BPQA: A BLIND POINT CLOUD QUALITY ASSESSMENT METHOD

Qingyang Zhou [1], Aolin Feng [1], Tsung-Shan Yang [1], Shan Liu [2], C.-C. Jay Kuo [1]

University of Southern California, Los Angeles, California, USA[1]
Tencent Media Lab, Palo Alto, California, USA[2]

2023 IEEE International Conference on Image Processing Challenges and Workshops (ICIPCW) | 979-8-3503-0258-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICIPC59416.2023.10328342

ABSTRACT

A point cloud quality assessment method, called the Blind Point Cloud Quality Assessment (BPQA), is proposed to evaluate the perceptual quality of point clouds compressed by various point cloud codecs. BPQA consists of three modules. First, it selects points of various salience degrees based on the color information. Second, it projects the local neighborhood of selected points along one of the three orthogonal axes to yield a five-channel map (namely, RGB, depth, and pairwise-point-distance-mean channels). Third, it extracts features using the channel-wise Saab transform (c/w Saab) and the relevant feature test (RFT) and trains an XGBoost regressor to predict the Mean Opinion Score (MOS). BPQA offers competitive performance in no-reference quality assessment tasks of the ICIP 2023 PCVQA Chanllenge.

***Index Terms*—** Point cloud, point cloud quality assessment, non-referenced quality assessment

1. INTRODUCTION

Point clouds are widely used in various fields such as computer-aided design, virtual reality, and autonomous driving. These applications require effective quality assessment metrics. The study of no-reference point cloud quality assessment has received considerable attention in recent years. Quite a few methods have been proposed to model the characteristics of the Human Visual System (HVS). However, most of them are not specifically designed for compression distortion evaluation. The emergence of new point cloud codecs poses a challenge in developing effective quality assessment metrics of high accuracy and low complexity.

In this work, we study the problem of compression distortion assessment caused by point cloud codecs, and propose a no-reference point cloud quality assessment method. It is named the Blind Point Cloud Quality Assessment (BPQA) method. BPQA consists of three modules: 1) salient points selection, 2) 3D-to-2D projection to generate multi-channel maps, and 3) map representation learning based on the channel-wise Saab (c/w Saab) transform, feature extraction via the relevant feature test (RFT) [1], and XGBoost regression for quality scores, respectively. Experimental results show that BPQA offers competitive performance in two tasks of the ICIP 2023 PCVQA Challenge.

The authors acknowledge the gift support from the Tencent Media Lab as well as the Center for Advanced Research Computing (CARC) at the University of Southern California for providing computing resources that have contributed to the research results reported within this publication. URL: https://carc.usc.edu.

2. REVIEW OF PREVIOUS WORK

The point cloud quality assessment tasks can be categorized into the full-reference and no-reference two types depending on whether the original point cloud is available as a reference. In the absence of a reference, powerful features and effective learning models are critical. The characteristics of the human visual system (HVS) have been leveraged to derive good features and models. For example, the visual masking effect on point cloud quality assessment has been investigated in [2, 3, 4]. The distortions perceived by HVS is different from the physical characteristics in local regions. Torlig *et al.* [5] showed the similarity between the 3D-to-2D projection and the HVS functionality.

Based on the HVS study, quite a few learning-based quality metrics have been developed. For instance, Chetouani *et al.* [6] utilized a neural network to learn low-level features for quality scoring. Tao *et al.* [7] proposed a point cloud projection and multiscale feature fusion network. Liu *et al.* [8] proposed a PCQA network comprising a feature extraction/fusion module, a distortion type identification module, and a quality prediction module sequentially. Zhang *et al.* [9] employed 3D natural scene statistics and entropy as features and used an SVM model to predict quality scores. Damme *et al.* [10] used KNN clustering and prediction techniques combined with pixel-based features. Recently, a sparse CNN-based method was proposed in [11] claiming a better performance over some FR metrics.

