



HAL
open science

Convolutional Neural Network for 3D Point Cloud Quality Assessment with Reference

Aladine Chetouani, Maurice Quach, Giuseppe Valenzise, Frédéric Dufaux

► **To cite this version:**

Aladine Chetouani, Maurice Quach, Giuseppe Valenzise, Frédéric Dufaux. Convolutional Neural Network for 3D Point Cloud Quality Assessment with Reference. IEEE International Workshop on Multimedia Signal Processing (MMSP'2021), Oct 2021, Tampere, Finland. pp.1-6, 10.1109/MMSP53017.2021.9733565 . hal-03298952

HAL Id: hal-03298952

<https://hal.science/hal-03298952>

Submitted on 24 Aug 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Convolutional Neural Network for 3D Point Cloud Quality Assessment with Reference

Aladine Chetouani
Laboratoire PRISME
Université d'Orléans
Orléans, France
aladine.chetouani@univ-orleans.fr

Maurice Quach, Giuseppe Valenzise, Frédéric Dufaux
L2S, Centrale Supélec
Université Paris-Saclay
Gif-sur-Yvette, France
{firstname.lastname}@l2s.centralesupelec.fr

Abstract—In recent years, the production of 3D content in the form of point clouds (PC) has increased considerably, especially in virtual reality applications. This enthusiasm is linked in particular to the development of acquisition technologies. In order to ensure a good quality of user experience, it is necessary to offer a high quality of visualization whatever the transmission medium used or the treatments applied. Thus, several metrics have been proposed which are essentially point-based metrics. In this article, we propose a deep learning-based method that efficiently predicts the quality of distorted PCs thanks to a set of features extracted from selected patches of the reference PC and its degraded version as well as the use of Convolutional Neural Networks (CNNs). The patches are selected randomly and the difference between corresponding patches is characterized by three attributes: geometry, curvature and color. The proposed method was evaluated and compared to state-of-the-art metrics using two datasets, including a large dataset more suited to deep learning models. We also compared different symmetrization functions and machine learning pooling as well as the ability of our method to predict the quality of unknown PCs through a cross-dataset evaluation. The results obtained show the relevance of the proposed framework with interesting perspectives.

Index Terms—3D point cloud, Quality assessment, Convolutional neural network

I. INTRODUCTION

Recent advances in capture technologies have increased the production of 3D content in the form of Point Clouds (PCs). As most multimedia contents, PCs may undergo different types of distortion introduced by several basic processing (acquisition, compression, transmission, visualization, etc.), usually applied to transmit or visualize such data [1], [2]. To estimate the perceptual impact of these distortions on the perceived quality, subjective and objective evaluations are usually conducted. Subjective evaluation gives scores that reflect the perception of human observers through psycho-visual tests, while objective evaluation aims to automatically predict the subjective scores. As for 2D images and videos, objective methods can be classified according to the availability of the reference PC: Full Reference (FR) approaches that need the reference PC, Reduced Reference (RR) approaches that exploit only partial information from the reference PC and No

Reference (NR) approaches that predict the quality from only the distorted version of the reference PC.

Different point cloud objective metrics have been proposed in the literature. Point-to-point metrics were among the first to be considered, and compute geometry distance between corresponding points in the original and distorted PC. On the other hand, the point-to-plane metric is an extension of the previous metric and consists in projecting the point-to-point error vector along the local normal [3]. Starting from these approaches, several point-based metrics have been then developed. In [4], the authors proposed to estimate the geometrical error between two PCs (i.e. reference PC and its distorted version) by measuring the angular similarity between tangent planes. In [5], the authors proposed a metric called PC-MSDM by extending the well-known SSIM metric [6], widely used for 2D images, to PC, by considering features including local mean curvature as they previously done for 3D meshes [7]. The authors proposed later a metric called PCQM that integrates color information [8]. In [9], the authors proposed a new approach that focuses more on the distribution of the data. They introduced a new type of correspondence from point to distribution characterized using the well-known Mahalanobis distance. In [10], the authors proposed a color-focused metric that integrates geometry information. In [11], the authors adapted also the SSIM metric for point clouds using a number of features, while in [12] the authors improved the point cloud PSNR metric. Interesting methods were also proposed for 3D meshes [13]–[15].

