https://doi.org/10.2352/ISSN.2470-1173.2021.9.IQSP-256

A NOVEL POINT CLOUD QUALITY ASSESSMENT METRIC BASED ON PERCEPTUAL COLOR DISTANCE PATTERNS

Rafael Diniz, Pedro Garcia Freitas, Mylène Farias

University of Brasília

ABSTRACT

In recent years, PCs have become very popular for a wide range of applications, such as immersive virtual reality scenarios. As a consequence, in the last couple of years, there has been a great effort to develop novel acquisition, representation, compression, and transmission solutions for PC contents in the research community. In particular, the development of objective quality assessment methods that are able to predict the perceptual quality of PCs. In this paper, we present an effective novel method for assessing the quality of PCs, which is based on descriptors that extract perceptual color distance-based texture information of PC contents, called Perceptual Color Distance Patterns (PCDP). In this framework, the statistics of the extracted information are used to model the PC visual quality. Experimental results show that the proposed framework exhibit good and robust performance when compared with several state-of-the-art point cloud quality assessment (PCQA) methods.

Index Terms— Point Cloud; Quality Assessment, Texture Descriptors, Local Binary Patterns, Local Color Pattern

1. INTRODUCTION

PC representations consist of collections of points in a 3D space, with their corresponding positions and attributes that describe surface properties. To accurately describe a 3D scene, PCs require a large number of points, which limits their use in real applications. As a consequence, in the last couple of years there has been a great effort in the research community (both in academia and industry) to develop efficient ways to represent, display, process, compress, and transmit PC contents [1, 2, 3, 4, 5, 6, 7, 8, 9]. Lately, a particular area that has attracted a lot of attention is the quality of PC contents and, in particular, the design of PC quality assessment (PCQA) methods that can automatically predict the perceived quality of PC content. It is worth mentioning that PCQA methods can be used not only to evaluate the quality of decoded, rendered, or transmitted PCs, but also to understand the representation and display requirements needed to achieve a good perceived quality of experience (QoE) for the end-user.

Thanks to CAPES agency for funding.

In recent years, a number of PCQA metrics have been proposed [10]. For example, Meynet *et al.*[11] introduced an adaptation for PCs of metric originally intended for meshes, which uses the curvature disparity to predict the PC quality [12]. Their work shows that curvature statistics have a higher capability to predict quality than pure geometric distance-based metrics. Torlig *et al.* [13] developed a quality metric that maps 3D volumes onto 2D images. Their metric uses orthographic projections in combination with conventional 2D image quality metrics, but it does not exploit the intrinsic 3D structures of PCs. Alexiou and Ebrahimi [14] developed simple metrics to capture the perceived geometric impairments of distorted PCs. Javaheri *et al.* proposed a PCQA method based on the generalized Hausdorff distance, which instead of taking the maximum distance over all the distances considers only the K-lowest distance values [15]. Viola *et al.* proposed a metric that combines color- and geometry-based metrics in order to provide a global quality score. Their metric takes into account the color statistics by analyzing the color histograms and the correlograms [16]. Meynet *et al.* proposed a metric that also takes into consideration geometry- and color-based features, using logistic regression to combine these features and produce a quality estimate [17]. Alexiou et al. [18] provides an interesting analysis of the effects of pre-processing and different rendering methods of PC on the performance of the state-of-the-art PCQA methods. More recently, Alexiou et al. [19] also proposed a PCQA based on local features extraction, with correlation data comparing different metrics with 2 data-sets, presenting promising results.

Most of the work on PCQA are full-reference (FR) proposals, which means a PC reference is used by the PCQA algorithm to assess the quality of the degraded version of a PC. Bello et al.[20] provided a review on the use of deep learning in 3D vision tasks, including classification, segmentation, and detection, while pointing out that local point relationships are more effective for modeling a PC data-driven approach. Liu et al. [21] proposed the first no-reference method (NR) for PCQA which uses a data-driven approach and applies a convolutional neural network (CNN).

In previous works, we have explored the use of texture descriptors to estimate the quality of PC contents, achieving good and promising results [22, 23]. In [22] we propose the joint use of an already available point-based geometry-only metric together with our texture-based color-only proposal in order to improve the PC quality prediction. In this paper, we explore a novel color-based feature extractor which considers a local neighborhood for the computation, and perceptual color distances based on CIELAB color-space distances. The proposed descriptor consider local neighborhood relations based on the perceptual color distances, providing relevant intrinsic information for a given PC, which we use for quality prediction.

The rest of this paper is organized as follows. Section 2 describes the proposed feature extractor. Section 3 describes the experimental setup. Section 4 discusses the performance of the proposed method when compared with state-of-the-art PCQA methods. Finally, Section 5 presents the conclusions.

2. PROPOSED METHOD

The proposed method evolves previous Diniz's works [22, 23, 24] by introducing a novel PC texture descriptor based on perceptual color distances. Figure 1 outlines the proposed framework, which is composed of multiple stages, such as the voxelization stage, the feature extraction stage, the feature distance measurement stage, and the prediction model. The feature extraction stage is divided into two blocks: the descriptor computation and the collection of its statistics. The distance metrics are used to compute the difference between the descriptors' statistics of the reference and test PCs. The distances are then fed to the prediction model to generate the quality score. In the next sections, we describe each step used in the proposed framework.

2.1. Voxelization

The voxelization stage models the geometric scenes into their corresponding discrete volume element (voxel) representations [25]. In other words, it approximates complex scenes with thousands of polygons into discrete 3D voxels, simplifying the task of computer graphics algorithms, such as volumetric rendering or object collision [26]. Voxelizing a 3D PC content consists of discretizing the continuous space points into a volumetric grid, where the parallelepipedic elements are the voxels. This procedure not only decreases the PC complexity but also enables better access and manipulation of the data. Voxels can be either occupied with a color value or can be empty. To voxelize the PC, we first set the voxel size (VS) as given by the following equation:

$$VS = ES^3, \tag{1}$$

where ES is the edge size (ES) of a voxel, which is computed with the following equation:

$$\text{ES} = \frac{k}{Q} \cdot \sum_{n=1}^{Q} \left(\frac{1}{k_{nn}} \cdot \sum_{i=1}^{k_{nn}} \mathbf{d}\left(N_i(P_n), P_n\right) \right), \tag{2}$$

where Q is the number of points of the PC, k is an ES multiplier (which we tested with different values), P_n is the n-th point of the PC, $N_i(P_n)$ are the coordinates of the i-th point nearest neighbor to P_n, and k_{nn} is the number of nearest neighbors considered for the voxel size calculation, which is 8 in our case (in a resemblance to 2D surface neighborhood shape). The function $\mathbf{d}\left(P_a, P_b\right)$ returns the Euclidean distance between points P_i and P_j.