Among existing point cloud quality assessment work, only a few have been tailored to compression distortion evaluation. With the growing interest in point cloud coding, standardized point cloud codecs such as V-PCC and G-PCC [12, 13, 14, 15] and non-standardized ones like Geo-CNN [16, 17], Sparse PCGC [18, 19, 20], and Green PCGC [21], have been developed in recent years. Because of the emerging trend, the assessment of point cloud compression distortion has become increasingly important. Additionally, the quality assessment model should have low complexity so that the metrics can be integrated inside the encoder or decoder for performance optimization in mobile/edge devices. This presents a great challenge on deep-learning-based models. In this work, we leverage the HVS characteristics and the green learning paradigm [22] to develop a blind point cloud quality metric.

3. PROPOSED BPQA METHOD

3.1. System Overview

The proposed BPQA method consists of three modules as shown in Fig. 1: 1) saliency-based point sampling, 2) multi-channel map generation, and 3) representation learning, feature extraction and regression. In the first module, the color saliency of each point in a point cloud scan is computed. Based on their saliency values, we

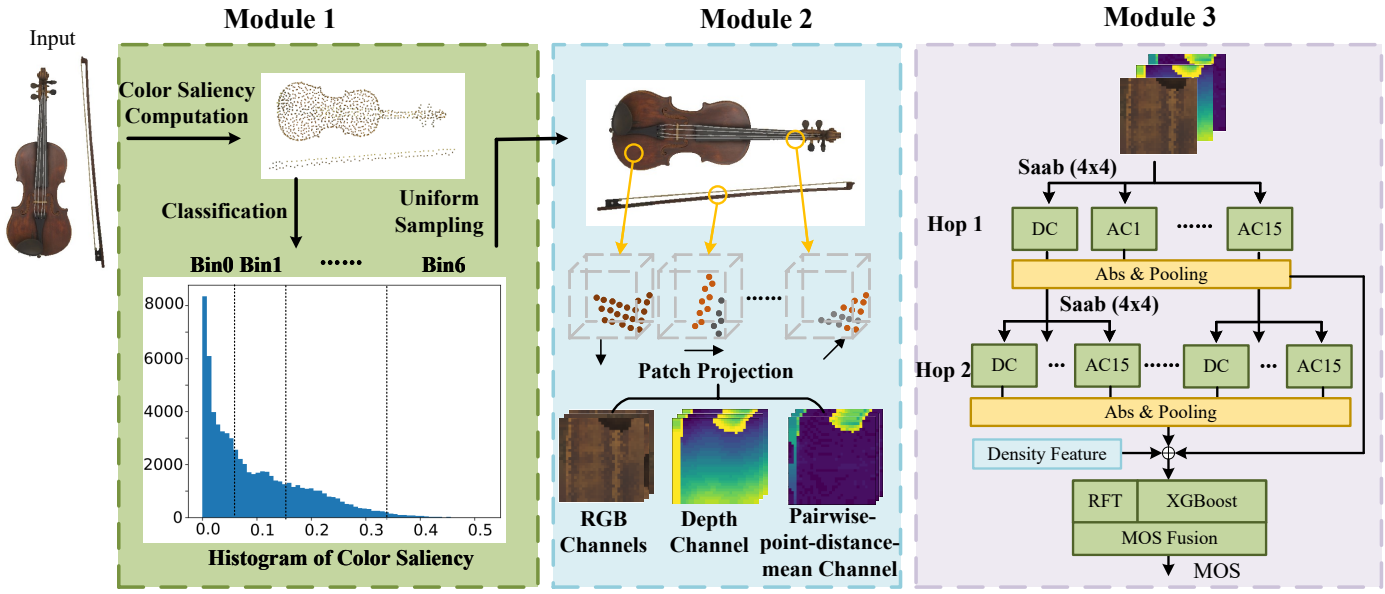

Fig. 1. An overview of the proposed BPQA method. The saliency-based point sampling is presented in Module 1. The 3D-to-2D patch projection yields multi-channel maps in Module 2. Representation learning, feature extraction and regression is presented in Module 3.

classify points into seven bins. We sample points uniformly from the 7 bins to get a set of points for further processing. In the second module, a local patch is constructed based on the neighborhood of each selected point from Module 1. Then, a 3D-to-2D projection along one of the $x-, y-, z-$ axes is conducted for the patch, which yields a 5-channel map, including the RGB values, the depth value and the pairwise-point-distance-mean value. In the third module, a new representation is obtained by a 2-hop cw-Saab[23, 24], powerful features are extracted based on RFT [1], and an XGBoost regressor is trained to predict the Mean Opinion Score (MOS) for each patch. Finally, the MOS of all patches are fused to yield the final MOS score of the whole point cloud. These three modules are elaborated in Sec. 3.2, Sec. 3.3, and Sec. 3.4, respectively.