In this paper, we present a *deep learning-based* method that efficiently predicts the quality of PCs. Our method extended our blind metric [16] and consists first of selecting a set of points from the reference PC and determining for each of them its nearest neighbor point in the distorted PC. We then define a patch of size 32×32 around each of the two points, each consisting of a point (i.e. selected point in the reference PC or its nearest neighbor point in the distorted) and its 1023 neighboring points. The structural information of each pair of patches is afterward compared by computing element-wise distances on three main attributes: geometry, curvature and color. The resulted patches (i.e. one per attribute) are stacked into a patch of size $32 \times 32 \times 3$ and fed as input to a Convolutional Neural Network (CNN) model to predict its

quality. After applying the process in both directions (i.e. from the reference PC to its distorted PC and from the distorted PC to its reference PC) for each pair of corresponding patches, the global quality index is finally given by averaging the predicted patch quality scores and employing either a symmetrization function or a pooling strategy. The proposed method was evaluated using two datasets, including a large dataset more suited to deep models. The results obtained showed the relevance of using such deep learning-based approach for predicting the quality of PCs.

The remainder of this paper is structured as follows: the proposed method is described in Section II. Experiment results are discussed in Section III, followed by the conclusion in Section IV.

II. PROPOSED METHOD

The method proposed in this paper is based on the prediction of Patch Quality Indexes (PQIs) through CNN models and their symmetrization or pooling. Fig. 1 summarizes the general framework of our method. From the two PCs, we first compute PQIs in both directions (i.e. from the reference PC to its distorted version: $PQI(X, Y)$ as well as from the distorted PC to its reference version: $PQI(Y, X)$) using a CNN model per direction. The Global Quality Index (GQI) is finally given by averaging the PQIs obtained and then either symmetrizing or pooling the resulted scores.

A. Patch Quality Indexes

As illustrated by Fig. 2, PQIs are computed through several steps: patch extraction, patch-based distances computation and patch quality prediction using a CNN model. Each of these steps is described in this section. It is worth noting that here we limit our description to one direction (i.e. from the reference PC to its distorted version), since it remains similar in the second direction.

1) **Patch extraction:** In order to extract patches from the two PCs, we first select a set of points N from the reference PC X . More precisely, 1000 points are randomly selected over all points of X and are here considered as key points from which the patch quality indexes are predicted. For each selected point N_i , we determine its corresponding point in the distorted PC Y by finding its nearest neighbor and define a region around each of the two points, delimited by the area covered by their 1023 neighboring points. The two regions are then reshaped into patches of size 32×32 based on the distance of each neighbor to the point, forming thus a pair of patches (see Fig. 3).

2) **Patch-based distances:** Element-wise patch-based distances are then computed to measure geometry, curvature and color distortions between patches of corresponding points. Let us first define A_k as the k^{th} considered attribute with $\{k = 1: \text{Geometry}; k = 2: \text{Curvature}; k = 3: \text{Color}\}$.

For each attribute A_k , the element-wise distance $D_{A_k}^n(i, j)$ between the n^{th} pair of patches of the reference X (P_X^n) and the distorted PCs Y (P_Y^n) is computed as follows:

$$D_{A_k}^n(i, j) = \sqrt{(P_{X_k}^n(i, j) - P_{Y_k}^n(i, j))^2}, \quad (1)$$

where $P_{X_k}^n(i, j)$ and $P_{Y_k}^n(i, j)$ represent the value of the k^{th} attribute at position (i, j) of the pair of patches n in the reference and distorted PCs, respectively.

The resulting patches are then stacked into a patch of size $32 \times 32 \times 3$ where 32×32 represents the initial size of the extracted pair of patches and 3 is the number of considered attributes (i.e. 1: geometry distances, 2: curvature distances and 3: color distances). It is worth noting that the size of the patches was fixed according to several studies dedicated for 2D images where the impact of the patch size was analyzed [17], [18]. The authors concluded that a size of 32×32 constituted a good trade-off between performance and computation time. We opted here for the same size. However, more in-depth analysis has to be carried-out in order to better analyze its impact on the performance.

3) **Architecture of the CNN models:** Finally, the PQIs are predicted using a CNN model which has as input the normalized stacked patches. A plethora of models have been proposed in the literature, starting from one of the first models such as AlexNet [19] to the more complex like ResNet [20] that introduced a residual module or inception [21] that employed parallel layers. In this study, we adopted a pre-trained VGG model [22] that was widely used for classification tasks and was successfully employed in several studies related to quality assessment as well [23].