It is worth pointing out that the VS value in Eq. 1 is computed independently for each PC. Also, when the voxelization is applied, given a voxel size, more than one PC point might be present inside the voxel, in this case, the points' color are averaged to provide the final color value for the voxel.

Figure 2 illustrates the effect of voxelization on the visualized PCs. In this figure, the first column corresponds to the original (not voxelized) PCs, and three levels of compression (a simple downsampling in the example), from low to high - low meaning low quality (strong downsampling), to high, meaning high quality (little downsampling). Also, the PC points in the first column are rendered with appropriate splatter size not to present holes, while the other six columns correspond to PCs voxelized using different voxel sizes. The first row corresponds to the uncompressed PC, while the other three rows correspond to compressed PCs with high-, medium-, and low-quality levels, respectively. Notice that the quality of the voxelized PCs depends on the compression level. For instance, when no voxelization is employed (1st column) the uncompressed and compressed PCs have more subtle perceived differences. However, for the voxelized versions in the second column, the impairment levels of the different compression levels are quite different. Moreover, for larger voxels (e.g. see the last 4 columns) the perceived quality differences seem to decrease again.

2.2. Feature Extraction

The feature extraction stage consists of computing the descriptor, which will be later used to estimate the PC quality. Here, the basic assumption is that visual impairments affect the outputs of the descriptor and, therefore, we can use their statistics to estimate the overall visual quality. Before computing the descriptors, the PC is voxelized as described in Section 2.1. Then, figure 3 depicts the next steps of our proposal: the PC points are converted from RGB to CIELAB colospace, in order we can obtain perceptual color distances between each point (in the picture, the first point in the list) and its neighbors, by using the CIEDE2000 distance [27], which is the latest and most accurate distance metric for mapping perceptual color differences in a linear way. The Perceptual Color Distance Patterns (PCDP) feature extraction works by obtaining the CIEDE2000 distance C for N neighbors of each point in a PC, and then using the distances C of each neighbor to create a label L, by using the algorithm in Equation 3, which is applied in a iterative manner. We tested with

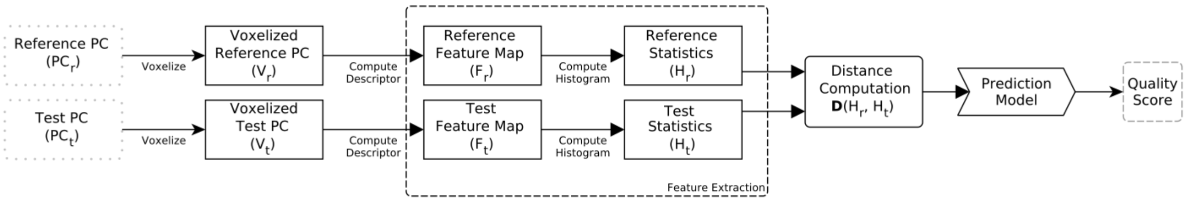

Fig. 1. Block diagram of the proposed framework for designing point cloud full-reference metrics.

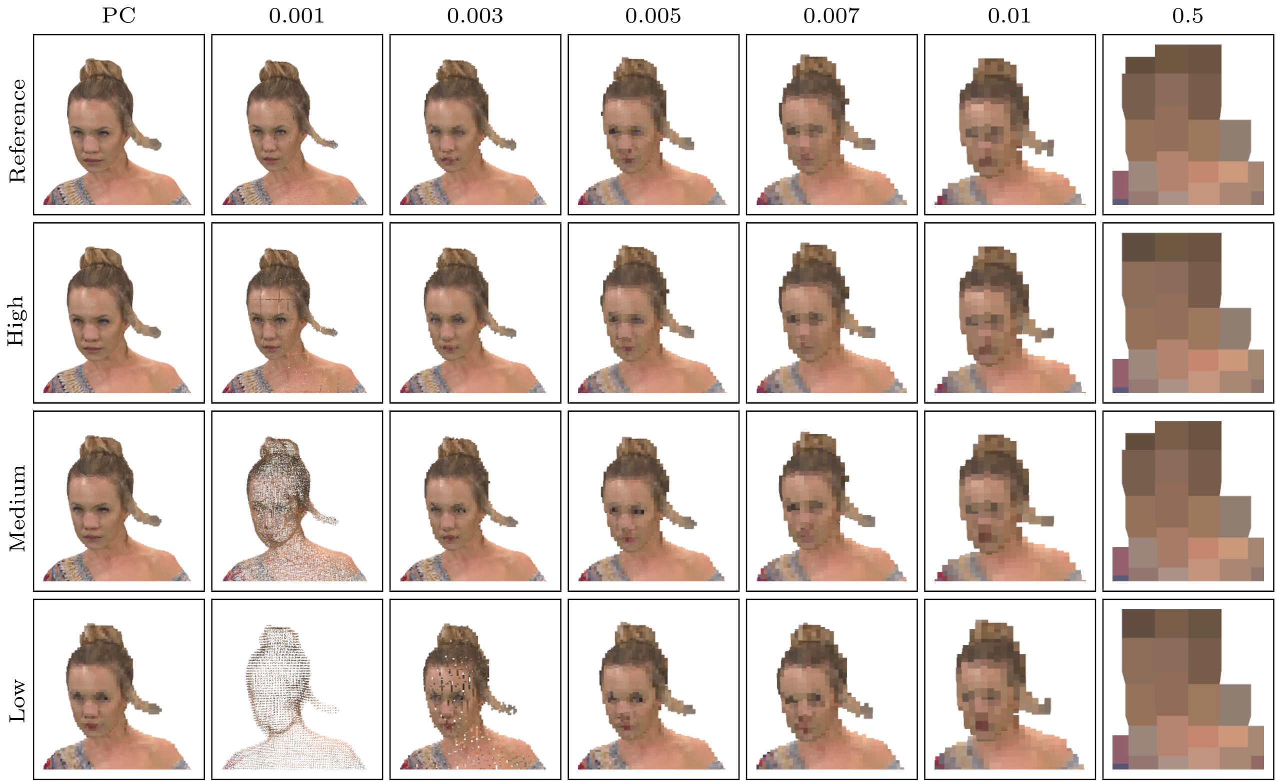

Fig. 2. The visual effect of the voxel size on the pristine reference of Long Dress PC and its impaired versions.

