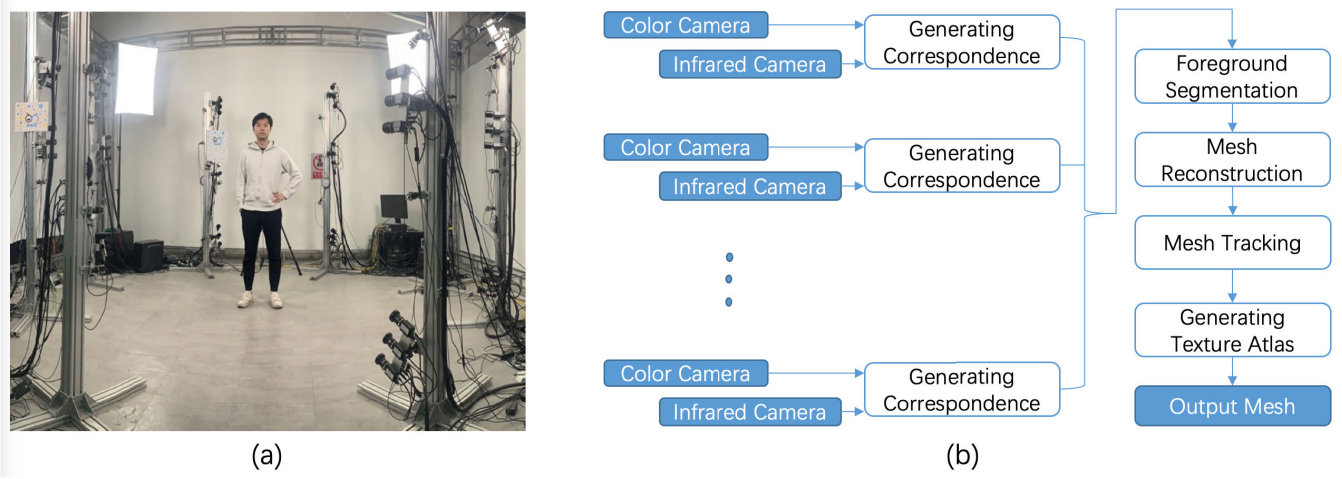

FIGURE 2. (a) Camera setup to capture 3D dynamic sequences. (b) Pipeline of data acquisition process.

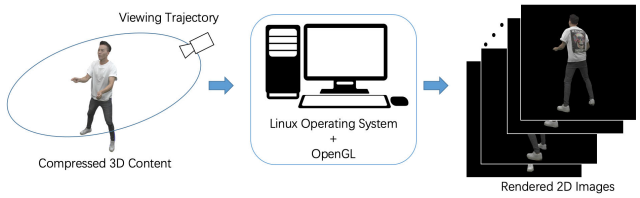

FIGURE 3. Illustration of virtual viewing trajectory and rendering process.

encoding parameters. Different observation distances could correspond to different applications. For example, teleconferencing might involve a close distance, whereas a user watching a performance might choose a far distance.

Then, for each target bitrate and observation distance, we aim to pick the compression parameters with the best visual quality for evaluation. Compression parameter sets are different for mesh and point cloud representations, and they are chosen as follows:

1) FOR MESH REPRESENTATION

This consists of two parts, the atlas image and vertex information. Atlas images are formed into a sequence and compressed with any video codec. We used FFmpeg (version 3.4.1) to compress it, involving two compression factors: atlas image size L and compression quantization parameter (QP) q. The vertex information contains vertex coordinates, vertex connection information, and vertex-atlas mapping information. Vertex coordinates are quantized with step size β. Vertex connection information is compressed losslessly with TFAN [14], chosen because MPEG adopted TFAN in their mesh coder. The vertex-atlas mapping determines the correspondence between a vertex coordinate and a pixel position in the atlas image which is a two dimensional vector for each vertex. Vertex-atlas mapping information is quantized with step size γ. All the vertex information after quantization and compression is further compressed as a subtitle bitstream in the FFmpeg framework. The number of triangles per frame N_t can also be controlled as a pre-processing step. The overall parameter set for mesh compression is formulated as the vector $(N_t, L, q, \beta, \gamma)$.

Starting with the raw data for each sequence, 6 versions of the mesh sequence are generated with different numbers of triangles per frame: 3k, 5k, 8k, 10k, 12k and 15k. For each version and each target bit rate, the compression parameter set that generates the best visual quality is manually chosen. That leads to 6 different compression parameter sets in total for each target bit rate. Then, 2D video is rendered according to a certain observation distance. We manually pick one parameter set from those 6 parameter sets as the one with the best 2D visual quality under this bit rate and this distance. For each observation distance and target bit rate, we repeat this procedure. In the end, given 5 bit rates and 3 distances for each sequence, we have 15 different compression parameter sets for each of the four test sequences.

2) FOR PC REPRESENTATION

We use TMC2, proposed as an MPEG standard, for PC compression. An important TMC2 parameter is the projected image size, l_x and l_y; the core idea of TMC2 is to project a PC onto a 2D image for both geometry and texture. Then, a conventional video codec is applied with QPs for geometry and texture ($QP_{geometry}$ and $QP_{texture}$). We used FFmpeg (version 3.4.1) as the video codec. Note that the size parameters l_x and l_y for PC compression, and L for mesh compression, are used only in the compression step, where they may differ for different bit rates or observation distances. However, following decompression, for purposes of rendering and display, all cases use the same interface and display size of 2048×2048.

Another important parameter, similar to the number of triangles per frame in mesh compression, is the down-scaling factor s which controls the number of cloud points. Down-scaling is performed over the source PC. Given a point (x, y, z) in the original PC, the scaled point is $(x/s, y/s, z/s)$.

Since the PC compression algorithm only takes integer input, the coordinates $(x/s, y/s, z/s)$ are rounded to integer. After shrinking all point coordinates and rounding to integers, some points that were distinct in the original PC will map to the same position in the down-scaled PC. These duplicate points are removed. In this way, a down-scaling of coordinates directly leads to a down-sampling of points. So the down-scaling reduces the number of points. Scaling-up is implemented when we render the scaled PC. We have 6 scale factors, 1.0, 1.25, 1.5, 1.75, 2.0 and 2.5. The compression parameter set for PC compression is $(s, l_x, l_y, QP_{geometry}, QP_{texture})$. For each target bit rate and each observation distance, we manually pick the parameter set that provides the best visual quality. In the end we have 15 parameter sets for 5 bit rates and 3 distances for each sequence.

In Table 2, we show compression parameter sets that are picked for our experiment for *Dancer*. Those parameter sets achieve the best visual quality for the given target bit rate and observation distance. To generate the candidates, we traversed all possible combinations of parameters that fit the bit rate constraint. After rendering all candidates to 2D videos, the investigators manually picked the parameter set with the best visual quality among the candidates for each observation distance. It is not always the case that the lowest QP was selected as the best candidate, because, for example, to meet the bit rate constraint, the lowest QP might occur with heavy down-scaling of the point cloud, a combination that might lead to poor quality. When we manually pick the parameters, the parameters are hidden from us to avoid bias, so we choose the parameters based on visual quality of the rendered videos. Table 2 shows a clear trend in different parameter sets for different bit rates, for example, for lower bit rates, a smaller number of triangles is a good choice for meshes, and heavier down-scaling is a good choice for point clouds. But Table 2 does not show a clear trend for observation distances, suggesting that some different combinations of parameters which achieve the same bit rate look visually similar.

There are 120 rendered videos in total, corresponding to 2 representations (mesh and point cloud), 4 sequences, and 15 sets of compression parameters (3 distances $\times$ 5 bit rates). Each video lasts 10 seconds. Fig. 4 and Fig. 5 present rendered images of the compressed PC and mesh from the 250th frame of *Model*. From the example frames, we notice that observation distance plays an important role in visual quality. At low bitrate, PC compression causes some cracks and outliers. While mesh compression appears to better preserve the geometry compared to PC compression, the texture information appears to suffer more distortion.