3.2. Module 1: Saliency-based Point Sampling

To lower the computational complexity, we sample representative points from the full point cloud scan. An input point cloud is first voxelized and evenly downsampled. A voxel point takes the average value of points in a neighborhood of dimension $s \times s \times s$. In this downsampled voxel representation, representative voxel points are further sampled based on their color saliency. This idea is borrowed from the saliency calculation method proposed in [25]. It evaluates the deviation of the attribute value of a target point from the averaged attribute value of its neighborhood. It yields a color saliency score that is independent of the point cloud geometry. However, the use of high-saliency points alone is not sufficient. Instead, a combination of points of different saliency values gives better results. Here, we develop a bin-wise sampling strategy based on saliency scores.

We partition saliency scores into 7 bins with the following steps: 1) sorting all voxel points in the training set with their saliency scores; 2) evenly dividing sorted saliency scores into 7 bins, 3) randomly sampling the same number of voxel points in each bin. To ensure the quality and representative capability of sampled voxel points, two more sampling criteria are imposed. First, if the number of neighboring points of the sampled one is less than a threshold, this point is discarded. Second, if a sampled point is too close to another one (e.g. the Euclidean distance is smaller than a threshold), the second sample is also excluded.

Assisted by the two criteria, we sample n voxel points from each bin, and a total of $7n$ representative voxel points are sampled for the raw point cloud scan. The saliency scores of voxel points in a point cloud scan typically follow an exponential-like distribution as shown in Figure 1. The great majority of points have low saliency scores. Given a budget of the total sampling number, voxel points of higher saliency have a higher likelihood of being selected under bin-wise resampling. This sampling strategy is better than a uniform sampling on points of the whole point cloud as presented in Sec. 4.

3.3. Module 2: Multi-channel Map Generation

Voxel points and points of the raw point cloud, called raw points, have different scales. A voxel point can be empty or it can contain multiple raw points. For this reason, we need to find a corresponding raw point with respect to each sampled voxel point. Suppose the coordinates of a sampled voxel point are (x, y, z). We find a point in the raw point cloud that has the nearest Euclidean distance to (x, y, z). Its coordinates are denoted by $(\tilde{x}, \tilde{y}, \tilde{z})$. Then, we use $(\tilde{x}, \tilde{y}, \tilde{z})$ as the center and get all raw points with a bounding cube of dimension $d \times d \times d$. We perform an orthogonal projection inside the cube along x, y, and z-axes, respectively, and choose the one that has the maximum projection area. To measure the projection area, we first discretize the 2D projected plane with a uniform grid consisting of $l \times l = l^2$ blocks. Then, the projection area is defined to be the number of non-empty blocks. This idea was first proposed in [21].

The projected 2D plane produces a 2D grid of size $l \times l$. Then, we define a five-channel map on this grid. The five channels are three color channels, one depth channel, and one pairwise-point-distance-mean channel as shown in Figure 1. The five-channel map is used

to represent the local color and the geometry information of a point cloud. When a block has only one projected point, the RGB value of the block is directly copied from that point. If a block has two or more projected points, the point closest to the projection plane is selected, where the projection plane is the plane passing through the cube centroid with its normal vector parallel to the projection direction. If a block has no projected points, it is marked as null. A similar operation is performed to yield the depth channel. Finally, a pairwise-point-distance-mean map is generated for all points inside the same 3D cube. To remove an isolated null value, we conduct the bilinear interpolation on each channel based on its non-null neighbors as a post-processing step to yield a smooth map.