varying number of neighbours (N), and opted to use 12, as it provided good results. The label size adopted was 8 bits, which was validated independently to other variables, to be a good performing option.

$$L = \begin{cases} L \vee (1 \ll \lfloor \frac{C[i]-2.5}{2.5} \rfloor), & \text{if } 2.5 \leq C[i] < 20.0; \\ L \vee (1 \ll 7), & \text{if } C[i] \geq 20. \end{cases} \quad (3)$$

where the symbol $\vee$ is a bitwise OR and $\ll$ is a bitwise left shift. After all neighbors are analyzed, a final 8-bits label L is obtained, as shown in figure 3, through a real example.

In Figure 3 (for simplicity, shown with just 8 neighbors), for the given target voxel we show the set of selected neighboring voxels. The CIEDE2000 distances, shown as "delta-E" in the picture, is then used to set a bit (or not, if delta-E is less than 2.5) in the PCDP label. The rationale of the PCDP is that smaller perceptual color distance among a neighbor will set a less significant bit, while large perceptual color distance will set a higher significant bit, and a very small difference (approximately smaller than the Just Noticeable Difference threshold, around 2.5), will not alter the label.

2.2.1. *Statistics Computation*

We extract the statistical information H from the labels created by the PCDP by computing its histograms:

$$H = \{h(\ell_0), h(\ell_1), h(\ell_2), h(\ell_3), \cdots, h(\ell_{255})\}, \quad (4)$$

where H represents the histogram and $h(\ell_k)$ is the frequency (or empirical probability) of the label $h(\ell_k)$ among all PC points. In our case, k ranges from 0 to 255, as it is 8 bits.

The histograms are used as feature vectors to compute the PC quality. In our proposal, the histograms are computed for the reference (pristine) and test (possibly impaired) PCs.

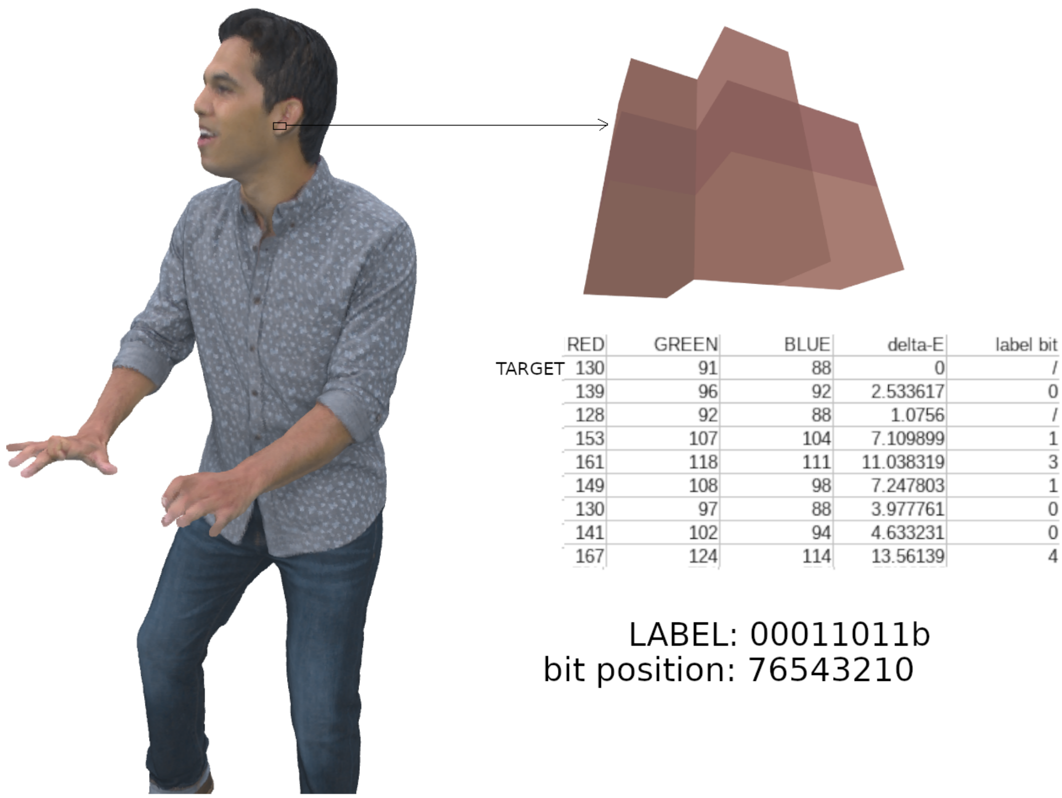

Fig. 3. Graphic schematic of the PCDP.

The motivation to collect the above statistics is because it is non-trivial to find the corresponding voxels from the reference and test PCs. By computing the reference and test histograms independently, we can reduce the difficulties of comparing texture information of a PC to a problem of comparing distributions, which can be done using statistical distances.

2.3. Prediction Model

The first steps of the quality prediction model proposed in this framework are the voxelization, feature extraction, histogram calculation and, as a full-reference proposal, histogram distances calculation. In order to calculate the histogram distances, many metrics are available, and after all these steps, a regression function is used to improve the correlation of the obtained distances with the ground truth data which is available for the selected data-sets used for comparison to other PCQA proposals. The prediction model used in this work is depicted in Figure 4, which contains the representation of the statistics of the reference and test PC features, histogram distance calculation and the regression model, which outputs the final quality score of this proposal. The histograms of the reference PC (H_r) and test PC (H_t) are compared using a distance metric $\mathbf{D} = D(h_r, h_i)$. To compare histograms, we can use several metrics, such as L_1 (Cityblock), L_2 (Euclidean), L_∞ (Chebyshev), f-divergences (e.g., Hellinger, Bhattacharyya, Kullback–Leibler, Jensen-Shannon [28], χ^2) [29], or the Wasserstein distance [30]. The prediction model uses a regression algorithm that takes as input the histogram distances $\mathbf{D}$ and maps them into a predicted quality score. The regression method used in this work is the Logistic function, as described by ITU [31]. The Logistic function, as discussed in previous work [24] was used fit the experimental data from the PCDP to the subjective scores, as it well matches the perceptual behavior of the human vision system.

