Representation Learning Optimization for 3D Point Cloud Quality Assessment without Reference

Marouane Tliba, Aladine Chetouani, Giuseppe Valenzise, Frédéric Dufaux

REPRESENTATION LEARNING OPTIMIZATION FOR 3D POINT CLOUD QUALITY ASSESSMENT WITHOUT REFERENCE

Marouane Tliba[1], *Aladine Chetouani*[1], *Giuseppe Valenzise*[2] *and Fréderic Dufaux*[2]

[1]Laboratoire PRISME, Université d'Orléans, Orléans, France
[2]Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des signaux et systèmes

ABSTRACT

Recent information and communication systems have employed 3D Point Cloud (PC) as an advanced geometrical representation modality for immersive applications. Like most multimedia data, PCs are often compressed for transmission and viewing purposes, which can impact the perceived quality. Developing robust and efficient objective quality metrics for PCs is still an open problem. In this paper, we propose an end-to-end deep approach for evaluating the perceptual effects of point cloud compression solutions without reference. Our approach focuses on leveraging the intrinsic point cloud characteristics to quantify the coding impairments from few distant randomly selected patches using supervised and unsupervised training strategies. To evaluate the performance of our method, two well-known datasets have been used. The results demonstrate the effectiveness and reliability of the proposed method compared to to state-of-the-art methods.

Index Terms— 3D Point Cloud, Point Cloud Compression, No-Reference Perceptual Quality Metric, Deep Learning, Unsupervised Learning.

1. INTRODUCTION

Following the advent of recent immersive technologies and the abundant availability of low-cost 3D sensors, several 3D representation media have come to birth allowing live interaction with the geometrical content from any point of view [1][2]. Among those representations, 3D Point Cloud (PC) has been adopted as one of the most preferable formats. PCs are sets of unordered points determined by their Cartesian's coordinates and potentially associated attributes such as color, curvatures, reflectance and normal vectors. It has emerged in many recent applications such as Extended Reality, Real-time Immersive Communications, Robotics, 3D gaming, and Cultural Heritage. In practice, the depiction of a realistic 3D PC scene with a high resolution needs up to millions of points. As this huge size of data requires further computational sensitive operations such as storing and transmission, the application of emerging compression schemes becomes inevitable, impacting the perceived quality. Achieving a balance between user satisfaction and data cost is thus important. Specifically, to evaluate the rate-distortion or performance of existing point cloud coding schemes, perceptual quality metrics need to be adopted in order to assess the perceptual fidelity of the decoded point cloud regarding a certain compression rate.

Generally, Point Cloud Quality Assessment (PCQA) could be obtained through subjective and objective experiments. The former requires a human intervention which is expensive and time-consuming, while the latter relies on computational methods that predict the perceptual quality. Depending on the availability of the reference image, objective quality metrics fall into three categories: Full-Reference [3], Reduced-Reference [4] and No-Reference [5, 6, 7]. Existing quality metrics for 3D PCs can be classified into three main groups: Point-based, Feature-based and Projection-based metrics. Point-based metrics such as Point-to-Point (Po2Po)[8], Point-to-Plane (Po2Pl)[9], Plane-to-Plane (Pl2Pl)[10] and Point-to-Mesh (Po2Mesh)[11], predict the quality through point-wise geometric and/or features distance between the reference PC and its distorted version. It is worth noting that MPEG is adopting Po2Po MSE and Po2Pl MSE with the associated PSNRs as the standard PC geometry quality metrics. Feature-based PC Quality Metrics, extract the geometry with the associated attributes form point-wise level in a global or local way. Among those metrics, we can cite, PC-MSDM [11] that extends the 2D SSIM metric [12] to PC by considering local curvature statistics, the Geotex [13] metric that exploits the Local Binary Pattern (LBP) [14] descriptors, and PCQM [15] that combines the geometry and color features. In projection-based PC Quality Metrics, the 3D points are projected into 2D regular grids and 2D quality metrics are applied on these views.

Lake of large 3D PC quality data-set, hinder the development of efficient deep based quality metrics, thus, finding a way to push representation learning from limited amount data is mandatory. Moreover, all the above-cited PCQA metrics need long time of pre-processing, which is computationally expensive and most of them require the reference PC. Therefore, deploying these metrics in edge devices for real-time quality assessment or adopting as loss function to optimize learnable compression models remains difficult.

