Deep learning-based blind quality assessment of 3D point clouds without reference

Aladine Chetouani, Maurice Quach, Giuseppe Valenzise, Frédéric Dufaux

DEEP LEARNING-BASED QUALITY ASSESSMENT OF 3D POINT CLOUDS WITHOUT REFERENCE

Aladine Chetouani[1,2] *Maurice Quach*[2] *Giuseppe Valenzise*[2] *Frédéric Dufaux*[2]

[1]Laboratoire PRISME, Université d'Orléans, Orléans, France

[2]L2S, Centrale Supélec, Université Paris-Saclay, Gif-sur-Yvette, France

ABSTRACT

Point cloud (PC) quality assessment is of fundamental importance to enable the efficient processing, coding and transmission of 3D data for virtual/augmented reality, autonomous driving, cultural heritage, etc. The quality metrics proposed so far aim at quantifying the distortion in the PC geometry and/or attributes with respect to a reference pristine point cloud, using simple features extracted by the points. In this work, we target instead a *blind* (no-reference) scenario in which the original point cloud is not available. In addition, we learn features from data using deep neural networks. Given the limited availability of subjectively annotated datasets of corrupted point clouds, and the consequent difficulty to learn in an end-to-end fashion PC quality features, in this work we use instead a two-step procedure. First, we extract from local patches three relevant low-level features which have been commonly used in other PC quality metrics, i.e., geometric distance, local curvature and luminance values. Afterwards, we employ a deep neural network to learn, from these low-level features, a mapping to the PC ground truth mean opinion score. Our results on two state-of-the-art PC quality datasets show the potential of the proposed approach. The code is available here : `https://github.com/achetouani/NR-CNN-3D-PC`

Index Terms— Point clouds, quality assessment, deep learning, no-reference

1. INTRODUCTION

3D point clouds (PC) are becoming and increasing popular data format for applications such as virtual and augmented reality, autonomous vehicles, cultural heritage, immersive communication, etc. PCs can undergo different types of distortion, due to acquisition, compression, transmission or rendering. Predicting and quantifying the perceptual impact of this distortion is therefore of paramount importance for the deployment of this kind of data. In the past few years, there has been an increasing interest in the multimedia research community towards the definition of accurate quality metrics for point clouds. These efforts have been also partially motivated by the concurrent standardization activities for point cloud coding in MPEG [1], as well as due to the need of relevant loss functions in learning-based PC compression methods [2, 3].

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 [4]. Starting from these approaches, several point-based metrics have been then developed. In [5], 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 [6], the authors proposed a metric called PC-MSDM by extending the well-known SSIM metric [7], widely used for 2D images, to PC, by considering features including local mean curvature as they previously did for 3D meshes [8]. The authors proposed later a metric called PCQM that integrates color information [9]. In [10], 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 [11], the authors proposed a color-focused metric that integrates geometry information. In [12], the authors adapted also the SSIM metric for point clouds using a number of features, while in [13] the authors improved the point cloud PSNR metric. The authors of [14] propose a different approach, by studying the effectiveness of learned features for the prediction of PC quality, obtaining some encouraging results. Interesting approaches for 3D meshes have also been proposed [15].

In this paper, we present a *deep learning-based* framework that efficiently predicts the quality of PCs without reference. The proposed method consists first in extracting patches around a set of points from the PC. The regions delimited by the extracted patches are then characterized by three at-

Funded by the ANR ReVeRy national fund (REVERY ANR-17-CE23-0020).

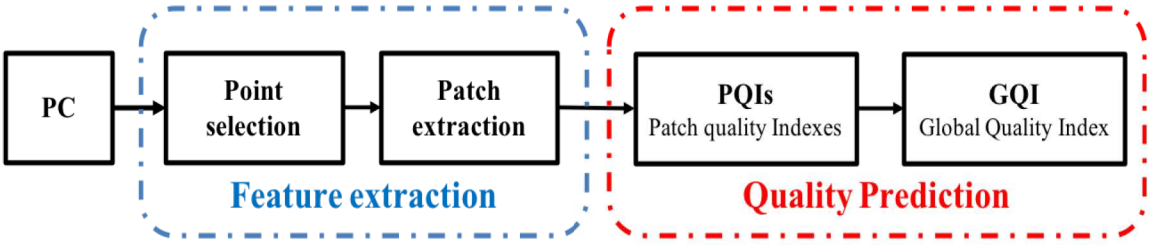

Fig. 1. General framework of the proposed method.

tributes: geometric distance, mean curvature and gray-level. The resulted patches (i.e., one per attribute) are stacked and fed as input to a Convolutional Neural Network (CNN) model to predict their quality. The global quality index is finally given by aggregating the predicted patch quality indexes. We assess the performance of our method on two datasets, including a large dataset more suited to deep models. The high correlations obtained showed the potential of the approach. The main contribution of this paper is the use of classical CNN models to predict the quality of 3D PC through extracted features.