This model was developed by the Oxford Visual Geometry Group in 2014. It uses small filters of size 3×3 in contrast to AlexNet [19] where 11×11 and 5×5 filters are used in the first two layers. It is also characterized by applying a max pooling layer after a succession of convolution layers. Various versions with different depths have been proposed. Here, we used VGG16 which is composed of 13 convolutional layers, followed by 3 fully connected layers. It initially takes an image of size $224 \times 224 \times 3$ as input and highlights one class among 1000 classes (i.e. output of size 1000).

To adapt the model to our context, several modifications were realized. More precisely, we replaced its input layer by another of size $32 \times 32 \times 3$ and substituted its fully connected layers with 3 other layers of size 64, 64 and 1, respectively. The first fully connected layer is followed by a ReLU layer, while the second one is followed by a ReLU layer and a dropout layer with a probability fixed to 0.5. The third and last layer is a regression layer that aims to predict the PQIs of the normalized stacked patches. The model thus modified was finally fine-tuned to adapt its weights to our context. It is worth noting that two similar CNN models with the architecture described above are used, each dedicated to a specific direction: CNN_{XY} (from the reference PC to the distorted one) and CNN_{YX} (from the distorted PC to the reference one).

To train our models, we used the stochastic gradient descent optimization algorithm and the Mean Squared Error (MSE) as loss function. The learning rate and the momentum were set to 0.001 and 0.9, respectively. The batch size was fixed to 64 and the training data was shuffled every epoch. The number of epochs was set to 100 with a validation frequency fixed

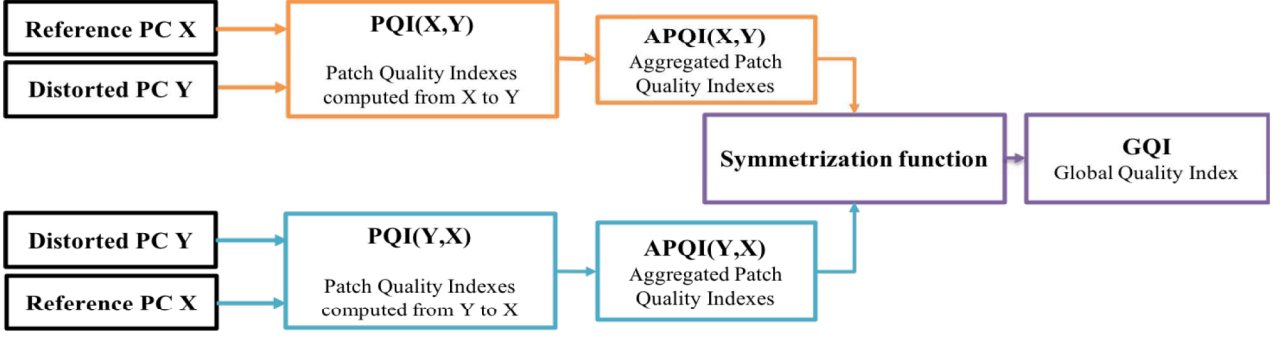


Fig. 1. General framework of the proposed method.

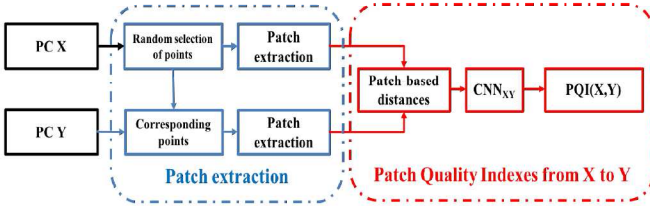


Fig. 2. Patch Quality Index (PQI) prediction.

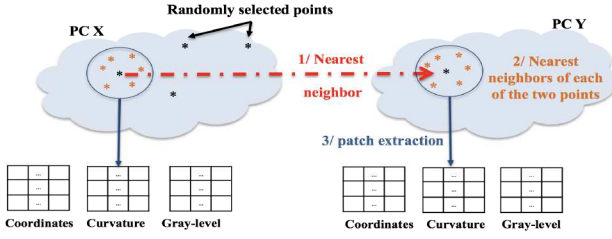


Fig. 3. Patch extraction process.

to 5000 iterations. At each epoch, the model was saved and the one that provides the best results was retained. It is worth noting that the target of each stacked patch was the subjective score of the whole distorted PC as commonly used to estimate the quality of several multimedia content [17], [24].