3.4. Module 3: Representation Learning, Feature Extraction and Regression

The third module is a data-driven learning process. We adopt the green learning paradigm [22] due to its low complexity and small model size advantages. It contains three steps: unsupervised representation learning, semi-supervised feature learning, and supervised machine learning. They are elaborated below.

a) Unsupervised representation learning. It consists of two cascaded stages called two hops, where Hop 1 and Hop 2 are designed to capture both local and global representations, respectively, as depicted in Figure 1 The channel-wise Saab transform (cw-Saab) [24] is performed in each hop. In Hop 1, the input multi-channel map is partitioned into non-overlapping blocks of size 4×4, where the Saab transform is performed to obtain the Saab coefficients. Each channel is processed separately, and the Saab coefficients are fed into a data aggregation layer that takes the absolute value and then uses the maximum pooling to reduce the feature dimension. Then, a similar process is repeated in Hop 2 to enlarge the receptive field. Apart from the Saab coefficients, we use other operations to generate two additional sets of representations. First, we compute the maximum, mean and standard deviation of the same coefficient across the spatial domain. They provide global presentations of the whole point cloud. Second, we compute the mean value of the point-wise shortest Euclidean distances in each 3D cube. Saab coefficients of five channels in Hop 1 and Hop 2 and these two additional sets are concatenated to form a long representation vector. Note that no labels are needed in this step.

b) Semi-supervised feature learning. We use the relevant feature test (RFT) [1] to evaluate the discriminating power of each feature. Basically, it partitions each representation, which is a scalar, into two regions at a set of discrete points that are chosen from the interval between the maximum and the minimum of the representation. For each partition, we calculate the mean value of samples in each region and use it to compute the mean-squared error (MSE) associated with each region. Then, we choose the partition point that minimizes the weighted MSE value. The corresponding MSE value is called the RFT cost of that representation. If a representation has a smaller RFT cost, it has a better approximation capability to reduce the regression error. Then, we sort all representations based on their RFT loss values from the smallest to the largest as shown in Figure 2, where the x-axis indicates the sorted representation index and the y-axis is the RFT loss value. Each curve indicates the RFT behavior of one channel. We see that all curves have a distinct elbow point. It means that representation before the elbow should be selected as features. This process can be applied to a subset of training samples. Thus, it is called the semi-supervised feature extraction.

c) Supervised decision learning. The Mean Opinion Score (MOS) prediction of each cube is implemented by the XGBoost regressor [26]. It maps a feature vector to a MOS value. If we sample a total of k cubes from a point cloud scan, we obtain k MOS values per object. We conduct a simple average of these k values to get the predicted MOS value for the point cloud scan. We need all training samples and their labels in this step to get better performance. Thus, it is fully supervised.

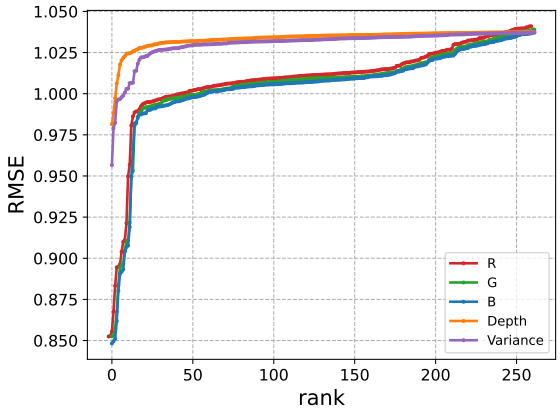

Fig. 2. The RFT curves of sorted representations for five channels. The majority of quality-aware representations come from RGB channels. The first few representations from the depth map and the distance map also contribute to the quality score prediction.

4. EXPERIMENTS

Experimental Setup. The down-sampling voxel size $d \times d \times d$ is set to $8 \times 8 \times 8$. The color saliency score is computed in a KNN neighborhood with $K = 64$. We select $n = 40$ representative points in each bin. The 3D cube and the 2D projected maps are set to $32 \times 32 \times 32$ and 32×32, respectively. In Module 3, the Saab kernel size is 4×4 in both hops, with a maximum-pooling window size of 2×2. For the XGBoost, the tree number is 1500, and the depth is 5. We use the training set from the Broad Quality Assessment of Static Point Clouds (BASICS) [27] provided by the ICIP PCVQA Challenge organizer. It contains 898 lossy-coded point cloud objects reconstructed from 45 static point cloud objects encoded by four point cloud codecs under different bitrates.