To overcome the aforementioned shortage of PC quality metrics, and motivated by the nature of visual impairment resulting from emerging point cloud compression solutions [16]

that introduce somehow uniform effects distributed over all the geometrical content.In this paper we look to learn an efficient representation from local intrinsic characteristics of PC patches. More precisely, we extract PC patches by first selecting M centroid points using the farthest point sampling algorithm [17]. Then, we apply the K-nearest neighbor clustering method to form a patch around each centroid. Finally, we learn a lightweight permutation invariant feature descriptor function, followed by a shallow regressor that estimates the final quality score. To further optimize the model, beside training our shallow network using the supervised loss, we also use a self-supervised ranking loss. The goal is forcing the model to learn richer descriptive representations capturing local intrinsic characteristics of point clouds, and discard the rest of cues, that result in learning an inner representation related to the point cloud global shape. To the best of our knowledge, we are the first who consider learning representation directly from 3D PC without applying projection or transformation such us voxelization.

The main contributions of this paper are summarized in what follow:

- We propose a novel efficient end-to-end and shallow deep model for assessing the perceptual effects of PC compression methods relying on the local intrinsic characteristics of sub-sets of points.
- We extend the self-supervised strategy of learning from rank as pseudo label to optimize the learning more on intrinsic point sub-set features, toward a potential descriptive representation related to our downstream task.
- Two well-known datasets are used to demonstrate the effectiveness of the proposed approach comparing to state-of-the-art methods.

The rest of the paper is organized as follows: In section 2, we describe the proposed method and the learning strategy applied. In Section 3, we present the adopted experimental protocol and discuss the results. And finally, we give some conclusions in Section 4.

2. PROPOSED METHOD

The main goal of this work is to provide an efficient blind metric for estimating the quality of 3D compressed PCs without adding any projection step or applying changes on the specificity of the points. Motivated by the nature of compression schemes in producing uniform distributed effect over local features, we assume that the distribution of compressed PCs could be represented as a set of hidden features or descriptors related to the compression effects such as point consistency, sparsity, and density; plus a set of descriptors related to other characteristics such as position, global shape, and structure. In order to learn efficiently the representation of these hidden or intrinsic features and inspired by PointNet [17], we use here a similar shallow permutation invariant encoder through series of shared 1D convolutions over all points. We then employ a shallow regressor to predict the quality. Inspired from previous works that learn representation from rank [18][19][20]. Our network is optimized to learn the inner representation from the labeled data, besides, the known implicit rank between data samples as pseudo labels, simultaneously. Fig. 1 demonstrates the training process and the architecture of the proposed approach.

2.1. Patch extraction and Pre-Processing

The PC is first divided into small patches by considering just the (x,y,z) coordinates and the RGB color information. For that, the farthest point sampling [17] and K nearest neighbor algorithms are used as follows:

- From a given PC X, we first apply the farthest point sampling algorithm to select M centroids (i.e. C = $\{ P_1, P_2, P_3, .., P_m \}$)
- In order to form a patch around each centroid, we then employ the K nearest neighboring clustering method. The sub-set of points of the i_{th} centroid, denoted here $\{ P_i^1, P_i^2, P_i^3, .., P_i^K \}$, represents the i_{th} patch. K was fixed here to 512.
- Finally, we compute the position and color differences between each sub-set of points and its corresponding centroid. We obtain the final sub-set of points, denoted here S_i, that is fed as input to the model $\{ P_i^1 - P_i, P_i^2 - P_i, P_i^3 - P_i, .., P_i^K - P_i \}$.

It is worth noting that the use of patches allows to augment the training data as the existing datasets for 3D PC quality are quite short. It also helps to generalize the learning from only local features, which leads to reduce the reliance of model to other global information and decrease the computational time.

2.2. Network Architecture

The design of our network is inspired from the efficient PointNet backbone [17]. Since we are working on patches, we focus to learn useful representation only from local intrinsic features rather than the global shape. To sum up, the architecture of our model is composed of a symmetric invariant features extractor (i.e. series of 1D convolution followed by a max polling operation over point representation) and a shallow deep regressor followed by a sigmoid activation in order to produce a probability that represents the estimated quality score.