B. Global Quality Index

The Global Quality Index (GQI) that reflects the quality of the whole distorted PC is finally given by aggregating the PQIs achieved for each direction and applying either a symmetrization function or a pooling method.

1) **Aggregated Patch Quality Indexes:** As mentioned above, each CNN model predicts the quality score of each stacked patch. In order to derive a single quality index for each direction, we aggregate the obtained PQIs by computing the average scores as follows:

$$APQI(X, Y) = \frac{1}{N} \sum_{i=1}^N PQI_i(X, Y), \quad (2)$$

and

$$APQI(Y, X) = \frac{1}{N} \sum_{i=1}^N PQI_i(Y, X), \quad (3)$$

where $APQI(X, Y)$ and $APQI(Y, X)$ are the Aggregated Patch Quality Indexes (APQIs) that represent the predicted quality score in each direction. N is the number of stacked patches and it was fixed to 1000.

It is worth noting that this strategy is commonly applied in image quality assessment domain for both handcrafted and deep learning-based methods. Indeed, handcrafted metrics like SSIM [6] or VDP [25] derive a single quality score from the predicted visibility map by aggregating the local scores. Whereas, the initial deep learning-based method [17] and most of those developed later [18], [26], [27] predict the quality of patches and derive a single quality score by averaging the predicted scores as well. For 3D contents, the same strategy was applied for 3D PC [5] and 3D meshes [7], [28] as well.

2) **Symmetrization or Pooling:** As most of the existing full reference PC quality metrics [3], [9], [12], the final quality index is computed in two directions (i.e. from the reference PC to the distorted one and from the distorted PC to the reference one) where each reflects a specific aspect. The two indexes are then usually symmetrized by applying a function f as follows:

$$GQI(X, Y) = f(APQI(X, Y), APQI(Y, X)), \quad (4)$$

where $GQI(X, Y)$ is the Global Quality Index computed between the reference PC X and its distorted version Y in both directions.

In this study, we considered three classical symmetrization functions: *min*, *mean* and *weighted mean (Wmean)*. It is worth noting that the *max* was not applied since the considered datasets provide the MOS (Mean Opinion Score) as subjective score. The results are compared to those obtained by pooling the indexes through a Support Vector Machine (SVR) and a Multi Layer Perceptron (MLP). More precisely, the SVR model had a Gaussian function as kernel and the MLP model was composed of 2 fully connected layers of size 30, followed by a regression layer of size 1. Both models were trained and tested by applying the protocol described in Section III-B and the results are shown in Section III-C.

III. EXPERIMENTAL RESULTS

Our method is evaluated in terms of correlation with subjective judgments. To do so, we first present the datasets used and describe the protocol applied to train and test our CNN models. After defining the performance evaluation criteria employed, we analyze the results obtained by either applying symmetrization functions or pooling strategies. Finally, the performance of our method is compared with state-of-the-art PC metrics and a cross-dataset evaluation is carried-out to show the generalization ability of our method to predict the quality of unknown PCs.

A. Datasets

The proposed method is evaluated using two recent 3D point cloud datasets: sjtu [29] and ICIP20 [30].

- **sjtu** is composed of 9 point clouds from which 378 degraded versions were derived (i.e. 42 distorted PC per reference PC) through 6 degradation types (OT: Octree-based compression, CN: Color Noise, DS: Downscaling, D+C: Downscaling and Color noise, D+G: Downscaling and Geometry Gaussian noise, GGN: Geometry Gaussian noise and, C+G: Color noise and Geometry Gaussian noise).
- **ICIP20** is composed of 6 common used point clouds from which 90 degraded versions were derived (i.e. 15 distorted PC per reference PC) through 3 types of compression: V-PCC, G-PCC with triangle soup coding and G-PCC with octree coding. Each reference point cloud was compressed using five different levels.

These datasets give a subjective score (i.e. MOS) for each distorted PC that are used to train the CNN models and evaluate the performance of our method to predict the global quality score.

B. Protocol and performance evaluation

To assess the quality prediction effectiveness of our method, each dataset is split into training and test sets F times (i.e. F folds). Each fold is composed of a reference PC and its distorted versions. Therefore, the training and test sets do not contain same PC (i.e. no overlap between the two). At each time, $F - 1$ folds are used to train the models and the rest to test it. In this study, F is equal to 7 and 6 for sjtu and ICIP20, respectively.