3. EXPERIMENTAL SETUP & PROTOCOLS

In our tests, we use the following datasets, which contain subjective scores collected from psychophysical experiments [13, 32, 5, 33].

- D_1 [13]: This database has 6 reference PCs, which include human bodies and inanimate objects. It includes 54 test PCs, impaired at 9 distortion levels. Distortions were produced using an octree-based codec, with color attributes encoded using the JPEG algorithm. Subjective scores were obtained in two different universities (UnB & EPFL), and we use the arithmetic average of both scores in the comparisons.
- D_2 [32]: This database has 6 reference PCs, which

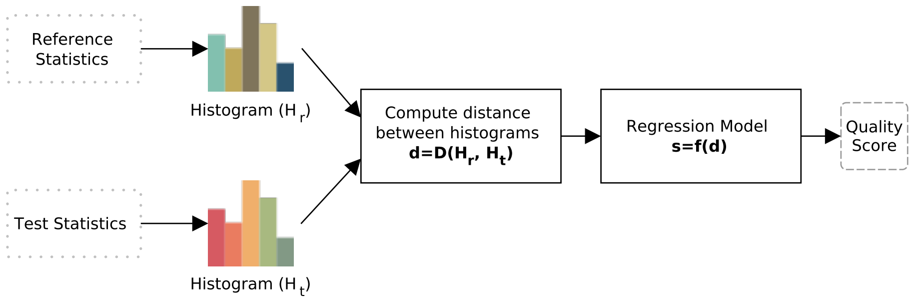

Fig. 4. Block diagram of the regression used in the prediction model in the proposed framework.

include human bodies, inanimate objects, and large-scale PCs. It includes 36 test PCs with 2 categories of distortions. Specifically, the distortions were obtained with the projection-based encoder (3DTK) and the octree-pruning (PCL) with three distortion levels (high, medium, low). Since this database was generated using inter-laboratory cross-validation, it contains multiple sets of MOS values. In this case, we also adopted the arithmetic average of the MOS values of the three universities which carry the subjective tests, namely UBI, UC, and UNIN.

- D_3 [5]: This database has 8 references and 232 test PCs and its contents are similar to D1. The distortions included in this database were generated by the MPEG-3dGC codecs, namely the video-based point cloud codec (V-PCC) and four variants of the geometric-based point cloud codec (G-PCC). The variants of G-PCC include the Region-Adaptive Hierarchical Transform with Trisoup (RAHT-Trisoup), RAHT with Octree (RAHT-Octree), Wavelet/Lifting-based with Trisoup (Lifting-Trisoup), and Wavelet/Lifting-based with octree (Lifting-octree). Subjective experiments were carried in two universities (UnB & EPFL), and we used the MOS values of the two universities averaged.
- D_4 [33]: This database has 6 references and 107 test PCs. This dataset contains human full-bodies and upper bodies, while the distortions were also created with the MPEG encoders, by the variants V-PCC and G-PCC. Subjective MOS values were obtained in 4 different universities - UBI, UC, UNIN and UTS.

We compared our metric with well-known state-of-the-art point-to-point (po2point), point-to-plane (po2plane), plane-to-plane (pl2plane), and projection-based (proj) PCQA metrics. We used the voxelization and kd-tree-based nearest neighbors search algorithms implemented in Open3D library[34]. Regression and statistical methods were taken from the Scikit-Learn library [35]. To test the performance of the proposed method, we compare the predicted scores with the subjective scores provided in the benchmark databases using the Spearman rank-order correlation coefficient (SROCC), shown in Equation 5; and Pearson linear correlation coefficient (PCC), shown in Equation 6. In this work, we used 9 distance metrics: Bray-Curtis, Canberra [36], Cityblock [37], Chebyshev, Cosine, Euclidean, Jensen-Shannon [28], Wasserstein [30], and Energy. These distance metrics were computed using the Scipy library [38].

$$PCC(m_i, p_i) = \frac{\sum_i (m_i - m_a)(p_i - p_a)}{\sqrt{\sum_i (m_i - m_a)^2}\sqrt{\sum_i (p_i - p_a)^2}}, \quad (5)$$

where m_i is the subjective MOS score, p_i is the predicted score, and m_a and p_a are their average.

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

where m_i is the subjective MOS score, p_i is the predicted score, r_i is the rank order of p_i and L is the number of test content PCs.

4. EXPERIMENTAL RESULTS & ANALYSIS

In order to evaluate our proposal, depicted in Fig. 1, we can vary the voxelization k parameter and the histogram distance metric. To investigate the most suitable parameter combination for each descriptor described in Section 2, we performed several test simulations, where we varied choices/parameters for the proposed PCQA method.

We tested the following voxelization parameters: 0.7, 1.0, 1.3, 1.6, 2.0, 3.0, 4.5, 6.0, 7.5 and 'novox'. The 'novox' parameter means no voxelization was performed, with the coordinates and attributes of the points being passed to the descriptors unchanged. Also, different distance metrics for the histogram distance calculation between reference and test content was evaluated. We present the results of the PCC and SROCC for PCDP and other metrics in figures 5 for D1, 6 for D2, 7 for D3 and 8 for D4.