2.3. Self-supervised and supervised learning

As mentioned previously, we adopt two strategies to optimize our model: supervised learning from the subjective scores and self-supervised learning from the rank. The supervised learning stage aims to learn a mapping function to the Mean Opinion Score (MOS), while the goal of the self-supervised stage is to learn a better representation that maximizes the learning from intrinsic features. We look here to force the model to

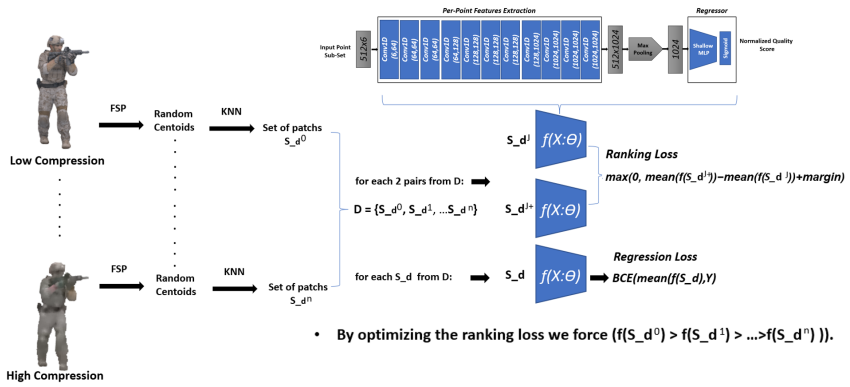

Fig. 1. : Training Process and model architecture.

maximize the learning of local intrinsic features from point sub-set, in label unsupervised manner, considering:

$$S_d = Q_d + N_d \tag{1}$$

where S_d represents the distribution of a point sub-set. Q_d is the distribution of a well representative intrinsic features related to the perceptual quality estimation, while N_d is the distribution of other features that are not related to our downstream task.

2.3.1. *Supervised Learning*

The aim from the supervised step is to find a function $f(S_d : \theta)$ with parameters θ that captures the relationship between S_d and quality scores Y; Where $(X, Y) = \{(S_{d0}, Y), (S_{d1}, Y), ..(S_{dm}, Y)\}$. Since the output of our model is a probability value that represents the perceptual quality, we consider the binary cross entropy (BCE) as loss function to minimize the regression empirical risk over the S_d set.

$$Rg\_loss(P, Y) = -Y \log(P) - (1 - Y) \log(1 - P) \tag{2}$$

where $P = f(S_d : \theta)$, and Y is the MOS of the point cloud sample.

2.3.2. *Self-Supervised Learning*

The goal from the self-supervised step is to have a proxy supervised task with rich pseudo labels [21]. It allows to learn the separation from different degradations, and at the same time maximize the learning over the Q_d distribution (i.e. intrinsic features).

As we know that higher compression level produces higher perceptual damage. So, for a given encoding module (i.e., MPEG G-PCC Octree/TriSoup [22] [23] [24], MPEG V-PCC), applying the compression levels l_j, and l_{j+} on point cloud X results on point clouds X_j and X_{j+}, respectively. Hence we could easily know the generated rank based on the introduced damage.

Considering the visual quality degradation of those encoding modules is uniform over the X geometry, we want our model to focus on intrinsic features and, ignore the learning from other features such as patch position (i.e., $f(S_d : \theta) = f(Q_d : \theta)$. In other word, whatever the selected patches from X_j and X_{j+}, the model should give outputs ordered according to the inverse order induced by the level of compression:

$$l_j < l_{j+} \implies mean(f({S_d}^j)) > mean(f({S_d}^{j+})) \tag{3}$$

To this end, we use a Siamese Network [25] to minimize the ranking loss between the activation of input pairs. Each branch of this Siamese Network is based on our model parameter.

$$Rank\_Loss = max(0, \Delta f + margin) \tag{4}$$

The gradient of the Rank_Loss in Eq (4) is:

$$\nabla_\theta Rank\_Loss = \begin{cases} 0, & \text{if } \Delta f + margin \leq 0 \\ \nabla_\theta(\Delta f), & \text{otherwise} \end{cases}$$

where $\Delta f = mean(f({S_d}^{j+})) - mean(f({S_d}^j))$

In much details, by optimizing the Siamese network through minimizing the $Rank\_Loss$, the gradient tends to zero when the network activation has the inverse order as induced by the compression. Whereas, the gradient of the higher activation decreases while the gradient of the lower activation increases when the activations have the same order as induced by the compression.