The rest of this paper is structured as follows: Section 2 describes the proposed framework. Experimental results are presented in Section 3 and the conclusion is given in Section 4. s

2. PROPOSED METHOD

As illustrated by Fig. 1, the framework proposed for estimating the quality of 3D point clouds without reference is based on 2 main steps: 1) feature extraction that aims to characterize the PC through a set of features as patches from selected points and, 2) quality prediction that aims to estimate the quality of the distorted PC via a CNN model. Both steps are described in details in this section.

2.1. Feature extraction as patches

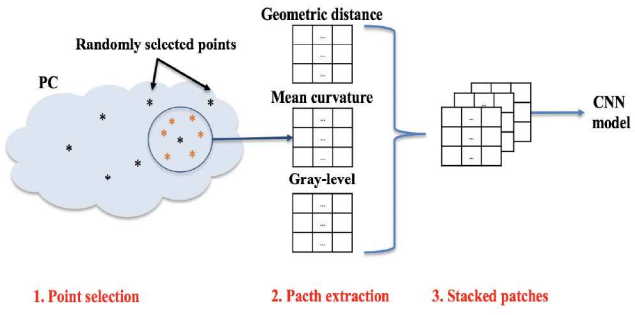

Fig. 2. Random point selection and patch extraction.

Fig. 2 details the feature extraction process. From a given distorted PC, we first randomly select a set of N points. The latter are here considered as reference points from which the quality is predicted. Then, we delimit a region around each point N_i by finding its K nearest neighbors. Next, the neighbors of each point N_i are characterized by three attributes: geometric distance, mean curvature and gray-level. The geometric distance D describes the spatial distribution of the neighbors and it is computed as follows:

$$D\left(K_j^i, N_i\right) = \sqrt{\left(x_{K_j^i} - x_{N_i}\right)^2 + \left(y_{K_j^i} - y_{N_i}\right)^2 + \left(z_{K_j^i} - z_{N_i}\right)^2}, \quad (1)$$

where $D\left(K_j^i, N_i\right)$ represents the Euclidean distance between N_i and its j^{th} nearest neighbor K_j^i. $\{x, y, z\}_{K_j^i}$ and $\{x, y, z\}_{N_i}$ are the spatial coordinates of the j^{th} neighbor and the point N_i, respectively.

The mean curvature allows to characterize the shape variations and it is computed for each point of the delimited region, including the point N_i. The latter is estimated through a quadric fitting. The grey level gives information more related to the perceived rendering and it derived from the color information (i.e. RGB to grey level). The features obtained for each attribute are encapsulated into a patch and the three resulted patches are then stacked to form a new patch SP_i of size $P \times P \times 3$ where three represents the number of attributes considered.

2.2. Quality prediction

Fig. 2 gives a detailed overview of the steps applied to predict the Global Quality Index (GQI) of the PC from the stacked patches SP_i.

2.2.1. CNN models and patch quality indexes

For each stacked patch SP_i, we first predict its Patch Quality Index (i.e. PQI_i) through a CNN model. To do so, we compared here 5 pre-trained models that were widely used for classification tasks and some of them were successfully employed in several studies related to quality assessment as well [16]. It is worth noting that these models were considered in

order to compare the performance of different architectures and analyze the impact of the depth.

- **AlexNet [17]:** This model is one of the pioneering models proposed by Alex Krizhevsky. It highlighted the relevance of using CNN models for classification tasks. The authors brought out three main points: the use of the Relu (Rectified Linear Units) function, the exploitation of the dropout to prevent the over-fitting and overlap during the pooling step.
- **VGG [18]:** This model was developed by the Oxford Visual Geometry Group in 2014. It is characterized by applying a max pooling layer after a succession of convolution layers. Here, we used $VGG16$ and $VGG19$ which are composed of 16 and 19 layers, respectively.
- **ResNet [19]:** This model was introduced in 2015 and it stands out by the integration of a residual module. The authors proposed to reformulate the output (H(x)=F(x)) of each series of Conv-ReLu-Conv by adding the input $\boldsymbol{x}$ as information ($\boldsymbol{H(x) = F(x)+x}$). $ResNet18$ and $ResNet50$ are used in this study.