Two evaluation criteria commonly used to evaluate the performance of quality metrics are adopted here: 1) Pearson Correlation Coefficient (PCC) and 2) Spearman Rank-Order Coefficient Correlation (SROCC). Both vary between 0 and 1 (i.e. absolute value) with 1 the best performance. These correlations are computed for each dataset over each fold and the mean correlations are then reported as results. It is worth noting that the same procedure was applied to the compared state-of-the-art metrics.

C. Performance analysis

In this section, we analyze the correlations obtained by each CNN model as well as their combination by either a symmetrization function (i.e. *min*, *mean* and *Wmean*) or using an SVR and a MLP. Table I shows the results obtained for each model and those obtained after symmetrization and pooling on sjtu dataset. Best results are highlighted in bold.

| Method | PCC | SROCC |
|---------------------------------------|--------------|--------------|
| Aggregated Patch Quality Index | | |
| APQI(X,Y) | 0.902 | 0.901 |
| APQI(Y,X) | 0.908 | 0.900 |
| Symmetrization function | | |
| GQI(X,Y)=min(APQI(X,Y),APQI(X,Y)) | 0.908 | 0.900 |
| GQI(X,Y)=mean(APQI(AB),APQI(X,Y)) | 0.908 | 0.902 |
| GQI(X,Y)=Wmean(APQI(X,Y),APQI(X,Y)) | 0.909 | 0.904 |
| Pooling (machine learning) | | |
| GQI(X,Y)=SVR(APQI(X,Y),APQI(X,Y)) | 0.918 | 0.907 |
| GQI(X,Y)=MLP(APQI(X,Y),APQI(X,Y)) | 0.927 | 0.906 |

TABLE I
MEAN CORRELATIONS OBTAINED ON SJTU DATASET BEFORE AND AFTER SYMMETRIZATION AND POOLING. BEST RESULTS ARE HIGHLIGHTED IN BOLD.

As can be seen, the aggregated indexes (i.e. APQI) given by computing the quality scores in both directions obtain close correlations. Their symmetrization through the three functions does not increase the global performance. Whereas, the pooling strategies applied improve the performance, particularly the linear correlations (i.e. PCC) without a significant increase in terms of mean SROCC. Both SVR and MLP models achieve close mean SROCC with a higher mean PCC for MLP. The improvement gains in terms mean PCC are about 1.8% and 2.8% for SVR and MLP, respectively. Based on these results, it seems that the symmetrization function, at least, does not always provide the optimal results and thus should be carefully employed.

D. Comparison with the state-of-the-art

Our method is here compared with point-based state-of-the-art metrics. More precisely, we consider po2pointMSE and po2planeMSE metrics that are pooled with MSE, PSNRpo2pointMSE and PSNRpo2planeMSE metrics that are pooled with PSNR as well as recent metrics po2dist [9] (i.e. point to distribution) pooled with MSE and PSNR. We also compare our method with a recent metric, called PCQM, which is based on both geometry and color features [8]. Tables II and III show the results obtained for sjtu and ICIP20 datasets, respectively. The best results are highlighted in bold.

On sjtu (see Table II), the proposed method performs the best with a performance gain in terms of mean PCC between 5% (compared to PCQM) and 79% (compared to po2pointHausdorff). Among the state-of-the-art metrics, PCQM achieves the best correlations with 0.879 and 0.888 as mean PCC and SROCC, respectively, far followed by

| Method | PCC | SROCC |
|-------------------------|--------------|--------------|
| po2pointMSE | 0.686 | 0.801 |
| PSNRpo2pointMSE | 0.799 | 0.844 |
| po2pointHausdorff | 0.517 | 0.686 |
| PSNRpo2pointHausdorff | 0.638 | 0.682 |
| po2planeMSE | 0.642 | 0.717 |
| PSNRpo2planeMSE | 0.744 | 0.722 |
| po2planeHausdorff | 0.539 | 0.682 |
| PSNRpo2planeHausdorff | 0.755 | 0.825 |
| po2distMSE (mmd) | 0.710 | 0.603 |
| PSNRpo2distMSE (mmd) | 0.621 | 0.603 |
| po2distMSE (msmd) | 0.706 | 0.603 |
| PSNRpo2distMSE (msmd) | 0.642 | 0.715 |
| PCQM | 0.879 | 0.888 |
| GQI (our method) | 0.927 | 0.906 |

TABLE II
COMPARISON WITH STATE-OF-THE-ART METHODS ON SJTU DATASET.
BEST RESULTS ARE HIGHLIGHTED IN BOLD.