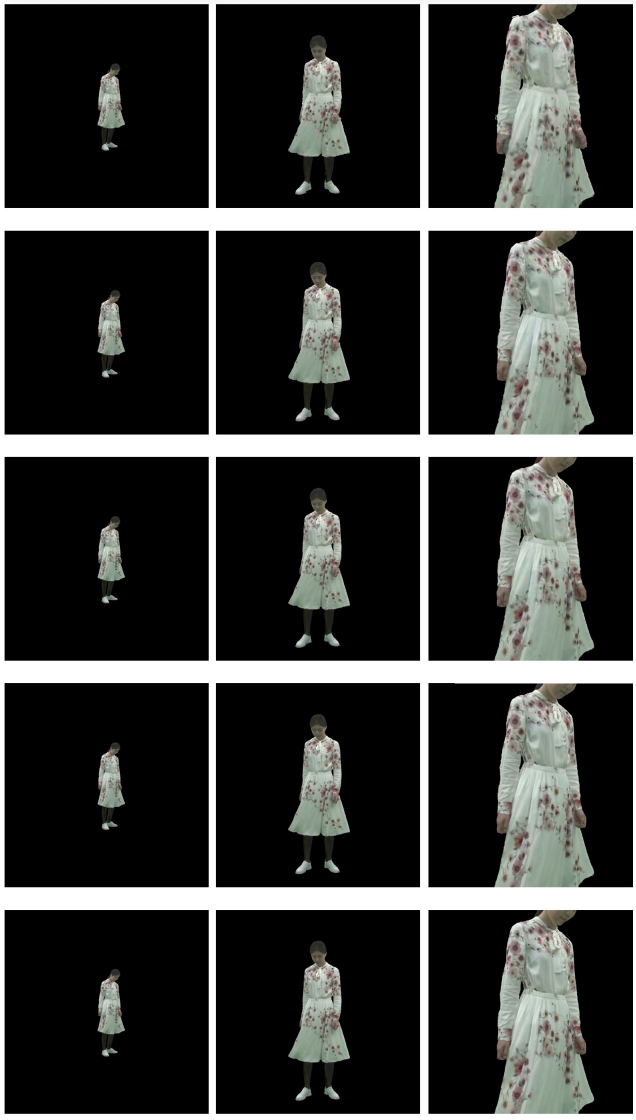

FIGURE 4. Rendered versions of the compressed PC from the 250th frame of *Model*. The bit rates are 3Mbps, 6Mbps, 9Mbps, 15Mbps and 25Mbps from the top to bottom row. From left to right, the observation distance decreases from 5m to 1.5m.

C. EXPERIMENT DESIGN

Thirty subjects (22 males, 8 females, age range 19-37) participated in our two-part experiment. For display, we used an Acer 24-Inch LED backlight monitor. In the first part, the subject scores each video from 0 (worst quality) to 100 (best quality). Before starting, subjects are shown a few example videos to calibrate their scoring standard. The 120 videos are shown in a random order. The user interface for the first part of the experiment is shown in Fig. 6. The size of the user interface (and of its displayed video) remains the same for all cases. Subjects input their score in the box according to the scale. After clicking 'Submit Score', the score is recorded and the next random video appears. Including training, this part takes approximately 30 minutes and includes 120 videos.

The second part aims to compare two videos (A and B) side by side, asking the subject which one is better based on visual quality. The videos are for the same sequence, bit rate and distance; one is from mesh representation and the other is from PC representation. The user interface for this part is shown in Fig. 7. The order of videos is random as is their placement as A or B. There are five choices: A is far better, A is a little better, they are similar, B is a little better and B

TABLE 2. Parameter sets for different target bitrates and observation distances for sequence *Dancer*.

Target Rate	3 Mbps	6 Mbps	9 Mbps	15 Mbps	25 Mbps
Mesh					
Close (1.5m)	(3k, 512, 31, 0.5, 1/512)	(3k, 1024, 27, 0.3, 1/1024)	(5k, 1024, 23, 0.1, 1/1024)	(10k, 2048, 28, 0.2, 1/2048)	(15k, 2048, 23, 0.1, 1/2048)
Middle (3.0m)	(3k, 512, 31, 0.5, 1/512)	(3k, 1024, 27, 0.3, 1/1024)	(8k, 1024, 26, 0.2, 1/1024)	(12k, 2048, 31, 0.1, 1/2048)	(15k, 2048, 23, 0.1, 1/2048)
Far (5.0m)	(3k, 512, 31, 0.5, 1/512)	(3k, 1024, 27, 0.3, 1/1024)	(5k, 1024, 23, 0.1, 1/1024)	(10k, 2048, 28, 0.2, 1/2048)	(15k, 2048, 23, 0.1, 1/2048)
Point cloud					
Close (1.5m)	(2, 1280, 1408, 32, 37)	(1.75, 1472,1616, 29, 31)	(1, 2560, 2656, 28, 39)	(1, 2560, 2656, 21, 35)	(1, 2560, 2656, 20, 26)
Middle (3.0m)	(2, 1280, 1408, 32, 37)	(1.75, 1472,1616, 29, 31)	(1.25, 2048, 2240, 27, 33)	(1.25, 2048, 2240, 22, 28)	(1.25, 2048, 2240, 18, 24)
Far (5.0m)	(2, 1280, 1408, 32, 37)	(1.5, 1712, 1872, 28, 35)	(1.25, 2048, 2240, 27, 33)	(1.25, 2048, 2240, 22, 28)	(1, 2560, 2656, 20, 26)

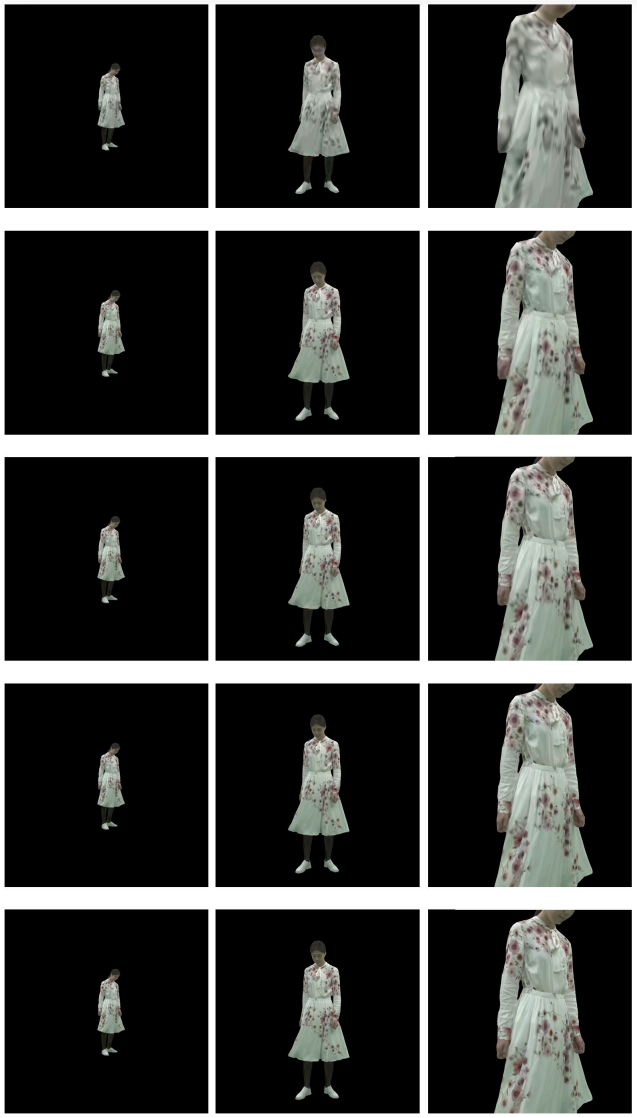

FIGURE 5. Rendered versions of the compressed mesh from the 250th frame of *Model*. The bit rates are 3Mbps, 6Mbps, 9Mbps, 15Mbps and 25Mbps from the top to bottom row. From left to right, the observation distance decreases from 5m to 1.5m.

is far better. The subject makes a choice and clicks 'Submit' to proceed to the next pair. This part of the experiment takes about 15 minutes and includes 60 video pairs.

IV. EXPERIMENT RESULTS

A. RATINGS OF MESH AND POINT CLOUD SEQUENCES

For the first part of the experiment, viewers rated sequences separately, and for this data we need to normalize scores and handle outliers. Given the rating range from 0 to 100, scores from different viewers tend to fall in quite different subranges, so we first normalize raw scores. We find the minimum and maximum scores given by each viewer for a specific sequence. For each sequence, the median of the minimum scores across viewers is denoted S_{min} (likewise S_{max}), and all viewers' scores for the sequence are normalized to the range from S_{min} to S_{max}.

We adopt a screening method from [57] to eliminate scores that are outliers or inconsistent. This method, also used in [58], makes use of the fact that our test contains videos at different bit rates.