The above-models were adapted to match their inputs and outputs to our context. We changed the size of their input layers and replaced their FC layers with three FC layers of size 64, 64 and 1, respectively. The two first FC layers are followed by a ReLu layer, while the last FC layer is preceded by a a dropout layer with a probability fixed to 0.5 and aims to predict the PQIs of the stacked patches (i.e. regression layer). We finally fine-tuned each model to adapt their weights to our context. It is worth noting that the target of each stacked patch was the subjective score of the whole distorted PC as commonly employed to estimate the quality of several multimedia content for 2D images [20, 21] as well as for stereo images [22] and 3D meshes [23].

During the training step, the stochastic gradient descent optimization algorithm was used with the Mean Squared Error (MSE) as loss function. The batch size, learning rate and the momentum were set to 64, 0.0001 and 0.9, respectively. We fixed the number of epochs to 30 with a validation each 5000 iterations. At each epoch, the training data was shuffled and the resulted model was saved. The model that provided the best results was finally retained.

2.2.2. *Global Quality Index*

Figure 3 illustrates how to obtain a Global Quality Index (GQI) that reflects the quality of the whole distorted PC by aggregating the predicted PQIs. To do so, we aggregate the obtained PQIs by computing the average scores as follows:

$$GQI = \frac{1}{N}\sum_{i=1}^{N} PQI_i, \tag{2}$$

where $\boldsymbol{N}$ is the number of stacked patches (i.e. the number of selected points).

It is worth noting that a plethora of handcrafted and deep-based methods already aggregated local predicted scores to derive a global quality index like SSIM [7], VDP [24] and the following deep-based methods as well [20, 25]. This strategy was also applied for 3D content as done in [6, 8].

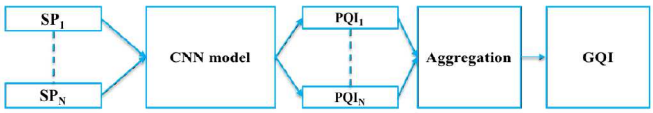

Fig. 3. Prediction of the Global Quality Index (GQI) from the stacked patches SP_i.

3. EXPERIMENTAL RESULTS

In this section, we evaluate the capacity of our method to predict subjective scores. For that, we first present the datasets used and describe the protocol applied to train and test our CNN models. After defining the performance evaluation criteria used, we discuss the results obtained by each considered CNN model and compare the best results with state-of-the-art PC metrics.

3.1. Datasets and experimental setup

The proposed method is evaluated using two recent 3D point cloud datasets: sjtu [26] and ICIP20 [27].

- **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 commonly 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.

Our method is evaluated by decomposing each dataset into F folds. At each fold, one reference PC and its distorted versions are used to test the model, while the rest are used to train the model. The number of folds is here equal to 7 for sjtu and and 6 for ICIP20.

In this study, the number of selected points (N) was set to 1000, while the number of neighbors (K) and the size of the

patch (P) were fixed to 1023 and $32x32$, respectively. The patch size was set following studies where its impact on the performance was analyzed [20, 25].

3.2. Performance evaluation

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 in absolute value, with 1 being 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.

3.3. Model comparison

Model	PCC	SROCC
AlexNet	0.894 ± 0.055	0.874 ± 0.083
VGG16	$\mathbf{0.923 \pm 0.031}$	$\mathbf{0.907 \pm 0.045}$
VGG19	$\mathbf{0.925 \pm 0.035}$	$\mathbf{0.912 \pm 0.042}$
ResNet18	0.886 ± 0.091	0.856 ± 0.113
ResNet50	0.885 ± 0.086	0.855 ± 0.109

Table 1. Model comparison on sjtu dataset. The two best results are highlighted in bold.

Table 1 shows the mean correlations obtained for each considered model. The two best results are highlighted in bold. From these results, several observations can be made:

- First of all, the performance differs from a model to another with a high gap between them (from 0.875 to 0.925 in terms of mean PCC and from 0.855 to 0.912 in terms of mean SROCC). The stability of the results over the folds differs as well. ResNet-based models obtain the higher standard deviations outperformed by AlexNet, while VGG-based models are more stable with the lower standard-deviations.