2.4. Training Protocol

At each training step, we update the weights by minimizing both the ranking self-supervised loss and the supervised loss as multi-task learning:

$$MultiTask\_loss = Rank\_Loss + 0.01 * Rg\_loss \tag{5}$$

It is worth noting that at each training step, the selected centroids for extracting patches change (i.e., $(C^j) \neq (C^{j+})$). In other words, we implicitly urge the model to discard the position and global shape information, and maximize the learning on the quality related information which represent the mutual information across (S_d) patches. In addition, this strict protocol (i.e., the training data changes at each epoch) makes the

model more robust and thus increase its generalization ability. The value of the margin changes based on the distance between compression levels, varying from 0.1 for the successive degradation level to 0.6 for the farthest degradation level. We set the initial model weights of randomly [26], and train the whole parameters in end-to-end manner for 500 epochs using Adam optimizer [27].

3. EXPERIMENTAL RESULTS

We evaluate our model through two well-known datasets that employ different compression schemes with multiple encoding levels: **ICIP20** [16] and **PointXR** [28]. **ICIP20** is composed of 6 reference PCs from which 90 degraded versions were derived through three types of compression: V-PCC, G-PCC with triangle soup coding and G-PCC with octree coding. Each reference PC was compressed using five different levels. **PointXR** is composed of 5 PCs from which 45 degraded versions were derived through G-PCC with octree coding for geometry compression and, Lifting and RAHT for color compression.

Model	PLCC ↑	SROCC ↑
po2pointMSE	0.945	0.950
po2planeMSE	0.945	0.959
PSNRpo2pointMSE	0.880	0.934
PSNRpo2planeMSE	0.916	0.953
Our-S	0.745	0.621
Our-(S+SSL) P2	**0.908**	**0.955**

Table 1. Results obtained on ICIP20 dataset

Model	PLCC ↑	SROCC ↑
po2pointMSE	0.887	0.978
po2planeMSE	0.855	0.942
PSNRpo2pointMSE	0.983	0.978
PSNRpo2planeMSE	0.972	0.950
Our	**0.964**	**0.970**

Table 2. Results obtained on PointXR dataset

To fairly study the performance and for the sake of providing a solid validation baseline protocol to future works, we randomly select 64 centroids for validation as we do during the training. We adopt a 6 fold cross validation protocol on ICIP20 dataset through splitting the dataset into training validation 6 times, where 6 refers to the number of reference point cloud samples. More precisely, at each iteration 5 reference point cloud samples and their compressed versions are used for training, and 1 reference point cloud sample and its compressed versions are used for testing. Lastly, to emphasize the effectiveness, and validate the generalization ability of our model on unseen compression, we apply a cross-dataset validation protocol by training our model on ICIP20 and testing it on PointXR as commonly used to rigorously benchmark deep learning methods. Pearson Correlation Coefficient (PCC) and Spearman Rank-Order Coefficient Correlation (SROCC) are computed to evaluate the quality prediction ability of our method. These correlations are computed regarding each fold independently during the validation, and finally the mean is reported.

Table 1 shows the performance of our method on ICIP20 dataset using only supervised learning, denoted here as Our-(S) and supervised with self-supervised as multi-task learning, denoted here as Our-(S+SSL). The results are also compared to a set of state-of-the-art methods. As can be seen, using the self-supervised and the supervised losses allows to considerably improve the performance. Moreover, our method (S+SSL) is quite competitive with state-of-the-art models, especially in terms of ranking (i.e. SROCC) the compression distortion effects. It is also worth noting that unlike our method all of listed methods require the reference point cloud and take long time during the testing.

Table 2 shows the results obtained for the cross-dataset evaluation. As can be seen, high correlations are reached by our method, outperforming some of the compared ones. This results shows the generalization ability of our method to predict the quality of unseen PCs. It is remarkable that our method is the most consistent one comparing to the performance achieved on ICIP20.

4. CONCLUSION

In this paper, we proposed an efficient end-to-end shallow deep learning-based approach for interpolating quality scores induced by emerging compression effects. Unlike previous methods, the proposed approach works directly on point cloud patches without requiring prior computational expensive processing step. We urge our model to learn potential representation from point sub-set by focusing on the local intrinsic features related to our downstream task. To optimize the model, we also used self-supervised as rank learning strategy in order to learn a direct mapping to the accurate quality scores, while maximizing the distance between the representation of different degradations. The obtained results using a strict k-fold cross validation and cross dataset validation protocols demonstrated the effectiveness of our lightweight model. It also showed that our model is competitive comparing to state-of-the-art full reference methods. We look in our future works to extend our model by robust learnable aggregation mechanism in order to consider better the local and global effects on point distribution.

5. REFERENCES

[1] A. Chetouani et al, "Classification of engraved pottery sherds mixing deep-learning features by compact bilinear pooling," *Pattern Recognition Letters*, vol. 131, pp. 1–7, 2020.

[2] M. Tliba et al, "2d-based saliency prediction framework for omnidirectional–360° video," in *11th International Conference of Pattern Recognition Systems (ICPRS 2021)*, 2021, vol. 2021, pp. 31–37.

[3] Aladine Chetouani, Maurice Quach, Giuseppe Valenzise, and Frédéric Dufaux, "Convolutional Neural Network for 3D Point Cloud Quality Assessment with Reference," in *IEEE MMSP*, Tampere, Finland, Oct. 2021.

[4] A. Chetouani et al, "A reduced reference image quality metric based on feature fusion and neural networks," in *2011 19th European Signal Processing Conference*, 2011, pp. 589–593.

[5] Aladine Chetouani and Leida Li, "On the use of a scanpath predictor and convolutional neural network for blind image quality assessment," *Signal Processing: Image Communication*, vol. 89, pp. 115963, 2020.

[6] I. Abouelaziz et al, "3D visual saliency and convolutional neural network for blind mesh quality assessment," *Neural Computing and Applications*, 2019.

[7] I. Abouelaziz, A. Chetouani, M. El Hassouni, L.J. Latecki, and H. Cherifi, "No-reference mesh visual quality assessment via ensemble of convolutional neural networks and compact multi-linear pooling," *Pattern Recognition*, vol. 100, pp. 107174, 2020.

[8] C. Tulvan R. Mekuria, Z. Li and P. Chou, "Evaluation criteria for pcc (point cloud compression)," in *in ISO/IEC MPEG Doc*. N16332, 2016.

[9] C. Feng R. Cohen D. Tian, H. Ochimizu and A. Vetro, "Geometric distortion metrics for point cloud compression," in *IEEE ICIP*, 2017.

[10] E. Alexiou and T. Ebrahimi, "Point cloud quality assessment metric based on angular similarity," in *IEEE ICME-W*, 2018.

[11] C. Rochinni P. Cignoni and R. Scopigno, "Metro: measuring errors on simplified surfaces,," in *Computer Graphics Forum*, 1998, vol. 17, pp. 167–174.

[12] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[13] Rafael Diniz, Pedro Garcia Freitas, and Mylène C. Q. Farias, "Towards a point cloud quality assessment model using local binary patterns," in *QoMEX*, 2020, pp. 1–6.

[14] Matti Pietikäinen and Guoying Zhao, "Two decades of local binary patterns: A survey," *CoRR*, vol. abs/1612.06795, 2016.

[15] J. Digne G. Meynet, Y. Nehmé and G. Lavoué, "Pcqm: A full-reference quality metric for colored 3d point clouds," 2020.

[16] L.Cruz J. Prazeres M. Pereira A. Pinheiro E. Dumic E. Alexiou T. Ebrahimi S. Perry, H. Cong, "Quality evaluation of static point clouds encoded using mpeg codecs," pp. 3428–3432, 2020.

[17] C. Qi, L. Yi, Hao Su, and Leonidas J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," 2017.

[18] Aliaksei Severyn and Alessandro Moschitti, "Learning to rank short text pairs with convolutional deep neural networks," pp. 373–382, 2015.

[19] X. Liu, J. Van De Weijer, and A. D. Bagdanov, "Rankiqa: Learning from rankings for no-reference image quality assessment," pp. 1040–1049, 2017.

[20] Thorsten Joachims, "Optimizing search engines using clickthrough data," *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002.

[21] M. Tliba et al, "Self supervised scanpath prediction framework for painting images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2022, pp. 1539–1548.

[22] S. Schwarz 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, 2019.

[23] E. Pavez et al, "Dynamic polygon clouds: representation and compression for vr/ar," vol. 7, 2018.

[24] Ricardo L. de Queiroz and Philip A. Chou, "Compression of 3d point clouds using a region-adaptive hierarchical transform," *IEEE Transactions on Image Processing*, vol. 25, pp. 3947–3956, 2016.

[25] R. Hadsell S. Chopra and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005, vol. 1, pp. 539–546 vol. 1.

[26] Xavier Glorot and Yoshua Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *AISTATS*, 2010.

[27] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

[28] N. Yang E. Alexiou and T Ebrahimi, "Pointxr: A toolbox for visualization and subjective evaluation of point clouds in virtual reality," 2020.