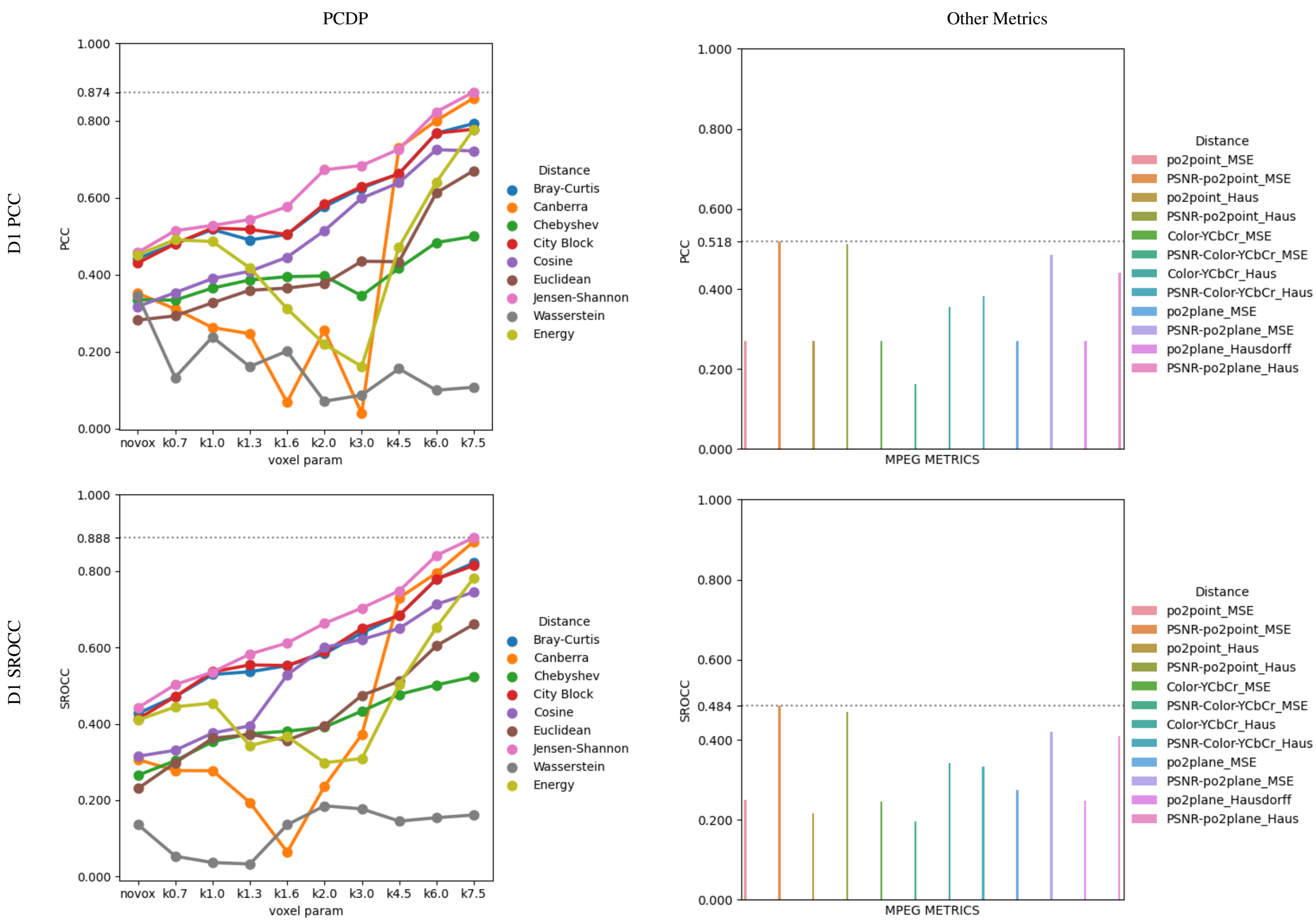

Fig. 5. Comparison of PCC and SROCC for PCDP, in terms of voxelization parameter k and different histogram distance methods, with other metrics, for D1 data-set.

5. CONCLUSIONS

In this paper, we proposed a novel color-based texture descriptor which considers local neighborhood perceptual color distances for quality assessment, named Perceptual Color Distance Patterns (PCDP). We also introduce a parametrized voxelization procedure to be applied prior to the feature extraction. To assess the quality of a test PC, we apply the descriptor to the reference and test PC contents and compare their statistics. We tested the proposed approach on four datasets and compared the results with state-of-the-art PCQA metrics. Results show that the proposed framework has a good and robust accuracy performance, outperforming or matching other methods while extending prior work PCQA methods based on local feature descriptors. This work also reinforces the good results obtained by using PC local neighborhoods for feature extraction, as suggested by our previous work [22, 24, 23], while it introduces a novel scale and rotation invariant PCQA method based on perceptual color differences, with promising results with strong performance when compared to other metrics.

Among the conclusions this paper brings to light are that the Jensen-Shannon distance presented the best results with our PCDP operator; that the voxel size influences the performance of the operator, so a way to optimize its selection is desirable, as different data-sets had different subjective procedures for quality assessment; when using the Jensen-Shannon distance, our proposal outscores all MPEG reference metrics in D1 and D2, in almost any k voxel setting, while in D3 and D4 data-sets we are third best with optimum k setting, after Point-to-Plane MSE and Point-to-Point MSE; the other tested metrics seem to work well when the content is degraded with the MPEG PC encoders, in which test conditions degrade geometry and color with analogous intensity.

Future works will focus on feature extractors for geometry features, for better quality prediction accuracy when used jointly to a color feature extractor. Among the limitations of our color-based PC quality metric framework is the limited capability of identifying geometry impairments. While some geometry impairments can modify local neighborhood color