Performance Analysis. The performance is evaluated by the Pearson Linear Correlation Coefficient (PLCC), the Spearman Rank Order Correlation Coefficient (SROCC), Difference/Similar Analysis quantified by Area Under the Curve (D/S AUC), and Better/Worse Analysis quantified by Correct Classification percentage (B/W CC)[28]. PLCC is A measurement for the linear correlation between predicted scores and subjective quality scores defined as

$$PLCC = 1 - \frac{\sum_i (p_i - p_m)(\hat{p_i} - \hat{p_m})}{\sqrt{\sum_i (p_i - p_m)^2}\sqrt{\sum_i (\hat{p_i} - \hat{p_m})^2}}, \quad (1)$$

where, p_i and $\hat{p_i}$ refer to the predicted score and the subjective quality score, respectively, while p_m and $\hat{p_m}$ denote the mean of the predicted score and the subjective quality score, respectively. SROCC is adopted to measure the monotonicity between predicted and subjective quality scores, which are defined as

$$SROCC = 1 - \frac{6\sum_{i=1}^{L}(m_i - n_i)^2}{L(L^2 - 1)}, \quad (2)$$

Track 2							
Team Name	SROCC	PLCC	D/S AUC	B/W CC	Runtime	Final Points (20 Max)	Final Ranking
ZZhang - SJTU MMLAB	0.8806	0.9076	0.8481	0.9626	16.10	18	1
QZhou - Q&A	**0.7933**	**0.8038**	**0.7878**	**0.9079**	**27.7**	**12**	**2**
RWatanabe - KDDIUSCJoint	0.7595	0.7950	0.7317	0.8911	11.53	11	3
OMessai - Ecole des Mines	0.5473	0.5883	0.6554	0.7764	5.53	8	4
YZhang - SlowHand	0.3901	0.5136	0.6232	0.7009	16.37	1	5

Track 4							
Team Name	SROCC	PLCC	D/S AUC	B/W CC	Runtime	Final Points (20 Max)	Final Ranking
ZZhang - SJTU MMLAB	0.6352	0.6103	0.6782	0.9141	16.10	18	1
QZhou - Q&A	**0.5526**	**0.4064**	**0.625**	**0.8691**	**27.70**	**12**	**2**
RWatanabe - KDDIUSCJoint	0.444	0.4167	0.579	0.7959	11.53	12	2
OMessai - Ecole des Mines	0.2761	0.1458	0.4939	0.6744	5.53	8	4
YZhang - SlowHand	0.0958	0.1065	0.4951	0.5569	16.37	2	5

Table 1. The scores on Tracks 2 and 4 in the challenge, where our method is given in boldface.

Methods	PLCC	SROCC
Random sampling	0.875	0.814
Lowest saliency selection	0.879	0.825
Highest saliency selection	0.864	0.789
Saliency-based sampling	**0.885**	**0.837**

Table 2. Comparison of different representative point selection schemes in Module 1.

Methods	PLCC	SROCC
RGB	0.860	0.789
RGB+depth	0.873	0.805
RGB+pairwise distance mean	0.874	0.814
RGB+depth+pairwise distance mean	**0.885**	**0.837**

Table 3. Comparison of different choices of multi-channel maps in Module 2.

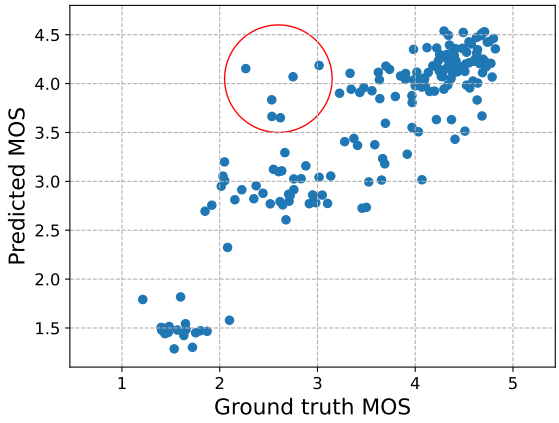

Fig. 3. Predicted MOS vs. ground truth MOS. Data points in the red circle indicate the failure cases, where our model overestimates the quality of some GEO-CNN-compressed [16] point clouds.

where m_i represents the rank of p_i in the predicted scores, n_i represents the rank of $\hat{p}_i$ in the subjective quality score and L is the number of images.

In the ICIP PCVQA Challenge, our method was benchmarked with other submissions in Track 2 (no-reference, broad-range quality estimation) and Track 4 (no-reference, high-quality range). Our method gives a competitive performance in both. For Track 2, our model has the second-highest scores in PLCC, SROCC, D/S AUC, and B/W CC. In Track 4, it achieves the second-highest scores in SROCC, D/S AUC, and B/W CC, and third in PLCC. The complexity of our method is dominated by saliency score computation. If we can replace the saliency computation with some other faster ones, the runtime of our model can be much reduced. Furthermore, our model was developed purely on the CPU platform. No optimization on the GPU platform is implemented yet.

Discussion on Failure Cases. Several cases with significant over-estimated point cloud quality scores are circled in Figure 3. They are samples coming from a deep-learning-based codec called GEO-CNN at lower bitrates. The codec tends to distort high-saliency regions by twisting and breaking textures apart. For example, the strings and bow of the "violin" object in Figure 1 are distorted into segmented pieces. Local cube-based metrics cannot estimate the distortion well without proper global information. We will find better solutions for these failure cases in our future work.

Ablation Study. To assess the effectiveness of different components of our method, we conduct ablation studies on the validation set. The sampling scheme in Module 1 is evaluated by comparing four schemes: the proposed saliency-based sampling, random sampling, using $7n$ points with the highest and the lowest saliency scores. Table 2 shows that the saliency-based sampling scheme offers the highest PLCC and SROCC scores. The contribution of multiple channels for map representation is also evaluated in Module 2. Different channel combinations are compared in Table 3. We see the importance of depth and pairwise-point-distance-mean maps.

5. CONCLUSION

A no-reference point cloud quality assessment method, called Blind Point Cloud Quality Assessment (BPQA), was proposed in this work. The method achieves competitive results in the ICIP PCVQA Challenge. In the future work, we would like to improve its prediction performance, computational complexity, and model size furthermore.

6. REFERENCES

[1] Y. Yang, W. Wang, H. Fu, and C. C. J. Kuo, "On supervised feature selection from high dimensional feature spaces," 2022.

[2] A. Javaheri, C. Brites, F. Pereira, and J. Ascenso, "Point cloud rendering after coding: Impacts on subjective and objective quality," *IEEE Transactions on Multimedia*, vol. 23, pp. 4049–4064, 2020.

[3] Q. Yang, H. Chen, Z. Ma, Y. Xu, R. Tang, and J. Sun, "Predicting the perceptual quality of point cloud: A 3d-to-2d projection-based exploration," *IEEE Transactions on Multimedia*, vol. 23, pp. 3877–3891, 2021.

[4] G. Meynet, Y. Nehmé, J. Digne, and G. Lavoué, "Pcqm: A full-reference quality metric for colored 3d point clouds," in *2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*, 2020, pp. 1–6.

[5] E. Torlig, E. Alexiou, T. Fonseca, R. De Queiroz, and T. Ebrahimi, "A novel methodology for quality assessment of voxelized point clouds," 09 2018.

[6] A. Chetouani, M. Quach, G. Valenzise, and F. Dufaux, "Deep learning-based quality assessment of 3d point clouds without reference," in *2021 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, 2021, pp. 1–6.

[7] W.-x. Tao, G.-y. Jiang, Z.-d. Jiang, and M. Yu, "Point cloud projection and multi-scale feature fusion network based blind quality assessment for colored point clouds," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 5266–5272.