1) First, we aim to remove outlier scores. For each video ζ, we determine the mean, standard deviation and kurtosis, denoted $\bar{u}_\zeta$, σ_ζ and $\beta_{2\zeta}$. When $2 < \beta_{2\zeta} < 4$, the distribution of scores for that video is close to normal, and a score outside the range $[\bar{u}_\zeta - 2\sigma_\zeta, \bar{u}_\zeta + 2\sigma_\zeta]$ could be regarded as an outlier.

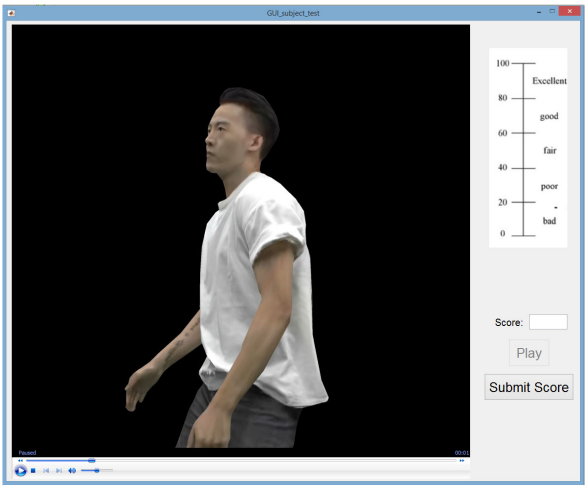

FIGURE 6. User interface for the first part of experiment.

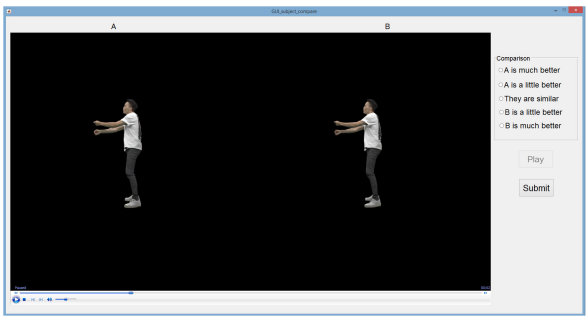

FIGURE 7. User interface for the second part of experiment.

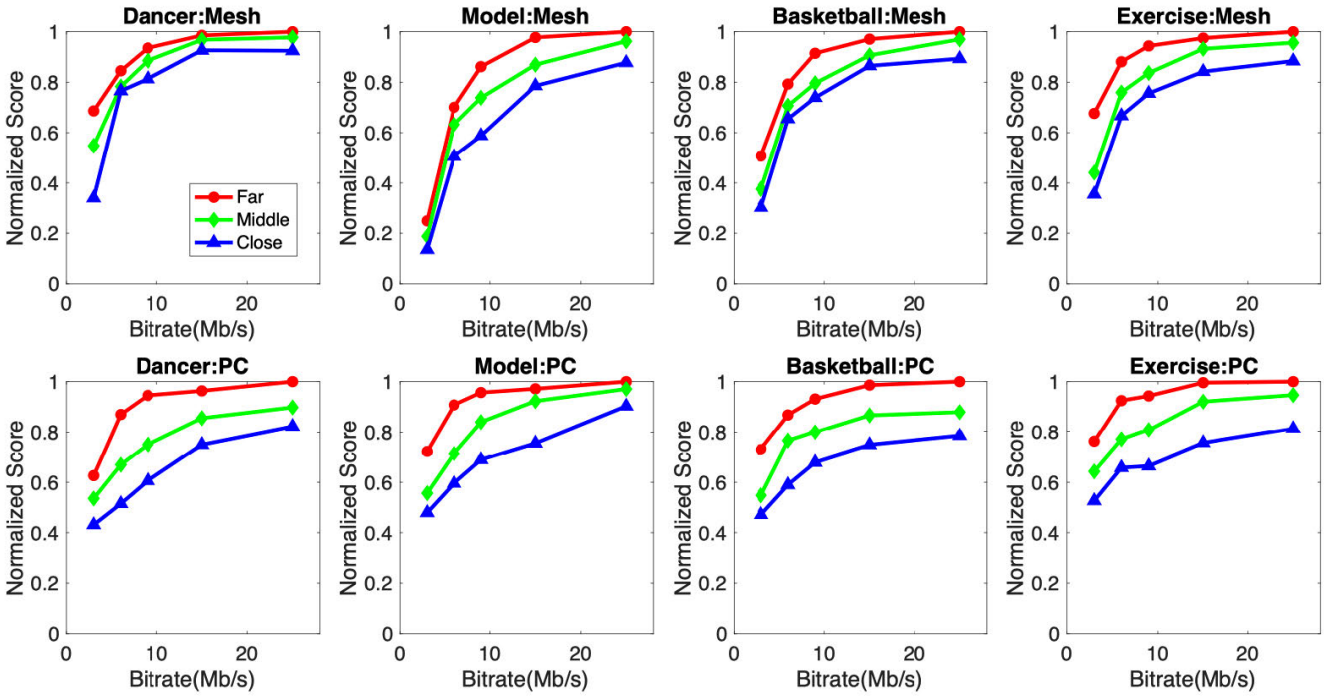

FIGURE 8. Curves of MOS vs. Bit rate for both mesh and PC compression at different observation distances.

If $\beta_{2\zeta}$ is not between 2 and 4, the range for inlier is enlarged to $[\bar{u}_\zeta - \sqrt{20}\sigma_\zeta, \bar{u}_\zeta + \sqrt{20}\sigma_\zeta]$. For each viewer, we will reject the 30 scores of this viewer for a given sequence if there are two or more scores above the upper end of the inlier range, or if there are two or more scores below the lower end of the inlier range.

2) If scores from a viewer are consistent, the rating for a lower bit rate should not be larger than that for a higher bit rate. All 30 scores of a viewer for a certain sequence are rejected if there are more than two times that the user gives a score at any lower bit rate more than K times larger than the score given by the same user at any higher bit rate for the same sequence and observation distance. Here $K = 1.3$ is chosen empirically; the consistency is good enough to show the scoring trend without rejecting too many scores from viewers.

After screening, there are on average 22 user ratings for each sequence. We average viewer scores for the same video to determine its mean opinion score (MOS). We divide all MOS values with the highest score for each sequence and each representation to normalize the highest score to 1. We plot curves of normalized MOS vs. bit rate in Fig. 8. The curves show that increasing bit rate produces better visual quality, and for a given bit rate, closer observation distances generally receive lower quality scores than farther observation distances. In Fig. 9, we plot curves of MOS vs. distance. Given a fixed bit rate, the relationship appears close to linear, and the slope and intercept for each line depend on the bit rate. In Section IV-C we will use the data in Figs. 8 and 9 to create a model that predicts the MOS as a function of bit rate and observation distance.

B. PREFERENCE FOR MESH AND POINT CLOUD COMPRESSION

In the second part of the experiment, viewers compare sequences side-by-side, and here we use all data points and for this preference there is no normalization done. Bar charts of the data are shown in Fig. 10. The sequence *Dancer* contains fast motion and the sequence *Model* contains detailed texture. Sequence-specific conclusions that we draw from these plots are:

1) For *Basketball*, *Exercise* and *Model*, people prefer PC over mesh compression at low bit rates (e.g. 3Mbps), especially for *Model*, whose texture is richest. Mesh texture appears more blurry than PC texture at low bit rates.
2) For *Dancer* at relatively low bit rates, mesh compression is better than PC. A PC is composed of discrete points which might cause bad artifacts (such as holes) when the motion is fast, as in *Dancer*.