- ResNet50 achieves the worst performance, lower even than the simpler and shallower model used (i.e. AlexNet). However, it is competitive to the full reference metric PCQM and outperforms all the other full reference compared metrics on sjtu (see Table 2).

- For VGG-based and ResNet-based models, we can see that the depth doesn't highly impact the performance since the differences are statistically not significant (p-value > 0.05).

- The architecture seems to have an impact since VGG-based models give the best correlations with an improvement mean PCC gain around 5%. Moreover, unlike for classification tasks the integration of residual modules doesn't improve the performance. The p-values between the VGG-based and ResNet-based models are smaller than 0.05, indicating a statistically significant difference between the two.

Based on these results, it seems that the architecture of the considered models has more impact on the performance than the depth.

3.4. Comparison with state-of-the-art metrics

Our method is here compared with state-of-the-art metrics. More precisely, we consider po2point-based and po2plane-based metrics pooled with MSE, PSNR and Hausdorff. We also considered recent metrics po2dist [10] (i.e. point to distribution) pooled with MSE and PSNR as well as PCQM which is based on both geometry and color features [9]. Results on sjtu and ICIP20 datasets are shown in Tables 2 and 3, respectively with the top-2 results highlighted in bold.

Method	PCC	SROCC
Baseline methods (**full reference**)		
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
Our method (**no reference**)		
GQI-VGG16	**0.923**	**0.907**
GQI-VGG19	**0.925**	**0.912**

Table 2. Comparison with state-of-the-art methods on sjtu dataset. The two best results are highlighted in bold.

On sjtu (see Table 2), the results obtained by our method with both models (i.e. VGG16 and VGG19) surpass all the compared state-of-the-art metrics with a

gain in terms of mean PCC that varies between 5% and 70%. PCQM achieves the third best performance with 0.879 and 0.888 as mean PCC and SROCC, respectively. The worst result is obtained by po2pointHausdorff, closely followed by po2planeHausdorff. PSNRpo2pointMSE, PSNRpo2planeHausdorff and po2distMSE obtain the best results among the po2point-based, po2plane-based and po2dist-based metrics, respectively. Whereas po2pointHausdorff, po2planeHausdorff and PSNRpo2distMSE achieve the worst results among the po2point-based, po2plane-based and po2dist-based metrics, respectively. Unlike the po2dist-based metrics, po2point-based and po2plane-based metrics pooled with MSE obtain higher correlations than those pooled with PSNR. Globally, the compared metrics, except PCQM, achieve low correlations since they focus more geometric information, failing to catch other distortions.

Method	PCC	SROCC
Baseline methods **(full reference)**		
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
Our method **(no reference)**		
GQI-VGG16	0.956	0.966
GQI-VGG19	0.952	0.966

Table 3. Comparison with state-of-the-art methods on ICIP20 dataset. The two best results are highlighted in bold.

On ICIP20 (see Table 3), 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 achieves the third mean PCC (0.956 for VGG16 and 0.952 for VGG19) and the fourth mean SROCC (0.966 for both models) without accessing the pristine PC (i.e. NR approach). 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.

Globally, the correlations reached on ICIP20 are higher than those obtained on sjtu, except for PCQM. These results can be justified by the fact that ICIP20 is composed only of compressed PCs with joint distortion of geometry and attributes, while sjtu contains a more wide set of distortions including color noise. We also evaluated the generalization capacity of the proposed method. However, the correlations are not as expected.

4. CONCLUSION

In this paper, we proposed a *deep learning-based* method that efficiently predicts the quality of distorted PCs without reference. After randomly selecting a set of points from the PC, we defined a region around each of them by finding the nearest neighbors. The delimited regions are then characterized through three attributes (geometric-distance, mean curvature and gray-level) and the obtained values are stacked into patches of size $32 \times 32 \times 3$ to predict their quality indexes (i.e. patch quality indexes) through a CNN model. The global quality index is finally given by aggregating the predicted patch quality indexes. We compared the performance of five pre-trained CNN models and the best results were compared with state-of-the-art methods. The results obtained on two datasets show the potential of the proposed approach.

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. Other point selection strategies will also be considered.