[8] Q. Liu, H. Yuan, H. Su, H. Liu, Y. Wang, H. Yang, and J. Hou, "Pqa-net: Deep no reference point cloud quality assessment via multi-view projection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 12, pp. 4645–4660, 2021.

[9] Z. Zhang, W. Sun, X. Min, T. Wang, W. Lu, and G. Zhai, "No-reference quality assessment for 3d colored point cloud and mesh models," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 11, pp. 7618–7631, 2022.

[10] S. Van Damme, M. T. Vega, J. van der Hooft, and F. De Turck, "Clustering-based psychometric no-reference quality model for point cloud video," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 1866–1870.

[11] Y. Liu, Q. Yang, Y. Xu, and L. Yang, "Point cloud quality assessment: Dataset construction and learning-based no-reference metric," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 19, no. 2s, feb 2023.

[12] D. Graziosi, O. Nakagami, S. Kuma, A. Zaghetto, T. Suzuki, and A. Tabatabai, "An overview of ongoing point cloud compression standardization activities: Video-based (v-pcc) and geometry-based (g-pcc)," *APSIPA Transactions on Signal and Information Processing*, vol. 9, p. e13, 2020.

[13] 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.

[14] MPEG, "V-pcc codec description," *ISO/IEC JTC 1/SC 29/WG 7 N00100*, 2020.

[15] MPEG, "G-pcc codec description v12," *ISO/IEC JTC 1/SC 29/WG 7 N00151*, 2021.

[16] M. Quach, G. Valenzise, and F. Dufaux, "Improved deep point cloud geometry compression," in *2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP)*, 2020, pp. 1–6.

[17] ——, "Learning convolutional transforms for lossy point cloud geometry compression," in *2019 IEEE international conference on image processing (ICIP)*. IEEE, 2019, pp. 4320–4324.

[18] J. Wang, D. Ding, Z. Li, X. Feng, C. Cao, and Z. Ma, "Sparse tensor-based multiscale representation for point cloud geometry compression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–18, 2022.

[19] J. Wang, D. Ding, Z. Li, and Z. Ma, "Multiscale point cloud geometry compression," in *2021 Data Compression Conference (DCC)*, 2021, pp. 73–82.

[20] J. Wang, H. Zhu, Z. Ma, T. Chen, H. Liu, and Q. Shen, "Learned point cloud geometry compression," *arXiv preprint arXiv:1909.12037*, 2019.

[21] Q. Zhou, S. Liu, and C.-C. J. Kuo, "Gpcgc: A green point cloud geometry coding method," 2023. [Online]. Available: https://arxiv.org/abs/2302.06062

[22] C.-C. J. Kuo and A. M. Madni, "Green learning: Introduction, examples and outlook," *Journal of Visual Communication and Image Representation*, p. 103685, 2022.

[23] C.-C. J. Kuo, M. Zhang, S. Li, J. Duan, and Y. Chen, "Interpretable convolutional neural networks via feedforward design," *Journal of Visual Communication and Image Representation*, vol. 60, pp. 346–359, 2019.

[24] Y. Chen, M. Rouhsedaghat, S. You, R. Rao, and C.-C. J. Kuo, "Pixelhop++: A small successive-subspace-learning-based (ssl-based) model for image classification," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 3294–3298.

[25] S. Chen, D. Tian, C. Feng, A. Vetro, and J. Kovačević, "Fast resampling of three-dimensional point clouds via graphs," *IEEE Transactions on Signal Processing*, vol. 66, no. 3, pp. 666–681, 2018.

[26] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 785–794. [Online]. Available: https://doi.org/10.1145/2939672.2939785

[27] A. Ak, E. Zerman, M. Quach, A. Chetouani, A. Smolic, G. Valenzise, and P. L. Callet, "Basics: Broad quality assessment of static point clouds in compression scenarios," *ArXiv*, vol. abs/2302.04796, 2023.

[28] L. Krasula, K. Fliegel, P. Le Callet, and M. Klíma, "On the accuracy of objective image and video quality models: New methodology for performance evaluation," in *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, 2016, pp. 1–6.