Combining all sequences, the left bar chart in Fig. 11 shows the total count of people who preferred mesh compression in the comparison. In the middle is the count of those choosing "similar", and PC preference is at the right. The differences between Fig. 11 and Fig. 10 are that Fig. 10 shows preferences for each sequence while Fig. 11 combines all sequences, and

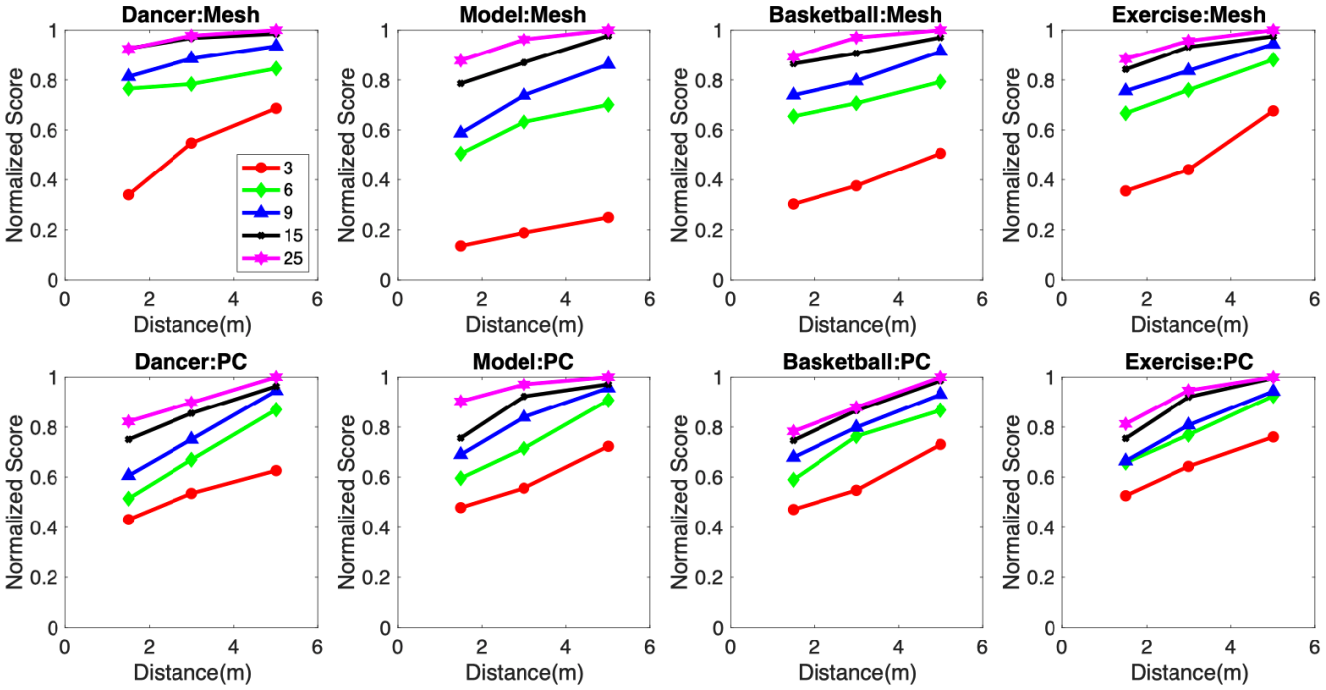

FIGURE 9. Curves of MOS vs. Distance for both mesh and PC compression for different bit rates (Mbps).

Fig. 11 combines 'a little better' and 'far better' together. Conclusions that we draw from these plots are:

1) There is a general trend that PC compression is preferred at low rates, and the two different representation types become more similar as the bit rate increases.
2) When we observe the sequence from afar, the two representations are similar except at low bit rates.
3) With decreasing observation distance, the preference for mesh compression increases.

C. OPINION SCORE MODEL

This section proposes an opinion score model that takes bit rate and observation distance as inputs. The model predicts the *MOS* score using:

$$MOS(r, d) = BQF(r) * ODCF(r, d) \tag{1}$$

Here r is bit rate and d is observation distance. Because the video version with the highest bit rate and farthest distance always gets the highest score, we have $MOS(25Mbps, 5) = 1$. $BQF(r)$ is Bitrate Quality Factor with the formulation adopted from [58]:

$$BQF(r) = \frac{1 - e^{-w\frac{r}{r_{max}}}}{1 - e^{-w}} \tag{2}$$

where r_{max} is the maximum bit rate which we take as 25Mbps and w is a parameter for $BQF(r)$. $ODCF(r, d)$ is Observation Distance Correction Factor which we define as the ratio between the score of distance d and the score of the far distance (5) for a certain bit rate r, so $ODCF(r, 5) = 1$. That

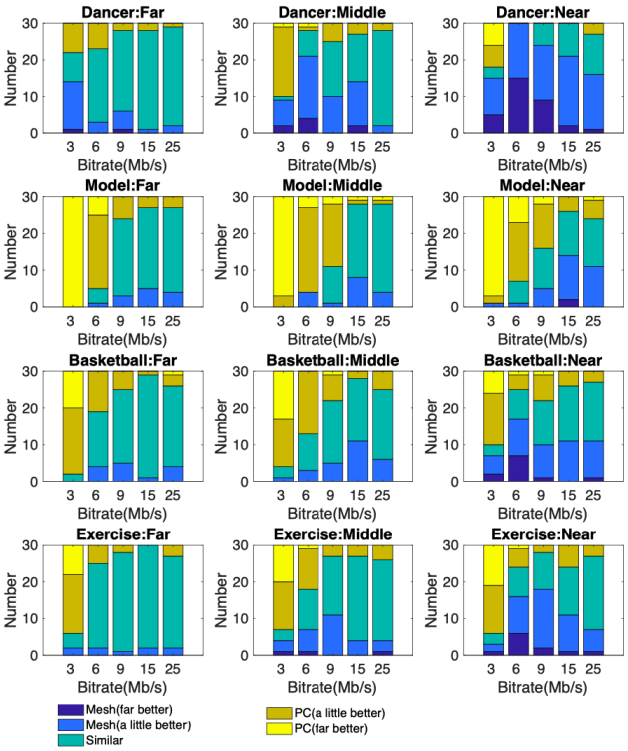

FIGURE 10. Preferences for mesh or PC compression for four different sequences at three different observation distances, across five bit rates.

leads to $MOS(r, 5) = BQF(r)$ and we could use the MOS for far distance to estimate the parameter for $BQF(r)$.

Then, we would like to determine the function $ODCF(r, d)$. We compute the $ODCF(r, d)$ value using Eq. 1. From the

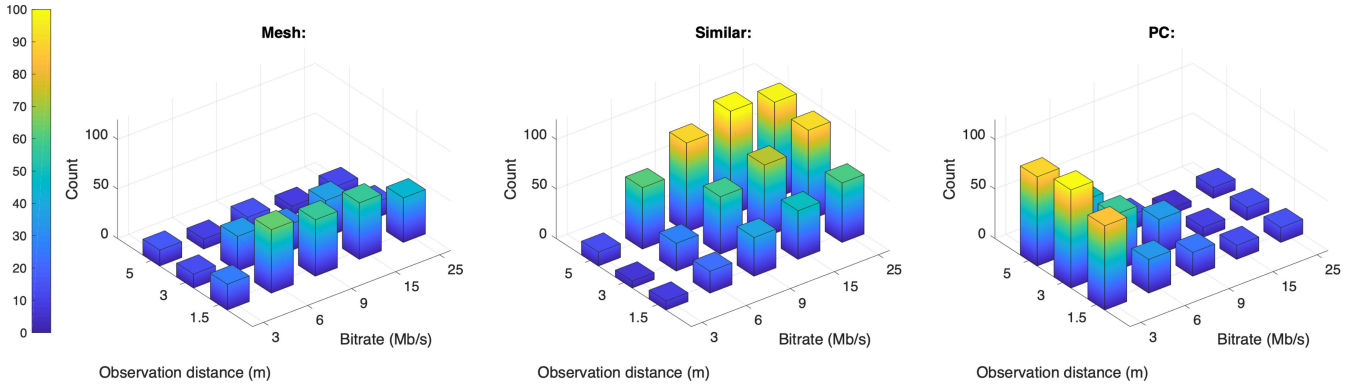

FIGURE 11. Bar chart for preferences: Mesh preferred, similar, and PC preferred.

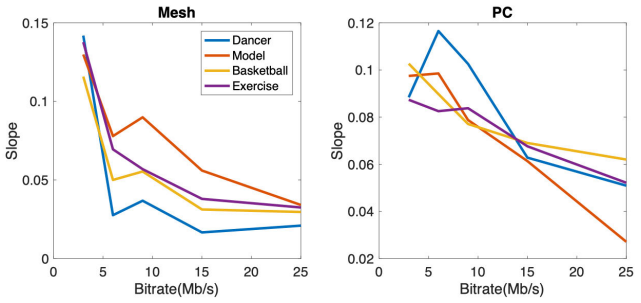

FIGURE 12. Slope vs. Bit rate from which we estimate *slope(r)*.

roughly linear curves of MOS vs. distance shown in Fig. 9, and taking the slope and intercept for each line to be dependent on the bit rate, we obtain:

$$ODCF(r,d) = slope(r) * d + b(r) \tag{3}$$

where $slope(r)$ is the slope and is a function of bit rate, and $b(r)$ is also a function of r. While the linear fit involves only three points for $ODCF(r,d)$, the three distances represent a wide range of likely observation distances, and the linear fit is simple and performs well in quality estimation, as will be shown in the following evaluation. Since $ODCF(r,5) = 1$, Eq. 3 yields $b(r) = 1 - slope(r) * 5$. We want to determine a relation between $slope(r)$ and r. First, for all 5 bit rates, we fit a linear model to $ODCF(r,d)$ and generate 5 different $slope(r)$. Curves of $slope(r)$ are shown in Fig. 12. The relation for mesh compression looks like an inverse proportional function $slope(r) = m/r$ and that for PC compression is a linear function $slope(r) = p * r + t$. Here m, p, t are all function parameters.