5. REFERENCES

[1] D. Graziosi, O. Nakagami, S. Kuma, A. Zaghetto, T. Suzuki, and A. Tabatabai, "An overview of ongoing point cloud compression standardization activities: video-based (V-PCC) and geometry-based (G-PCC)," *APSIPA Trans. on Signal and Information Process.*, vol. 9, 2020.

[2] Maurice Quach, Giuseppe Valenzise, and Frederic Dufaux, "Learning Convolutional Transforms for Lossy Point Cloud Geometry Compression," in *2019 IEEE Intl. Conf. on Image Process. (ICIP)*, Sept. 2019, pp. 4320–4324, ISSN: 1522-4880.

[3] Maurice Quach, Giuseppe Valenzise, and Frederic Dufaux, "Improved Deep Point Cloud Geometry Compres-

sion," in *2020 IEEE Intl. Workshop on Multimedia Signal Process. (MMSP)*, Oct. 2020.

[4] Dong Tian et al., "Geometric distortion metrics for point cloud compression," in *2017 IEEE Intl. Conf. on Image Process. (ICIP)*, Beijing, Sept. 2017, pp. 3460–3464, IEEE.

[5] Evangelos Alexiou and Touradj Ebrahimi, "Point Cloud Quality Assessment Metric Based on Angular Similarity," in *2018 IEEE Intl. Conf. on Multimedia and Expo (ICME)*, July 2018, pp. 1–6, ISSN: 1945-788X.

[6] Gabriel Meynet, Julie Digne, and Guillaume Lavoué, "PC-MSDM: A quality metric for 3D point clouds," in *2019 11th Intl. Conf. on Quality of Multimedia Experience (QoMEX)*, June 2019, pp. 1–3, ISSN: 2472-7814, 2372-7179.

[7] Zhou 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.

[8] Guillaume Lavoué, "A Multiscale Metric for 3D Mesh Visual Quality Assessment," *Computer Graphics Forum*, vol. 30, pp. 1427–1437, July 2011.

[9] Gabriel 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.

[10] 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.

[11] 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.

[12] E. Alexiou and T. Ebrahimi, "Towards a Point Cloud Structural Similarity Metric," in *2020 IEEE Intl. Conf. on Multimedia Expo Workshops (ICMEW)*, July 2020, pp. 1–6.

[13] 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.

[14] Maurice Quach, Aladine Chetouani, Giuseppe Valenzise, and Frédéric Dufaux, "A deep perceptual metric for 3D point clouds," in *Image Quality and System Performance, IS&T International Symposium on Electronic Imaging (EI 2021)*, San Francisco, United States, Jan. 2021.

[15] Ilyass Abouelaziz, Aladine Chetouani, Mohammed El Hassouni, Longin Jan Latecki, and Hocine 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.

[16] 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.

[17] A. Krizhevsky, "One weird trick for parallelizing convolutional neural networks," *CoRR, abs/1404.5997*, 2014.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR, abs/1409.1556*, 2014.

[19] S. Ren K. He, X. Zhang and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[20] 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.

[21] Aladine Chetouani, "A blind image quality metric using a selection of relevant patches based on convolutional neural network," in *2018 26th European Signal Processing Conference (EUSIPCO)*, 2018, pp. 1452–1456.

[22] Oussama Messai, Aladine Chetouani, F. Hachouf, and Zianou Seghir, "No-reference stereoscopic image quality predictor using deep features from cyclopean image," 01 2021.

[23] Ilyass Abouelaziz, Aladine Chetouani, Mohammed El Hassouni, LJ Latecki, and Hocine Cherifi, "3D visual saliency and convolutional neural network for blind mesh quality assessment," *Neural Computing and Applications*, 2019.

[24] Scott J. Daly, "Visible differences predictor: an algorithm for the assessment of image fidelity," in *Human Vision, Visual Processing, and Digital Display III*, Bernice E. Rogowitz, Ed. International Society for Optics and Photonics, 1992, vol. 1666, pp. 2 – 15, SPIE.

[25] S. Bosse, D. Maniry, K. Müller, T. Wiegand, and W. Samek, "Deep neural networks for no-reference and full-reference image quality assessment," *IEEE Trans. on Image Process.*, vol. 27, no. 1, pp. 206–219, Jan 2018.

[26] 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 Trans. on Multimedia*, 2020.

[27] Stuart 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.