PSNRpo2pointMSE and PSNRpo2planeMSE. All the compared metrics obtained a mean PCC and SROCC lower than 0.88 and 0.89, respectively. The worst result are obtained by po2pointHausdorff and po2planeHausdorff. The low correlations achieved by these metrics, except PCQM, can be explained by the fact that those metrics focus only on geometric aspects and therefore totally fail to capture other distortions like color noise.

| Method | PCC | SROCC |
|-------------------------|--------------|--------------|
| po2pointMSE | 0.945 | 0.950 |
| PSNRpo2pointMSE | 0.880 | 0.934 |
| po2pointHausdorff | 0.717 | 0.690 |
| PSNRpo2pointHausdorff | 0.597 | 0.763 |
| po2planeMSE | 0.945 | 0.959 |
| PSNRpo2planeMSE | 0.916 | 0.953 |
| po2planeHausdorff | 0.753 | 0.763 |
| PSNRpo2planeHausdorff | 0.939 | 0.970 |
| po2distMSE (mmd) | 0.965 | 0.963 |
| PSNRpo2distMSE (mmd) | 0.865 | 0.965 |
| po2distMSE (msmd) | 0.967 | 0.965 |
| PSNRpo2distMSE (msmd) | 0.902 | 0.972 |
| PCQM | 0.796 | 0.832 |
| GQI (our method) | 0.961 | 0.966 |

TABLE III
COMPARISON WITH STATE-OF-THE-ART METHODS ON ICIP20 DATASET.
BEST RESULTS ARE HIGHLIGHTED IN BOLD.

On ICIP20 (see Table III), po2distMSE (msmd) obtains the best mean PCC (0.967), closely followed by po2distMSE (mmd) (0.965). Whereas the two best mean SROCC is reached by PSNRpo2distMSE (0.972) and PSNRpo2planeHausdorff (0.970), respectively. Our metric obtains competitive results and surpasses most of the compared methods. PCQM is outperformed by most of the compared metrics, except po2point-based metrics pooled with Hausdorff (i.e. po2pointHausdorff and PSNRpo2pointHausdorff) and po2planeHausdorff. Similarly to the results obtained on sjtu, po2point-based and po2plane-based metrics pooled with MSE obtain higher correlations than those pooled with PSNR, while the po2dist-based metrics pooled with PSNR give better results than those pooled

with MSE. The high correlations obtain by the point-based metrics, especially po2pointMSE and po2planeMSE, can be explained by the fact that contrary to sjtu, this dataset contains only compressed PCs with joint distortion of geometry and attributes.

We also evaluate the impact of the random selection step on the performance by repeating the selection process 5 times on sjtu dataset. To this end, we computed the p-values and F-values between the correlations obtained over the 5 iterations using the One-way analysis of variance (ANOVA) test. A p-value close to 1 indicates that the difference is not high, while a F-value close to 0 indicates that the means and the standard deviations are similar. Table IV shows the obtained values. As can be seen, all the p-value are higher than the significance level (i.e. 0.05), indicating that there is no statistically significant difference between them. Whereas the F-values are close to 0, indicating that the correlation distributions are essentially identical.

| Iteration | 1 | 2 | 3 | 4 | 5 |
|-----------|------------------|------------------|------------------|------------------|------------------|
| 1 | 1.00/0.00 | 0.83/0.05 | 0.99/0.00 | 0.88/0.02 | 0.89/0.02 |
| 2 | 0.83/0.05 | 1.00/0.00 | 0.83/0.05 | 0.94/0.01 | 0.93/0.01 |
| 3 | 0.99/0.00 | 0.83/0.05 | 1.00/0.00 | 0.88/0.02 | 0.89/0.02 |
| 4 | 0.88/0.02 | 0.94/0.01 | 0.88/0.02 | 1.00/0.00 | 0.99/0.00 |
| 5 | 0.89/0.02 | 0.93/0.01 | 0.89/0.02 | 0.99/0.00 | 1.00/0.00 |

TABLE IV
P-VALUES AND F-VALUES BETWEEN THE CORRELATIONS OBTAINED OVER THE 5 RANDOM SELECTION (P-VALUE/F-SCORE).