Then the final MOS could be computed as the multiplication of $BQF(r)$ and $ODCF(r,d)$:

$$MOS_{mesh}(r,d) = BQF(r) * (\frac{m}{r} * d + 1 - 5 * \frac{m}{r}) \tag{4}$$

$$MOS_{PC}(r,d) = BQF(r) * ((p * r + t) * d + 1 - 5 * (p * r + t)) \tag{5}$$

The fitting is performed using Matlab's built-in functions *fittype()* and *fit()*. Algorithm 1 shows the fitting process as pseudo-code. When using all four sequences for parameter estimation, the fitting result is shown in Fig. 13. Pearson Correlation (ρ), Mean Squared Error (MSE), Spearman's Rank Correlation Coefficient (SRCC) [59], Kendall Rank Correlation Coefficient (KRCC) [59] and perceptually weighted rank correlation (PWRC) [60] are used for evaluation. The evaluation results are in Table 3.

For comparison, we use the MOS prediction model from [53] which takes $MOS_{predicted} = Ax^3 + Bx^2 + Cx + D$ where A, B, C and D are parameters to be estimated and x is the score from the FR 3D objective quality metric VIFp (Visual Information Fidelity, pixel domain version) which was shown to achieve the best correlation with human perceptual quality in MOS prediction in [53]. VIFp is carried out on the projected 2D images of 3D content, and it considers color information and geometry information jointly, giving an overall score for input 3D content. VIFp can handle both mesh and PC representations.

Algorithm 1 Model Fitting for $MOS(r,d)$

Require: Scores for each videos
Consider when $d = 5$, $ODCF(r,5) = 1$
$MOS(r,5) = BQF(r)$
Fit model $BQF(r)$ with scores at far distance to get w

Compute $ODCF(r,d) = MOS(r,d)/BQF(r)$
Fit linear model $ODCF(r,d) = slope(r) * d + b(r)$
Based on definition, $ODCF(r,5) = 1$
$b(r) = 1 - slope(r) * 5$
$ODCF(r,d) = slope(r) * d + b(r) = slope(r) * (d-5) + 1$

if Sequence is Mesh **then**
  $slope(r) = m/r$
  Fit $ODCF(r,d)$ to get m
else {Sequence is PC}
  $slope(r) = p * r + t$
  Fit $ODCF(r,d)$ to get p, t
end if
$MOS(r,d) = BQF(r) * ODCF(r,d)$

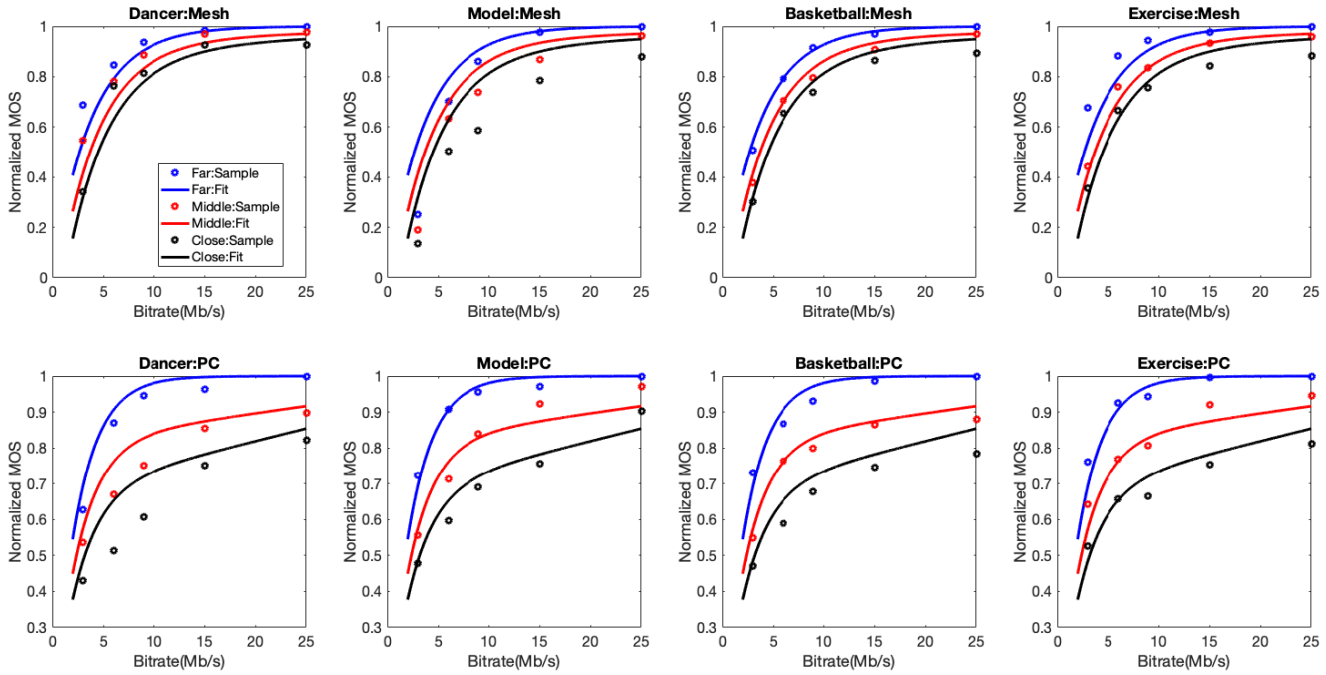

FIGURE 13. Fitting curve of proposed model.

TABLE 3. Fitting evaluation of our proposed model and of the VIFp-based model from [53].

Sequence	Mesh					Point Cloud				
	ρ	MSE	SRCC	KRCC	PWRC	ρ	MSE	SRCC	KRCC	PWRC
Our model:										
Dancer	0.977	0.0050	0.979	0.943	0.936	0.983	0.0041	0.993	0.962	0.976
Basketball	0.994	0.0008	0.986	0.943	0.936	0.985	0.0013	0.989	0.943	0.955
Model	0.983	0.0180	0.996	0.981	0.986	0.980	0.0011	0.982	0.924	0.929
Exercise	0.973	0.0029	0.968	0.867	0.878	0.969	0.0014	0.979	0.924	0.933
Model in [53]:										
Dancer	0.870	0.0164	0.964	0.867	0.894	0.946	0.0052	0.946	0.848	0.828
Basketball	0.837	0.0145	0.904	0.810	0.844	0.946	0.0032	0.921	0.790	0.767
Model	0.905	0.0208	0.963	0.880	0.891	0.974	0.0035	0.961	0.867	0.863
Exercise	0.850	0.0104	0.911	0.790	0.818	0.958	0.0033	0.961	0.848	0.854

TABLE 4. Fitting cross validation of our proposed model and of the VIFp-based model from [53].

Sequence	Mesh					Point Cloud				
	ρ	MSE	SRCC	KRCC	PWRC	ρ	MSE	SRCC	KRCC	PWRC
Our model:										
Dancer	0.972	0.0078	0.975	0.924	0.916	0.982	0.0057	0.989	0.943	0.955
Basketball	0.994	0.0008	0.986	0.943	0.936	0.983	0.0015	0.989	0.943	0.955
Model	0.979	0.0292	0.996	0.981	0.986	0.976	0.0013	0.982	0.924	0.929
Exercise	0.964	0.0047	0.939	0.829	0.836	0.966	0.0019	0.979	0.924	0.933
Model in [53]:										
Dancer	0.872	0.0257	0.964	0.867	0.894	0.945	0.0068	0.946	0.848	0.828
Basketball	0.836	0.0156	0.904	0.810	0.844	0.936	0.0042	0.921	0.790	0.767
Model	0.900	0.0289	0.9634	0.880	0.891	0.969	0.0056	0.961	0.867	0.863
Exercise	0.850	0.0111	0.911	0.790	0.818	0.953	0.0057	0.961	0.848	0.854

As in [53], we render each frame of each sequence into 2D along six axis directions, then compute the averaged VIFp of all six directions as the final VIFp value of this frame with the provided tool, Video Quality Measurement Tool (VQMT) [61]. VIFp of the whole sequence is computed as the averaged VIFp across frames. Then we apply the VIFp as input to estimate the parameters of their model based on our measured MOS data. The comparison results of predicted MOS are in Table 3. Compared to the model from [53] which is also fit to our collected MOS scores, our model better predicts MOS.