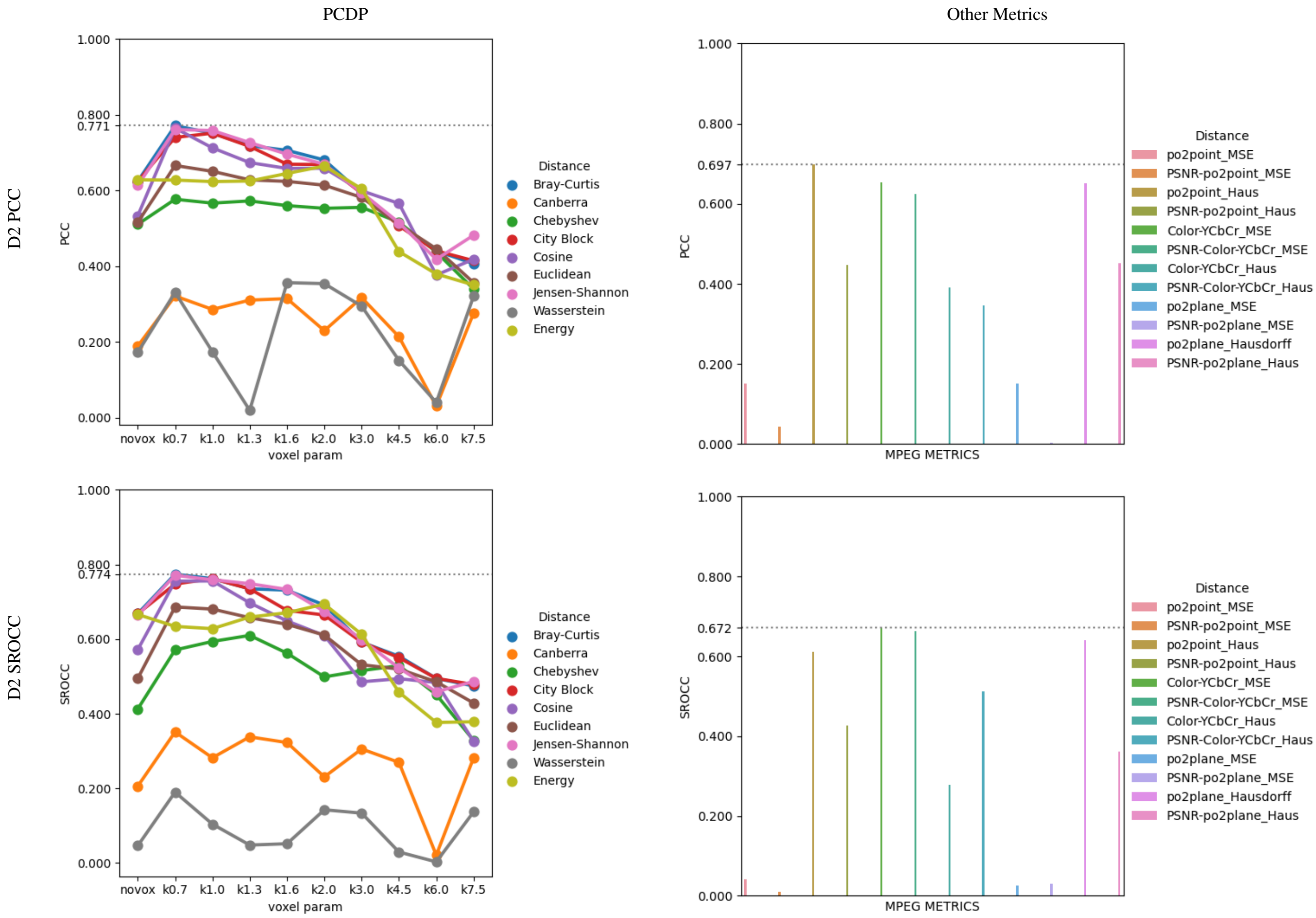

Fig. 6. Comparison of PCC and SROCC for PCDP, in terms of voxelization parameter k and different histogram distance methods, with other metrics, for D2 data-set.

statistics and indeed be captured by our proposal, some geometric impairments can not be captured by this indirect manner.

Acknowledgments

This work was supported by the Fundação de Apoio a Pesquisa do Distrito Federal (FAP-DF), by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), and by the University of Brasília (UnB).

6. REFERENCES

[1] Li Cui, Rufael Mekuria, Marius Preda, and Euee S Jang, "Point-cloud compression: Moving picture experts group's new standard in 2020," *IEEE Consumer Electronics Magazine*, vol. 8, no. 4, pp. 17–21, 2019.

[2] Euee S Jang, Marius Preda, Khaled Mammou, Alexis M Tourapis, Jungsun Kim, Danillo B Graziosi, Sungryeul Rhyu, and Madhukar Budagavi, "Video-based point-cloud-compression standard in mpeg: From evidence collection to committee draft [standards in a nutshell]," *IEEE Signal Processing Magazine*, vol. 36, no. 3, pp. 118–123, 2019.

[3] Gauthier Lafruit, Daniele Bonatto, Christian Tulvan, Marius Preda, and Lu Yu, "Understanding mpeg-i coding standardization in immersive vr/ar applications," *SMPTE Motion Imaging Journal*, vol. 128, no. 10, pp. 33–39, 2019.

[4] Weixin Lu, Guowei Wan, Yao Zhou, Xiangyu Fu, Pengfei Yuan, and Shiyu Song, "Deepvcp: An end-to-end deep neural network for point cloud registration," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 12–21.

[5] Evangelos Alexiou, Irene Viola, Tomás M Borges, Tiago A Fonseca, Ricardo L de Queiroz, and Touradj Ebrahimi, "A comprehensive study of the rate-distortion

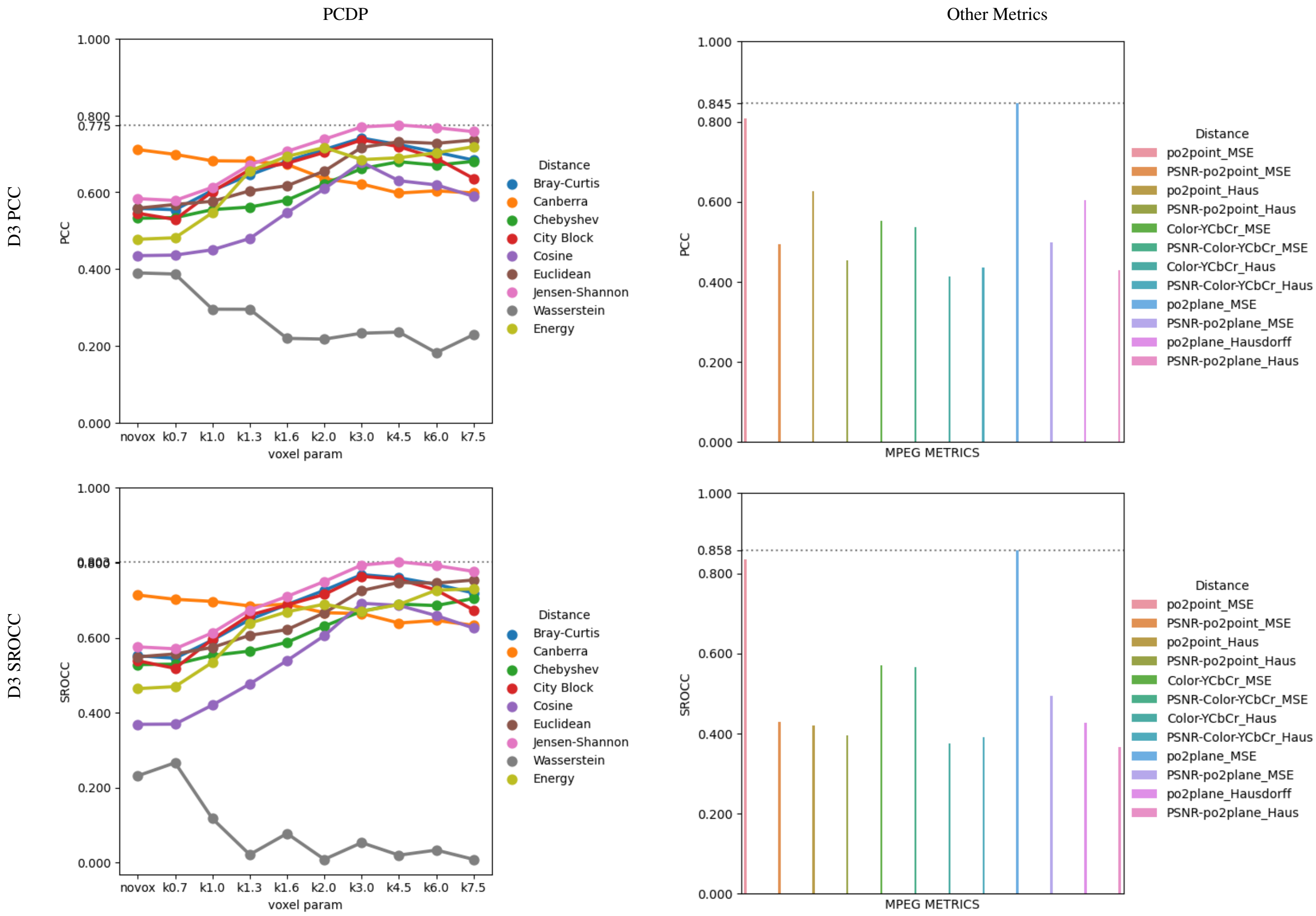

Fig. 7. Comparison of PCC and SROCC for PCDP, in terms of voxelization parameter k and different histogram distance methods, with other metrics, for D3 data-set.

performance in mpeg point cloud compression," *APSIPA Transactions on Signal and Information Processing*, vol. 8, 2019.

[6] Wei Hu, Xiang Gao, Gene Cheung, and Zongming Guo, "Feature graph learning for 3d point cloud denoising," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2841–2856, 2020.

[7] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 770–779.

[8] Tianhang Zheng, Changyou Chen, Junsong Yuan, Bo Li, and Kui Ren, "Pointcloud saliency maps," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1598–1606.

[9] Jiageng Mao, Xiaogang Wang, and Hongsheng Li, "Interpolated convolutional networks for 3d point cloud understanding," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 1578–1587.

[10] Fernando Pereira, "Point cloud quality assessment: Reviewing objective metrics and subjective protocols," in *ISO/IEC JTC1/SC29/WG1 M78036. JPEG*, pp. 1–8. JPEG, 2018.

[11] Gabriel Meynet, Julie Digne, and Guillaume Lavoué, "Pc-msdm: A quality metric for 3d point clouds," in *2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2019, pp. 1–3.

[12] Guillaume Lavoué, "A multiscale metric for 3d mesh visual quality assessment," in *Computer Graphics Forum*. Wiley Online Library, 2011, vol. 30, pp. 1427–1437.

[13] Eric M Torlig, Evangelos Alexiou, Tiago A Fonseca, Ricardo L de Queiroz, and Touradj Ebrahimi, "A novel methodology for quality assessment of voxelized point clouds," in *Applications of Digital Image Process-*

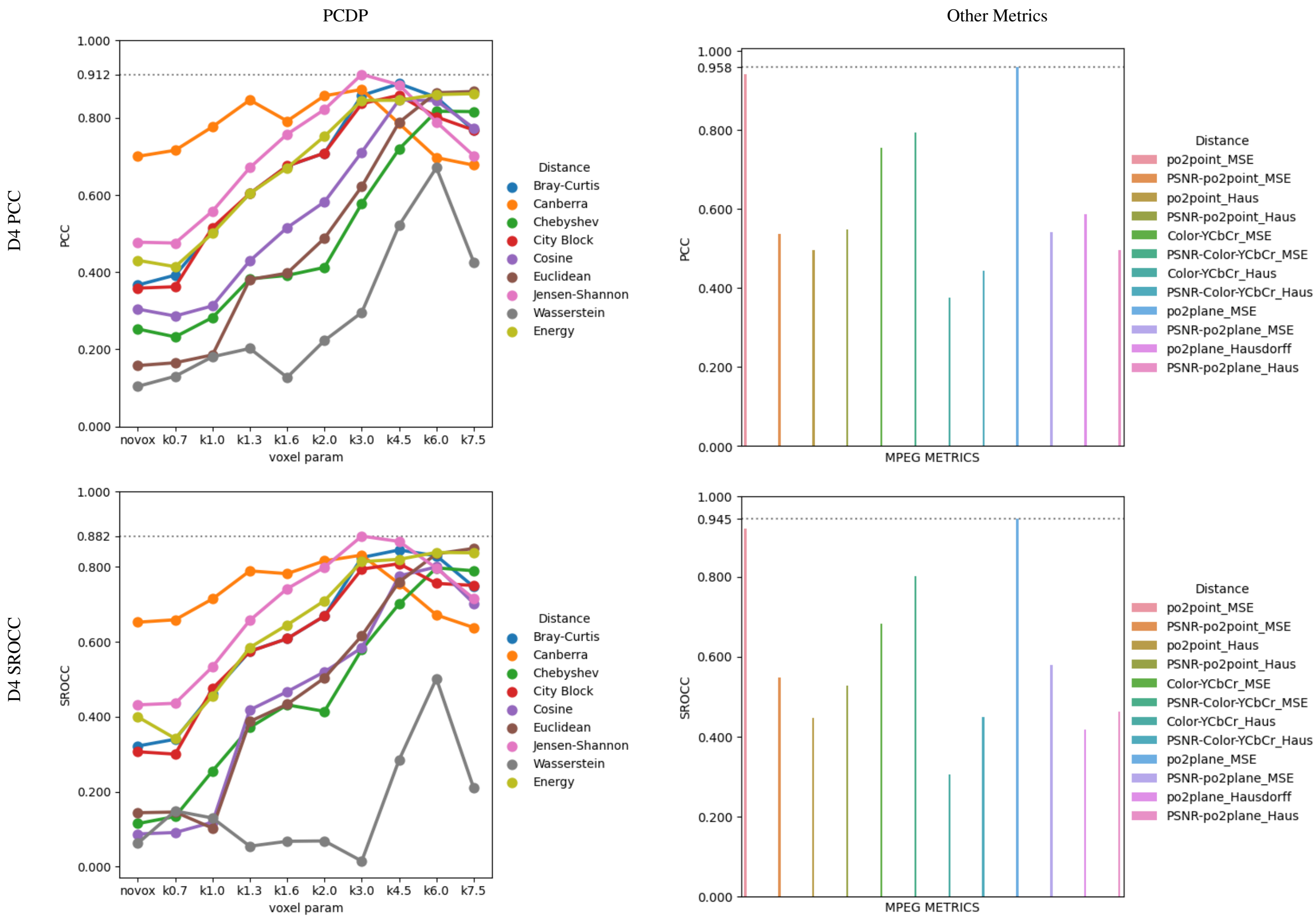

Fig. 8. Comparison of PCC and SROCC for PCDP, in terms of voxelization parameter k and different histogram distance methods, with other metrics, for D4 data-set.

ing XLI. International Society for Optics and Photonics, 2018, vol. 10752, p. 107520I.

[14] Evangelos Alexiou and Touradj Ebrahimi, "Point cloud quality assessment metric based on angular similarity," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2018, pp. 1–6.

[15] Javaheri Javaheri, Catarina Brites, Fernando Pereira, and Joao Ascenso, "A generalized hausdorff distance based quality metric for point cloud geometry," in *2020 12th International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2020, pp. 1–4.

[16] Irene Viola, Shishir Subramanyam, and Pablo Cesar, "A color-based objective quality metric for point cloud contents," in *2020 12th International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2020, pp. 1–6.

[17] Gabriel Meynet, Yana Nehme, Julie Digne, and Guillaume Lavoué, "PCQM: A full-reference quality metric for colored 3d point clouds," in *2020 12th International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2020, pp. 1–6.

[18] Evangelos Alexiou, Marco V Bernardo, Luis A da Silva Cruz, Lovorka Gotal Dmitrovic, Carlos Duarte, Emil Dumic, Touradj Ebrahimi, Dragan Matkovic, Manuela Pereira, Antonio Pinheiro, et al., "Point cloud subjective evaluation methodology based on 2d rendering," in *10th International Conference on Quality of Multimedia Experience (QoMEX)*, 2018, number CONF.

[19] Evangelos Alexiou and Touradj Ebrahimi, "Towards a point cloud structural similarity metric," in *2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, 2020, pp. 1–6.

[20] Saifullahi Aminu Bello, Shangshu Yu, Cheng Wang, Jibril Muhmmad Adam, and Jonathan Li, "deep learning on 3d point clouds," *Remote Sensing*, vol. 12, no. 11, pp. 1729, 2020.

[21] Yipeng Liu, Qi Yang, Yiling Xu, and Le Yang, "Point cloud quality assessment: Large-scale dataset construction and learning-based no-reference approach," *arXiv preprint arXiv:2012.11895*, 2020.

[22] Rafael Diniz, Pedro Garcia Freitas, and Mylene CQ Farias, "Multi-distance point cloud quality assessment," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 1–5.

[23] Rafael Diniz, Pedro Garcia Freitas, and Mylene CQ Farias, "Towards a point cloud quality assessment model using local binary patterns," in *2020 12th International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2020, pp. 1–6.

[24] Rafael Diniz, Pedro Garcia Freitas, and Mylène CQ Farias, "Local luminance patterns for point cloud quality assessment," in *2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2020, pp. 1–6.

[25] Daniel Cohen-Or and Arie Kaufman, "Fundamentals of surface voxelization," *Graphical models and image processing*, vol. 57, no. 6, pp. 453–461, 1995.

[26] Denis Haumont and Nadine Warzée, "Complete polygonal scene voxelization," *Journal of Graphics Tools*, vol. 7, no. 3, pp. 27–41, 2002.

[27] M Ronnier Luo, Guihua Cui, and Bryan Rigg, "The development of the cie 2000 colour-difference formula: Ciede2000," *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, vol. 26, no. 5, pp. 340–350, 2001.

[28] Dan-Dan Shi, Dan Chen, and Gui-Jun Pan, "Characterization of network complexity by communicability sequence entropy and associated jensen-shannon divergence," *Physical Review E*, vol. 101, no. 4, pp. 042305, 2020.

[29] Igor Vajda, "On the f-divergence and singularity of probability measures," *Periodica Mathematica Hungarica*, vol. 2, no. 1-4, pp. 223–234, 1972.

[30] Aaditya Ramdas, Nicolás García Trillos, and Marco Cuturi, "On wasserstein two-sample testing and related families of nonparametric tests," *Entropy*, vol. 19, no. 2, pp. 47, 2017.

[31] IT Union, "Objective perceptual assessment of video quality: Full reference television. itu-t telecommunication standardization bureau," 2004.

[32] Luis Cruz, Emil Dumic, Evangelos Alexiou, Joao Prazeres, Rafael Duarte, Manuela Pereira, Antonio Pinheiro, and Touradj Ebrahimi, "Point cloud quality evaluation: Towards a definition for test conditions," in *11th International Conference on Quality of Multimedia Experience (QoMEX 2019)*, 2019, pp. 1–6.

[33] Stuart Perry, Huy Phi Cong, Luís A da Silva Cruz, João Prazeres, Manuela Pereira, Antonio Pinheiro, Emil Dumic, Evangelos Alexiou, and Touradj Ebrahimi, "Quality evaluation of static point clouds encoded using mpeg codecs," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 3428–3432.

[34] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun, "Open3D: A modern library for 3D data processing," *arXiv:1801.09847*, 2018.

[35] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al., "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[36] Giuseppe Jurman, Samantha Riccadonna, Roberto Visintainer, and Cesare Furlanello, "Canberra distance on ranked lists," in *Proceedings of advances in ranking NIPS 09 workshop*. Citeseer, 2009, pp. 22–27.

[37] Bappaditya Mandal, Zhikai Wang, Liyuan Li, and Ashraf A Kassim, "Performance evaluation of local descriptors and distance measures on benchmarks and first-person-view videos for face identification," *Neurocomputing*, vol. 184, pp. 107–116, 2016.

[38] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake Vand erPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1. 0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.