E. Cross Dataset Evaluation

In this section, we evaluate the generalization ability of our method to predict the quality of unknown PCs using sjtu as training/validation set and ICIP20 as test set without overlap between both. More precisely, sjtu was split into two sets: 80% for the training and 20% for the validation. We obtain high correlations with 0.889 as mean PCC and 0.930 as mean SROCC. These results can be explained by the fact that ICIP20 is composed of only compressed PCs using G-PCC (octree and trisoup) and V-PCC, while sjtu contains compressed PCs using octree as well as PCs with several other distortions, including color noise. In other words, there is a data overlap in terms of distortion types between both datasets, which enables the trained models to well predict the quality. In addition, these results show that using such deep learning-based method allows to predict well the quality of unknown PCs with close unknown distortions. Indeed, training the considered CNN models on sjtu that contains PCs compressed with octree-based compression, even allows to well predict the quality of trisoup-based compressed PCs as well as those compressed with V-PCC. Nevertheless, due to the relatively wide variety of distortions contained in sjtu, training those CNN models on ICIP20 does not provide high correlations (~ 0.5). The latter result is related to the generalization ability of those deep learning models which is one of the challenging issues that remains open.

IV. CONCLUSION AND PERSPECTIVES

In this paper, we proposed a *deep learning-based* method that efficiently predicts the quality of distorted PCs with reference. We first randomly selecting a set of points from the reference PC and its distorted version and, determined the nearest neighbor of each of them. We then defined a region of size 32×32 for each of the two points and the structural distortions are computed through element-wise patch-based distances by considering three attributes: geometry, curvature and color. The resulted patches are then stacked into a patch of size $32 \times 32 \times 3$ and fed as input to CNN models to predict its quality (i.e. one CNN model per direction). The global quality index is finally given by aggregating the predicted patch quality indexes and applying either a symmetrization function or a pooling strategy using machine learning tools (SVR or MLP). The best result was provided by MLP and the use of symmetrization functions did not improve the performance. The results are compared with state-of-the-art methods and a cross-dataset evaluation was carried-out to show the generalization ability of our method to predict the quality of unknown PCs. The proposed method obtained promising results on two datasets and showed a good ability to predict unknown PCs with close unknown distortions. Despite the effectiveness of our method, some points should be deeper analyzed, including the use of more efficient deep learning models, the impact on the performance of the number of selected points as well as the patch size.

REFERENCES

- [1] M. Quach, G. Valenzise, and F. Dufaux, "Learning Convolutional Transforms for Lossy Point Cloud Geometry Compression," in *2019 IEEE Intl. Conf. on Image Process. (ICIP)*, Sep. 2019, pp. 4320–4324, iSSN: 1522-4880.
- [2] —, "Improved Deep Point Cloud Geometry Compression," in *2020 IEEE Intl. Workshop on Multimedia Signal Process. (MMSP)*, Oct. 2020. [Online]. Available: <http://arxiv.org/abs/2006.09043>
- [3] D. Tian *et al.*, "Geometric distortion metrics for point cloud compression," in *2017 IEEE Intl. Conf. on Image Process. (ICIP)*. Beijing: IEEE, Sep. 2017, pp. 3460–3464. [Online]. Available: <http://ieeexplore.ieee.org/document/8296925/>
- [4] E. Alexiou and T. Ebrahimi, "Point Cloud Quality Assessment Metric Based on Angular Similarity," in *2018 IEEE Intl. Conf. on Multimedia and Expo (ICME)*, Jul. 2018, pp. 1–6, iSSN: 1945-788X.
- [5] G. Meynet, J. Digne, and G. Lavoué, "PC-MSDM: A quality metric for 3D point clouds," in *2019 11th Intl. Conf. on Quality of Multimedia Experience (QoMEX)*, Jun. 2019, pp. 1–3, iSSN: 2472-7814, 2372-7179.
- [6] Z. Wang *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE Tran. on Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [7] G. Lavoué, "A Multiscale Metric for 3D Mesh Visual Quality Assessment," *Computer Graphics Forum*, vol. 30, pp. 1427–1437, Jul. 2011. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-01354459>
- [8] G. Meynet *et al.*, "PCQM: A Full-Reference Quality Metric for Colored 3D Point Clouds," in *2020 12th Intl. Conf. on Quality of Multimedia Experience (QoMEX 2020)*, Athlone, Ireland, May 2020. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-02529668>
- [9] A. Javaheri *et al.*, "Mahalanobis Based Point to Distribution Metric for Point Cloud Geometry Quality Evaluation," *IEEE Signal Process. Lett.*, vol. 27, pp. 1350–1354, 2020.
- [10] I. Viola, S. Subramanyam, and P. Cesar, "A Color-Based Objective Quality Metric for Point Cloud Contents," in *2020 12th Intl. Conf. on Quality of Multimedia Experience (QoMEX)*, May 2020, pp. 1–6, iSSN: 2472-7814.
- [11] E. Alexiou and T. Ebrahimi, "Towards a Point Cloud Structural Similarity Metric," in *2020 IEEE Intl. Conf. on Multimedia Expo Workshops (ICMEW)*, Jul. 2020, pp. 1–6.
- [12] A. Javaheri *et al.*, "Improving PSNR-based Quality Metrics Performance For Point Cloud Geometry," in *2020 IEEE Intl. Conf. on Image Process. (ICIP)*, Oct. 2020, pp. 3438–3442, iSSN: 2381-8549.
- [13] I. Abouelaziz, A. Chetouani, M. E. Hassouni, L. Latecki, and H. Cherifi, "Convolutional neural network for blind mesh visual quality assessment using 3d visual saliency," in *2018 IEEE International Conference on Image Processing, ICIP 2018, Athens, Greece, October 7-10, 2018*, 2018, pp. 3533–3537.
- [14] I. Abouelaziz, A. Chetouani, M. El Hassouni, L. Latecki, and H. Cherifi, "3D visual saliency and convolutional neural network for blind mesh quality assessment," *Neural Computing and Applications*, 2019. [Online]. Available: <https://hal-univ-orleans.archives-ouvertes.fr/hal-02384188>
- [15] A. Chetouani, "Three-dimensional mesh quality metric with reference based on a support vector regression model," *J. Electronic Imaging*, vol. 27, no. 04, p. 043048, 2018. [Online]. Available: <https://doi.org/10.1117/1.JEL.27.4.043048>
- [16] A. Chetouani, M. Quach, G. Valenzise, and F. Dufaux, "Deep learning-based blind quality assessment of 3D point clouds," in *IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, Shenzhen (virtual), China, Jul. 2021. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-03198099>
- [17] L. Kang, P. Ye, Y. Li, and D. Doermann, "Convolutional neural networks for no-reference image quality assessment," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1733–1740.
- [18] S. Bosse, D. Maniry, K. Müller, T. Wiegand, and W. Samek, "Deep neural networks for no-reference and full-reference image quality assessment," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 206–219, Jan 2018.
- [19] A. Krizhevsky, "One weird trick for parallelizing convolutional neural networks," *CoRR, abs/1404.5997*, 2014.
- [20] S. R. K. He, X. Zhang and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.
- [21] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.
- [22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR, abs/1409.1556*, 2014.
- [23] S. Dodge and L. Karam, "Understanding how image quality affects deep neural networks," in *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, 2016, pp. 1–6.
- [24] O. Messai, A. Chetouani, F. Hachouf, and Z. Seghir, "No-reference stereoscopic image quality predictor using deep features from cyclopean image," 01 2021.
- [25] S. J. Daly, "Visible differences predictor: an algorithm for the assessment of image fidelity," in *Human Vision, Visual Processing, and Digital Display III*, B. E. Rogowitz, Ed., vol. 1666, International Society for Optics and Photonics. SPIE, 1992, pp. 2 – 15. [Online]. Available: <https://doi.org/10.1117/12.135952>
- [26] A. Chetouani and L. Li, "On the use of a scanpath predictor and convolutional neural network for blind image quality assessment," *Signal Processing: Image Communication*, vol. 89, p. 115963, 2020. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0923596520301405>
- [27] A. Chetouani, "Image quality assessment without reference by mixing deep learning-based features," in *IEEE International Conference on Multimedia and Expo, ICME 2020, London, UK, July 6-10, 2020*. IEEE, 2020, pp. 1–6. [Online]. Available: <https://doi.org/10.1109/ICME46284.2020.9102903>
- [28] 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, p. 107174, 2020. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0031320319304741>
- [29] Q. Yang, 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*, 2020.
- [30] S. Perry *et al.*, "Quality Evaluation Of Static Point Clouds Encoded Using MPEG Codecs," in *2020 IEEE Intl. Conf. on Image Process. (ICIP)*, Oct. 2020, pp. 3428–3432, iSSN: 2381-8549.