Because the MOS scores for all sequences are used to fit the models, Table 3 is overly optimistic, for both our approach and that of [53], about the correlation between actual and predicted MOS scores. Therefore, cross validation of the fitting is carried out by removing the scores from one sequence and estimating the model parameters with the remaining three sequences. Then we compute the correlation

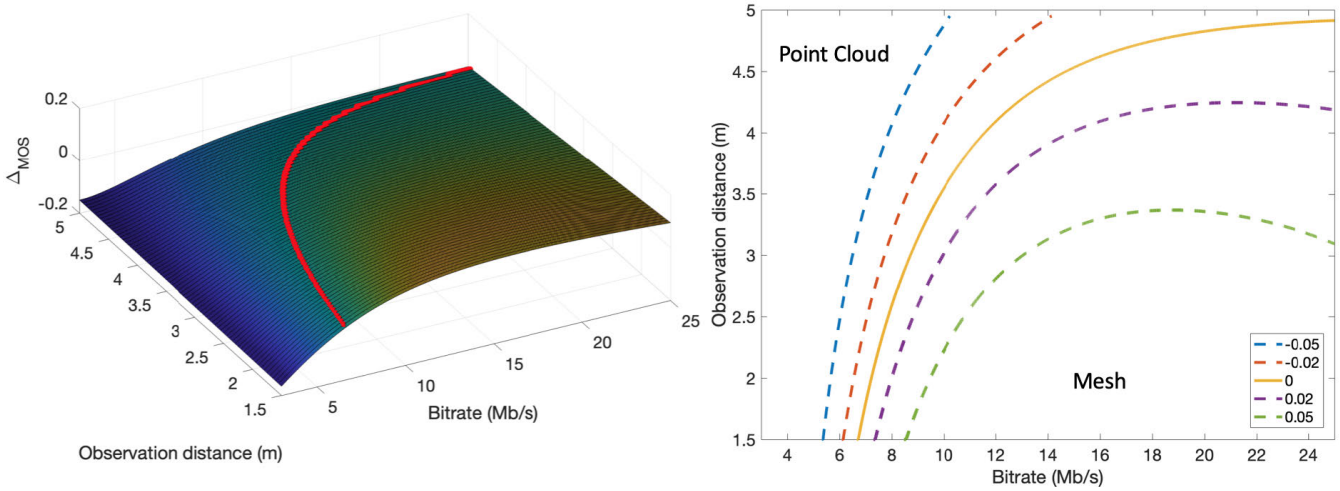

FIGURE 14. (Left) Surface plot of the difference between predicted MOS for mesh and predicted MOS for point cloud, as a function of bit rate and observation distance, (Right) Isocontours where the difference in predicted MOS takes on the values -0.05, -0.02, 0, 0.02, and 0.05.

measures and MSE between the actual scores and the estimated scores for the sequence left out. Pearson Correlation values, MSE, SRCC, KRCC and PWRC values are shown in Table 4 for all four sequences. From Table 4, we notice that the correlation coefficients are still promising even though the parameters of the model are estimated from other sequences, and the approach generally outperforms the VIFp model [53] evaluated using the same cross validation strategy.

V. IMPLICATIONS FOR COMPRESSION

These subjective test results have implications for the choice of compression method and parameters.

A. CHOICE BETWEEN POINT CLOUD AND MESH

The second part of the subjective test indicates which representation is better for certain bit rates and observation distances. Fig. 10 and Fig. 11 provide some guidance on choosing the compression method. If the required bit rate is low, such as 3Mbps, we should choose PC regardless of observation distance. Secondly, if the application requires a close observation distance and the required bit rate is not low, mesh should be chosen. For other cases, there is no preference between the representations.

We could also choose the better representation based on the opinion score model. The bit rate and distance are inputs to our proposed models. In the left plot of Fig. 14, Δ_{MOS} is defined as the difference between the MOS of mesh and PC compression, $\Delta_{MOS} = MOS_{mesh} - MOS_{point\_cloud}$. The red curve on the surface is where $\Delta_{MOS} = 0$. The surface relates to the bar charts in Fig. 11. The right plot of Fig. 14 pictures the boundary of $\Delta_{MOS} = 0$ shown as a solid line and the curves where $\Delta_{MOS} = -0.05, -0.02, 0.02, 0.05$ are shown with dashed lines. To the upper left of the solid curve, point cloud compression is preferred, and to the bottom right, the preference is for mesh. Larger values of Δ_{MOS} indicate stronger preference.

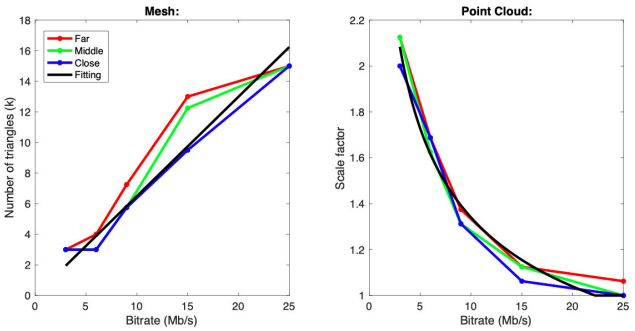

FIGURE 15. Curves of number of triangles vs. bitrate and scale factor vs. bitrate.

B. CHOICE OF NUMBER OF TRIANGLES AND SCALE FACTOR

There are several compression parameters for PC and mesh compression, including the number of points per frame for a point cloud which could be controlled by the scale factor, and the number of triangles for a mesh. From the manually chosen compression parameter sets described in Sec. III B, we plot the number of triangles N_t and scale factor s vs. bitrate in Fig. 15. In this plot, N_t and s are averaged across sequences. From the plot, we notice that the trends are similar across different observation distances, especially for PC compression. The curve for mesh compression at close distance is slightly different because at close distance, people notice more texture details so texture should account for more of the fixed bit rate. That leads to the lower number of triangles for best visual quality.

With increasing bit rate, the number of triangles tends to increase and the scale factor tends to decrease. We do linear and power law fitting as follows:

$$N_t = k * r \tag{6}$$

$$s = \begin{cases} a * r^c, & a * r^c > 1 \\ 1, & otherwise \end{cases} \tag{7}$$

where $k = 0.6496$, $a = 3.118$ and $c = -0.3667$ are model parameters. The curve fitting result is shown in Fig. 15 in black. These models can give a useful rule-of-thumb for estimating the number of triangles or scale factor given the bit rate.

VI. CONCLUSION

This study provides several new results regarding subjective quality of compressed 3D content as it relates to choice of representation, observation distance, bit rate, and scaling.

The main conclusions for this paper are as follows:

- We designed a subjective test to compare the compression quality for point cloud and mesh representations. Point cloud compression is better for low bit rates, whereas mesh compression is preferred when the observation distance is close and the bit rate is not low. When the bit rate is high, there is little difference between the two representations.
- For the two representations, we propose two models that estimate people's opinion scores and fit the experimental data well under cross-validation. The model can be used to choose a representation based on observation distance and target bit rate.

In addition, when we generated parameter sets for our experiment, we found the general trend that reducing the number of mesh triangles or reducing the number of cloud points improved visual quality at low bit rates. Suggestions for reducing the number of mesh triangles and choosing the point cloud scale factor are provided. Such reductions are not routinely considered part of the compression pipeline for 3D content, but our finding fits the well-known result for 2D content that spatial down-sampling is useful at low bit rates (see, e.g., [62], [63]). Such reductions can play an important role in preserving quality at low bit rates for 3D content as well, and would be worthy of a further subjective study.

There remains considerable room for further study of subjective quality of PC and mesh compression. We chose the bit rate and the observation distance for a compressed point cloud because they are two very important factors which affect quality, and which also do not require any computation on the actual distorted sequence at the decoder. With increased complexity, there are many other factors which affect quality at a given bit rate and given observation distance, such as the color variation in the texture, the inherent geometric complexity of the sequence, or the rapidity of the motion. Prediction of head and eye movement based on past movement or based on saliency has been carried out for 360 degree video [64], [65], and such work could inform the compression approaches as well as the viewing trajectories for future subjective experiments. For future work, more factors will also be considered, such as the sequence's complexity in geometry or texture, or the rapidity of motion, to make the prediction of quality more accurate. One limitation of this study is the limited number of sequences used as the test data set. A larger number of test sequences that can cover diverse visual content should also be involved in our future work.

REFERENCES

[1] R. Ma, T. Maugey, and P. Frossard, "Optimized data representation for interactive multiview navigation," *IEEE Trans. Multimedia*, vol. 20, no. 7, pp. 1595–1609, Jul. 2018.

[2] O. Stankiewicz, M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, and J. Samelak, "A free-viewpoint television system for horizontal virtual navigation," *IEEE Trans. Multimedia*, vol. 20, no. 8, pp. 2182–2195, Aug. 2018.

[3] P. Pourashraf and F. Safaei, "Perceptual pruning: A context-aware transcoder for immersive video conferencing systems," *IEEE Trans. Multimedia*, vol. 19, no. 6, pp. 1327–1338, Jun. 2017.

[4] C. Zhang, Q. Cai, P. A. Chou, Z. Zhang, and R. Martin-Brualla, "Viewport: A distributed, immersive teleconferencing system with infrared dot pattern," *IEEE MultimediaMag.*, vol. 20, no. 1, pp. 17–27, Jan. 2013.

[5] H. Sabirin, Q. Yao, K. Nonaka, H. Sankoh, and S. Naito, "Toward real-time delivery of immersive sports content," *IEEE MultimediaMag.*, vol. 25, no. 2, pp. 61–70, Apr. 2018.

[6] G. Makransky and L. Lilleholt, "A structural equation modeling investigation of the emotional value of immersive virtual reality in education," *Educ. Technol. Res. Develop.*, vol. 66, no. 5, pp. 1141–1164, 2018.

[7] G. Lavoué, "A multiscale metric for 3D mesh visual quality assessment," *Comput. Graph. Forum*, vol. 30, no. 5, pp. 1427–1437, Aug. 2011.

[8] M. Corsini, M.-C. Larabi, G. Lavoué, O. Petřík, L. Váša, and K. Wang, "Perceptual metrics for static and dynamic triangle meshes," *Comput. Graph. Forum*, vol. 32, no. 1, pp. 101–125, Feb. 2013.

[9] R. Mekuria, K. Blom, and P. Cesar, "Design, implementation, and evaluation of a point cloud codec for tele-immersive video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 4, pp. 828–842, Apr. 2017.

[10] E. Alexiou and T. Ebrahimi, "On subjective and objective quality evaluation of point cloud geometry," in *Proc. 9th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, May 2017, pp. 1–3.

[11] E. Alexiou, T. Ebrahimi, M. V. Bernardo, M. Pereira, A. Pinheiro, L. A. Da Silva Cruz, C. Duarte, L. G. Dmitrovic, E. Dumic, D. Matkovics, and A. Skodras, "Point cloud subjective evaluation methodology based on 2D rendering," in *Proc. 10th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, May 2018, pp. 1–6.

[12] A. Javaheri, C. Brites, F. Pereira, and J. Ascenso, "Subjective and objective quality evaluation of 3D point cloud denoising algorithms," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2017, pp. 1–6.

[13] *Call for Proposals for Point Cloud Compression V2*, Standard ISO/IEC JTC1/SC29/WG11, Hobart, TAS, Australia, Apr. 2017.

[14] K. Mamou, T. Zaharia, and F. Prêteux, "TFAN: A low complexity 3D mesh compression algorithm," *Comput. Animation Virtual Worlds*, vol. 20, nos. 2–3, pp. 343–354, Jun. 2009.

[15] J. Rossignac, "3D compression made simple: Edgebreaker with ZipandWrap on a corner-table," in *Proc. Int. Conf. Shape Modeling Appl. (SMI)*, 2001, pp. 278–283.

[16] R. Mekuria, M. Sanna, E. Izquierdo, D. C. A. Bulterman, and P. Cesar, "Enabling geometry-based 3-D tele-immersion with fast mesh compression and linear rateless coding," *IEEE Trans. Multimedia*, vol. 16, no. 7, pp. 1809–1820, Nov. 2014.

[17] A. S. Lalos, I. Nikolas, E. Vlachos, and K. Moustakas, "Compressed sensing for efficient encoding of dense 3D meshes using model-based Bayesian learning," *IEEE Trans. Multimedia*, vol. 19, no. 1, pp. 41–53, Jan. 2017.

[18] E. Vlachos, A. S. Lalos, A. Spathis-Papadiotis, and K. Moustakas, "Distributed consolidation of highly incomplete dynamic point clouds based on rank minimization," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3276–3288, Dec. 2018.

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

[20] J. Kammerl, N. Blodow, R. B. Rusu, S. Gedikli, M. Beetz, and E. Steinbach, "Real-time compression of point cloud streams," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2012, pp. 778–785.

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

[22] Y. Fan, Y. Huang, and J. Peng, "Point cloud compression based on hierarchical point clustering," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA)*, Oct. 2013, pp. 1–7.
[23] R. L. de Queiroz and P. A. Chou, "Compression of 3D point clouds using a region-adaptive hierarchical transform," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3947–3956, Aug. 2016.
[24] P. A. Chou and R. L. De Queiroz, "Motion-compensated compression of dynamic voxelized point clouds," U.S. Patent Appl. 15/168 019, Nov. 30, 2017.
[25] J.-Y. Chen, C.-H. Lin, P.-C. Hsu, and C.-H. Chen, "Point cloud encoding for 3D building model retrieval," *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 337–345, Feb. 2014.
[26] S. Schwarz, M. Preda, V. Baroncini, M. Budagavi, P. Cesar, P. A. Chou, R. A. Cohen, M. Krivokuca, 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 J. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 1, pp. 133–148, Mar. 2019.
[27] K. Mammou, A. M. Tourapis, D. Singer, and Y. Su, *Video-Based and Hierarchical Approaches Point Cloud Compression*, Standard ISO/IEC JTC1/SC29/WG11, Macau, China, Oct. 2017.
[28] K. Cao, Y. Xu, and P. C. Cosman, "Patch-aware averaging filter for scaling in point cloud compression," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Nov. 2018, pp. 390–394.
[29] E. Dumic, C. R. Duarte, and L. A. da Silva Cruz, "Subjective evaluation and objective measures for point clouds—State of the art," in *Proc. 1st Int. Colloq. Smart Grid Metrol. (SmaGriMet)*, Apr. 2018, pp. 1–5.
[30] J. Zhang, W. Huang, X. Zhu, and J.-N. Hwang, "A subjective quality evaluation for 3D point cloud models," in *Proc. Int. Conf. Audio, Lang. Image Process.*, Jul. 2014, pp. 827–831.
[31] E. Alexiou, E. Upenik, and T. Ebrahimi, "Towards subjective quality assessment of point cloud imaging in augmented reality," in *Proc. IEEE 19th Int. Workshop Multimedia Signal Process. (MMSP)*, Oct. 2017, pp. 1–6.
[32] E. Alexiou and T. Ebrahimi, "On the performance of metrics to predict quality in point cloud representations," *Appl. Digit. Image Process. XL*, vol. 10396, Sep. 2017, Art. no. 103961H.
[33] M. Seufert, J. Kargl, J. Schauer, A. Nüchter, and T. Hoßfeld, "Different points of view: Impact of 3D point cloud reduction on QoE of rendered images," in *Proc. 12th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, May 2020, pp. 1–6.
[34] E. Zerman, P. Gao, C. Ozcinar, and A. Smolic, "Subjective and objective quality assessment for volumetric video compression," *Electron. Imag.*, vol. 2019, no. 10, pp. 1–323, 2019.
[35] R. Schatz, G. Regal, S. Schwarz, S. Suettc, and M. Kempf, "Assessing the QoE impact of 3D rendering style in the context of VR-based training," in *Proc. 10th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, May 2018, pp. 1–6.
[36] K. Desai, S. Raghuraman, R. Jin, and B. Prabhakaran, "QoE studies on interactive 3D tele-immersion," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2017, pp. 130–137.
[37] J. van der Hooft, M. T. Vega, C. Timmerer, A. C. Begen, F. De Turck, and R. Schatz, "Objective and subjective QoE evaluation for adaptive point cloud streaming," in *Proc. 12th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, May 2020, pp. 1–6.
[38] E. Zerman, C. Ozcinar, P. Gao, and A. Smolic, "Textured mesh vs coloured point cloud: A subjective study for volumetric video compression," in *Proc. 12th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, May 2020, pp. 1–6.
[39] S. Kanumuri, P. C. Cosman, A. R. Reibman, and V. A. Vaishampayan, "Modeling packet-loss visibility in MPEG-2 video," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 341–355, Apr. 2006.
[40] G. Lavoué and M. Corsini, "A comparison of perceptually-based metrics for objective evaluation of geometry processing," *IEEE Trans. Multimedia*, vol. 12, no. 7, pp. 636–649, Nov. 2010.
[41] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[42] L. Dong, Y. Fang, W. Lin, and H. S. Seah, "Perceptual quality assessment for 3D triangle mesh based on curvature," *IEEE Trans. Multimedia*, vol. 17, no. 12, pp. 2174–2184, Dec. 2015.
[43] M. Corsini, E. D. Gelasca, T. Ebrahimi, and M. Barni, "Watermarked 3-D mesh quality assessment," *IEEE Trans. Multimedia*, vol. 9, no. 2, pp. 247–256, Feb. 2007.
[44] F. Torkhani, K. Wang, and J.-M. Chassery, "A curvature tensor distance for mesh visual quality assessment," in *Proc. Int. Conf. Comput. Vis. Graph.* Berlin, Germany: Springer, 2012, pp. 253–263.
[45] I. Abouelaziz, M. Omari, M. El Hassouni, and H. Cherifi, "Reduced reference 3D mesh quality assessment based on statistical models," in *Proc. 11th Int. Conf. Signal-Image Technol. Internet-Based Syst. (SITIS)*, Nov. 2015, pp. 170–177.
[46] K. Wang, F. Torkhani, and A. Montanvert, "A fast roughness-based approach to the assessment of 3D mesh visual quality," *Comput. Graph.*, vol. 36, no. 7, pp. 808–818, Nov. 2012.
[47] I. Abouelaziz, M. El Hassouni, and H. Cherifi, "No-reference 3D mesh quality assessment based on dihedral angles model and support vector regression," in *Proc. Int. Conf. Image Signal Process.* Cham, Switzerland: Springer, 2016, pp. 369–377.
[48] I. Abouelaziz, M. El Hassouni, and H. Cherifi, "A curvature based method for blind mesh visual quality assessment using a general regression neural network," in *Proc. 12th Int. Conf. Signal-Image Technol. Internet-Based Syst. (SITIS)*, 2016, pp. 793–797.
[49] E. Alexiou, I. Viola, T. M. Borges, T. A. Fonseca, R. L. de Queiroz, and T. Ebrahimi, "A comprehensive study of the rate-distortion performance in MPEG point cloud compression," *APSIPA Trans. Signal Inf. Process.*, vol. 8, no. e27, 2019.
[50] H. Su, Z. Duanmu, W. Liu, Q. Liu, and Z. Wang, "Perceptual quality assessment of 3d point clouds," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 3182–3186.
[51] D. Tian, H. Ochimizu, C. Feng, R. Cohen, and A. Vetro, "Geometric distortion metrics for point cloud compression," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3460–3464.
[52] E. Alexiou and T. Ebrahimi, "Point cloud quality assessment metric based on angular similarity," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2018, pp. 1–6.
[53] E. M. Torlig, E. Alexiou, T. A. Fonseca, R. L. de Queiroz, and T. Ebrahimi, "A novel methodology for quality assessment of voxelized point clouds," *Appl. Digit. Image Process. XLI*, vol. 10752, Sep. 2018, Art. no. 107520I.
[54] E. Alexiou and T. Ebrahimi, "Towards a point cloud structural similarity metric," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2020, pp. 1–6.
[55] A. Collet, M. Chuang, P. Sweeney, D. Gillett, D. Evseev, D. Calabrese, H. Hoppe, A. Kirk, and S. Sullivan, "High-quality streamable free-viewpoint video," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 1–13, Jul. 2015.
[56] H. Li, B. Adams, L. J. Guibas, and M. Pauly, "Robust single-view geometry and motion reconstruction," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 1–10, Dec. 2009.
[57] *Methodology for the Subjective Assessment of the Quality of Television Pictures*, document ITU-R BT.500-11, BT, Recommendation ITU-R, International Telecommunication Union, 2002.
[58] Y.-F. Ou, Z. Ma, T. Liu, and Y. Wang, "Perceptual quality assessment of video considering both frame rate and quantization artifacts," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 3, pp. 286–298, Mar. 2011.
[59] Z. Wang and A. C. Bovik, "Modern image quality assessment," *Synth. Lectures Image, Video, Multimedia Process.*, vol. 2, no. 1, pp. 1–156, Jan. 2006.
[60] Q. Wu, H. Li, F. Meng, and K. N. Ngan, "A perceptually weighted rank correlation indicator for objective image quality assessment," *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2499–2513, May 2018.
[61] *VQMT: Video Quality Measurement Tool*. Accessed: Jul. 14, 2020. [Online]. Available: https://mmspg.epfl.ch/vqmt
[62] W. Lin and L. Dong, "Adaptive downsampling to improve image compression at low bit rates," *IEEE Trans. Image Process.*, vol. 15, no. 9, pp. 2513–2521, Sep. 2006.
[63] A. M. Bruckstein, M. Elad, and R. Kimmel, "Down-scaling for better transform compression," *IEEE Trans. Image Process.*, vol. 12, no. 9, pp. 1132–1144, Sep. 2003.
[64] Y. Zhu, G. Zhai, and X. Min, "The prediction of head and eye movement for 360 degree images," *Signal Process., Image Commun.*, vol. 69, pp. 15–25, Nov. 2018.
[65] Y. Zhu, G. Zhai, X. Min, and J. Zhou, "The prediction of saliency map for head and eye movements in 360 degree images," *IEEE Trans. Multimedia*, vol. 22, no. 9, pp. 2331–2344, Sep. 2020.

KEMING CAO (Student Member, IEEE) received the B.E. degree in electronic engineering from Tsinghua University, Beijing, China, in 2014, and the M.S. degree in electrical and computer engineering from the University of California at San Diego, La Jolla, in 2017, where he is currently pursuing the Ph.D. degree in electrical and computer engineering. His research interests include the areas of computer vision, machine learning, image processing, and point cloud compression. He was a recipient of the UC San Diego Department of Electrical and Computer Engineering Fellowship, from 2014 to 2015.

YI XU received the bachelor's degree in electrical engineering from Tsinghua University, Beijing, in 2013, and the master's degree in electrical and computer engineering from Cornell University, Ithaca, in 2016. After that, he quit the Ph.D. Program in Cornell University and went back to Beijing to join a startup company, Owlii Inc., as a Tech Lead. In 2019, the startup was acquired by Kwai Inc., Beijing, where he is currently as a Computer Vision Researcher. His research interests include computer vision, computer graphics, and information theory.

PAMELA COSMAN (Fellow, IEEE) received the B.S. degree (Hons.) in electrical engineering from the California Institute of Technology, in 1987, and the Ph.D. degree in electrical engineering from Stanford University, in 1993.

Following an NSF Postdoctoral Fellowship with Stanford University and with the University of Minnesota (1993–1995), she joined the Faculty of the Department of Electrical and Computer Engineering, University of California at San Diego, where she is currently a Professor. Her research interests include the areas of image and video compression and processing, and wireless communications. She has written over 250 technical articles in these fields, and one children's book, *The Secret Code Menace*, that introduces error correction coding through a fictional story.

Dr. Cosman is a member of Tau Beta Pi and Sigma Xi. Her awards include the ECE Departmental Graduate Teaching Award, the Career Award from the National Science Foundation, the GLOBECOM 2008 Best Paper Award, the HISB 2012 Best Poster Award, the 2016 UC San Diego Affirmative Action and Diversity Award, the 2017 Athena Pinnacle Award (Individual in Education), the 2019 National Diversity Award from the Electrical and Computer Engineering Department Heads Association, and the Qualcomm Faculty Awards in 2019 and 2020. Her administrative positions include the Director of the Center for Wireless Communications, from 2006 to 2008, the ECE Department Vice Chair, from 2011 to 2014, and the Associate Dean for Students, from 2013 to 2016. She has been a member of the Technical Program Committee or the Organizing Committee for numerous conferences, including most recently serving as the Technical Program Co-Chair of ICME 2018. She was an Associate Editor of the IEEE COMMUNICATIONS LETTERS, from 1998 to 2001, and the IEEE SIGNAL PROCESSING LETTERS, from 2001 to 2005. She was the Editor-in-Chief (2006–2009) and a Senior Editor (2003–2005 and 2010–2013) of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS.