A VIEWPORT-DRIVEN MULTI-METRIC FUSION APPROACH FOR 360-DEGREE VIDEO QUALITY ASSESSMENT

Roberto G. de A. Azevedo, Neil Birkbeck, Ivan Janatra, Balu Adsumilli, Pascal Frossard

ABSTRACT

We propose a new viewport-based multi-metric fusion (MMF) approach for visual quality assessment of 360-degree (omnidirectional) videos. Our method is based on computing multiple spatio-temporal objective quality metrics (features) on viewports extracted from 360-degree videos, and learning a model that combines these features into a metric that closely matches subjective quality scores. The main motivations for the proposed method are that: 1) quality metrics computed on viewports better captures the user experience than metrics computed on the projection domain; 2) no individual objective image quality metric always performs the best for all types of visual distortions, while a learned combination of them is able to adapt to different conditions. Experimental results, based on the largest available 360-degree videos quality dataset, demonstrate that the proposed metric outperforms state-of-the-art 360-degree and 2D video quality metrics.

Index Terms— visual quality assessment, omnidirectional video, 360-degree video, multi-metric fusion

1. INTRODUCTION

Omnidirectional (or 360-degree) visual content are spherical signals captured by cameras with a full 360-degree field-of-view (FoV). In particular, when consumed via head-mounted displays (HMDs), 360-degree content supports immersive experiences. At each time instant, the portion of the sphere in the user's field of view, i.e., the *viewport*, is rendered and presented to the user. The viewports (one per eye) are seamlessly updated following the user's head movements, which provides an increased sense of presence. Similarly to traditional audiovisual multimedia content, quality assessment of omnidirectional content plays a central role in shaping processing algorithms and systems, as well as their implementation, optimization, and testing.

Quality assessment of processed 360-degree visual content consumed through HMDs brings its own specificities compared to the assessment of 2D or stereoscopic 3D visual content. For instance, to reuse existing image and video processing technologies, the 360-degree visual content is generally to a 2D plane (the projection domain) and stored as a rectangular image [1, 2]. Examples of projections include: equirectangular (ERP), cube map (CMP), and equiangular cube map (EAC) [3]. The coupled interaction between projection and compression of the resulting rectangular images, however, brings new types of visual distortions [1]. Also, the magnification of the content, the supported increased field-of-view, the fact that the user is completely immersed, and the new interactive dimension, all contribute to changing the end user perceived quality of experience (QoE) [1]. Such new features call for the development of new methods and good practices related to the quality and QoE assessment of 360-degree visual content.

New subjective evaluation methodologies and objective quality metrics specifically developed for 360-degree video have been proposed recently [1, 2]. Subjective VQA methods collect quality judgments from human viewers through psychophysical experiments. They have the advantage of being more reliable (since humans are the ultimate receiver of the multimedia content) but they are expensive, time-consuming, and not suitable for real-time deployment. Thus, objective quality assessment algorithms are required to estimate quality automatically. PSNR-based objective image quality assessment (IQA) metrics that take into account the properties of 360-degree images have been recently proposed in the literature, such as S-PSNR [4] and WS-PSNR [5]. Such methods are easy to implement and can be efficiently integrated into video coders, but their correlation with subjective judgements are far from satisfactory. Moreover, when used for video quality assessment they lack a proper modelling of the temporal characteristics of the human-visual system (HVS). Therefore, more perceptual-oriented IQA metrics are still required for the objective evaluation of 360-degree video quality.

Differently from current proposals, this paper proposes a viewport-based multi-metric fusion (MMF) approach for 360-VQA. The proposed approach is based on extracting spatio-temporal quality features (i.e., computing objective IQA metrics) from viewports, temporally pooling them taking the characteristics of the human-visual system (HVS) into consideration, and then training a random forest regression model to predict the 360-degree video quality. On the one hand, working with viewports allow us to better account for the final viewed content and naturally supports different projections [6, 7]. On the other hand, the use of multiple objec-

R. Azevedo and P. Frossard are with Signal Processing Lab. (LTS4), École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland.

N. Birkbeck, I. Janatra, and B. Adsumilli are with Youtube, Mountain View, California, USA.

tive metrics computed on these viewports allow our method to have a good performance for the complex and diversifying nature of visual distortions appearing in 360-degree videos. Indeed, previous work in both traditional 2D [8, 9, 10] and 360-degree [7] VQA have recognized that even with the multitude of available objective IQA metrics, there is no single one that always performs best for all distortions. The combination of multiple metrics is thus a promising approach that can take advantages of the power of individual metrics to correlate with subjective scores on different distortions [11, 10]. Experimental results, based on the largest publicly available 360-degree video quality dataset, VQA-ODV [12], show the viability of our proposal, which outperforms state-of-the-art methods for 360-degree quality assessment.

Section 2 presents the related work. Section 3 describes our proposal. Section 4 presents the experimental setup used to validate the proposed approach and the experimental results. Finally, Section 5 brings our conclusions.

2. RELATED WORK

The use of standard 2D image/video objective metrics for quality assessment in the projection domain is straightforward, but it does not properly model the perceived quality of the 360-degree visual content. The main issues with such an approach are that: (1) it gives the same importance to the different parts of the spherical signal, which besides being sampled very differently from classical images, also have different viewing probabilities; (2) even for traditional images, these metrics are known to have limitations for different visual distortion types; hence, none is universally satisfactory.

To cope with the sampling issue of the projection domain, recent proposals have been developed to tackle the specific geometry of 360-degree images: S-PSNR [4], CPP-PSNR [13], and WS-PSNR [5]. In S-PSNR (Spherical PSNR), sampling points uniformly distributed on a spherical surface are re-projected to the original and distorted images respectively to find the corresponding pixels, followed by the PSNR calculation. In CPP-PSNR (Craster Parabolic Projection PSNR), PSNR is computed between samples in the Craster Parabolic Projection (CPP) domain, in which pixel distribution is close to a uniform one in the spherical domain. In WS-PSNR (Weighted-to-Spherical PSNR), the PSNR computation at each sample is performed directly in the planar domain, but its value is weighted by the area covered by that sample on the sphere. The use of viewports [6, 7] and voronoi patches [14] for computing individual IQA metrics have also been discussed, which we acknowledge as more perceptually-correct ways of assessing 360 visual quality. These methods, however, simply compute the quality of the 360 image as the average of the viewports (or patches).

Recent works have also proposed deep learning architectures to estimate 360-degree video quality [12, 15, 16]. One of the main issues with such approaches is that the current 360-VQA datasets are not big enough to satisfactorily train deep learning methods. Thus, they need to perform data augmentation, such as splitting the original image into patches or rotating the original 360-degree images. In both cases, however, it is not clear if the new generated patches or rotated images share the same quality scores of the original content.

Finally, all the metrics proposed for 360-VQA mentioned above do not explicitly model the temporal dimension of 360-degree videos. They usually compute the overall quality simply as the average of the quality of each individual frame. Different from IQA, however, VQA metrics shall ideally take the temporal dimension into consideration and properly integrate the temporal properties of the HVS. VQM [17], MOVIE [18], and Vis3 [19] are examples of metrics developed for 2D VQA that take the temporal dimension into consideration. However, these methods does not consider 360 videos specifically.

We address the above mentioned issues by computing IQA-based objective metrics in the viewport domain, temporally pooling them taking into account the HVS, and employing a multi-metric fusion approach that closely match subjective scores. Being a multi-metric fusion approach, our proposal shares some of the principles of such approaches, e.g., VMAF [9], one of the most successful metrics for traditional videos, but it takes into account the specific features of 360-degree videos. Moreover, compared to VMAF, our method uses a different set of individual spatial and temporal features and a temporal pooling method which support a better correlation to subjective tests. Finally, our proposal uses a random forest regression model whereas VMAF uses support vector regression, since our preliminary tests indicated that the random forest regression provide more robust results.

3. VIEWPORTS-BASED MMF FOR 360-VQA

Fig. 1 shows our proposed 360-VQA approach. The possible space of visible viewports is represented by using N viewports from different viewing directions. Both the original and the distorted content are rendered for each viewport and 2D objective metrics are computed individually within the viewport and then temporally pooled using an HVS-based method that considers the temporal quality variation. Finally, based on the per-viewport pooled scores, we train a random forest model that is able to learn a combination of the individual objective metrics into a new objective metric that closely relates to subjective scores. In what follows, let $\mathcal{R} = \{R_f, f = 0, 1, ..., F-1\}$ and $\mathcal{D} = \{D_f, f = 0, 1, ..., F-1\}$ respectively be the reference and distorted sequences of the same 360-degree video content in the projection domain. R_f and D_f denote the f'th frame of $\mathcal{R}$ and $\mathcal{D}$, respectively, and F the total number of frames in both $\mathcal{R}$ and $\mathcal{D}$.

3.1. Viewports sampling

First, for each frame f, we compute a set of viewports $\mathcal{V}_f^{\mathcal{R}} = \{V_f^{\mathcal{R},0}, ..., V_f^{\mathcal{R},N-1}\}$ and $\mathcal{V}_f^{\mathcal{D}} = \{\mathcal{V}_f^{\mathcal{D},0}, ..., V_f^{\mathcal{D},N-1}\}$, for the respective reference and distorted frames. A viewport (Fig. 2) is the gnomonic projection [20] of the omnidirectional signal

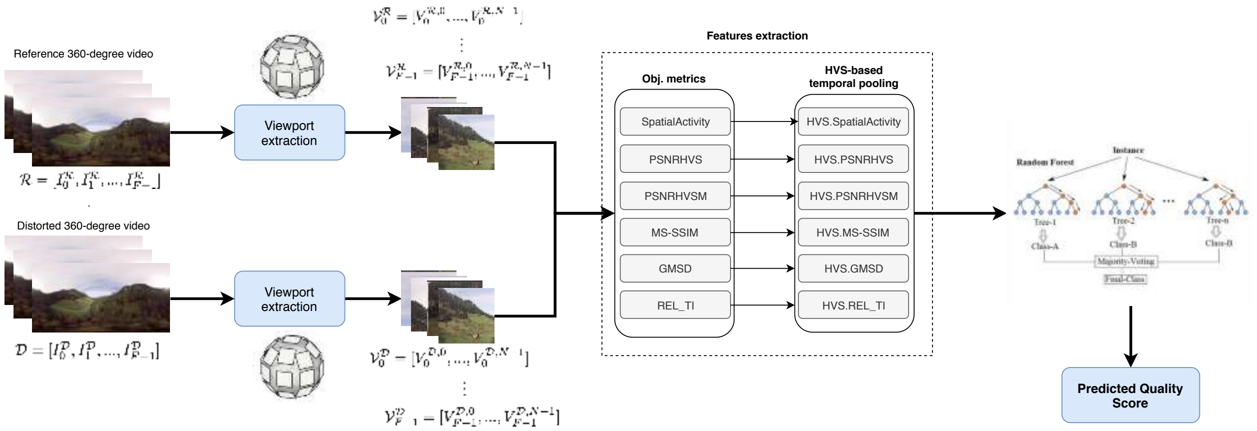
Reference 360-degree video
$\mathcal{R} = [I_0^{\mathcal{R}}, I_1^{\mathcal{R}}, ..., I_{F-1}^{\mathcal{R}}]$
Distorted 360-degree video
$\mathcal{D} = [I_0^{\mathcal{D}}, I_1^{\mathcal{D}}, ..., I_{F-1}^{\mathcal{D}}]$
Viewport extraction
Viewport extraction
$\mathcal{V}_0^{\mathcal{R}} = [V_0^{\mathcal{R},0}, ..., V_0^{\mathcal{R},N-1}]$
$\mathcal{V}_{F-1}^{\mathcal{R}} = [V_{F-1}^{\mathcal{R},0}, ..., V_{F-1}^{\mathcal{R},N-1}]$
$\mathcal{V}_0^{\mathcal{D}} = [V_0^{\mathcal{D},0}, ..., V_0^{\mathcal{D},N-1}]$
$\mathcal{V}_{F-1}^{\mathcal{D}} = [V_{F-1}^{\mathcal{D},0}, ..., V_{F-1}^{\mathcal{D},N-1}]$
Features extraction
Obj. metrics
SpatialActivity
PSNRHVS
PSNRHVSM
MS-SSIM
GMSD
REL_TI
HVS-based temporal pooling
HVS.SpatialActivity
HVS.PSNRHVS
HVS.PSNRHVSM
HVS.MS-SSIM
HVS.GMSD
HVS.REL_TI
Random Forest
Instance
Tree-1
Tree-2
Tree-n
Class-A
Class-B
Class-B
Majority-Voting
Final-Class
Predicted Quality Score

Fig. 1: Overview of the proposed 360-VQA multi-metrics fusion approach.

to a plane tangent to the sphere. It is defined by: the viewing direction (el_o, az_o), which specifies the center O' where the viewport is tangent to the sphere; its resolution $[vp_w, vp_h]$; and its horizontal and vertical field-of-view, FoV$_h$ and FoV$_v$, respectively. When considering a viewport-based metric for 360-degree videos, we need to define a process that given an omnidirectional image, I, and the viewports parameters vp_w, vp_h, FoV_w and FoV_h, generates N viewports from different viewing directions (i.e., different O's).

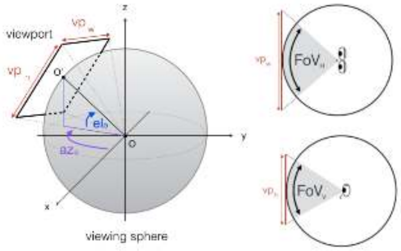

Fig. 2: Viewport parameters [21].

Fig. 3 shows three examples of the viewport sampling process: *uniform*, *tropical*, and *equatorial* [6], which sample respectively 9, 16, and 25 viewports. For a specific sampling procedure, larger FoVs might result in both larger overlapped regions between the viewports and larger geometry distortions. On the one hand, duplicated content can increase the relative importance of such duplicated areas when computing objective metrics on the viewports. On the other hand, smaller FoVs might more viewports to completely cover the sphere area, whichalso results in higher computational costs. Based on preliminary experiments, we have found that the uniform sampling with a FoV of 40-degree provides a good trade-off between the number of viewports and the overlapped regions, which is used in the experiments of Section 4. The viewport resolution $[vp_w, vp_h]$ should also ideally be the same as the HMD resolution used to visualize the content.

3.2. Features

Based on the previously computed viewports, we compute for each pair of reference and distorted viewports, $V_f^{r,n}$ and $V_f^{d,n}$, $0 \leq n < N$, a set of M objective metrics, denotated as: $\mathbf{Q}_f^n = \{Q_0(V_f^{r,n}, V_f^{d,n}), ..., Q_{m-1}(V_f^{r,n}, V_f^{d,n})\}$. In particular, the following metrics are computed for each viewport pair:

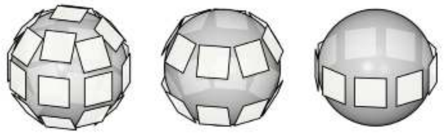

Fig. 3: Uniform (left), tropical (center), and equatorial (right) viewports sampling for computing viewport-based objective metrics.

Spatial Activity. (SA) of a pair of frames is defined as the root mean square (RMS) difference between the Sobel maps of each of the frames [22]. The Sobel operator, S, is defined as:

$$S(z) = \sqrt{(G_1 * z)^2 + (G_1^T * z)^2}, \tag{1}$$

where z is the frame picture and $*$ denotes the 2-dimensional convolution operation, G_1 is the vertical Sobel filter:

$$G_1 = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \tag{2}$$

and G_1^T is the transpose of G_1 (horizontal Sobel filter).

Let u and v be same frame from SRC and PVS, respectively. Then, we define the difference between the Sobel maps of both frames as: $s = S(u) - S(v)$ and compute SA as:

$$SA(v, u) = \sqrt{\frac{1}{MN} \sum_{i,j} |s_{ij}|^2}, \tag{3}$$

where i, j are the horizontal and vertical indices of s, and M and N are the height and width of the frames, respectively.

PSNRHVS and PSNRHVSM. PSNR-HVS and PSNR-HVS-M [23] are two models that have been designed to improve the performance of PSNR taking into consideration the HVS properties. PSNRHVS divides the image into 8x8 pixels non-overlapping blocks. Then the difference between the original and the distorted blocks is weighted for every 8x8 block by the coefficients of the Contrast Sensitivity Function (CSF). PSNR-HVS-M [23] is defined similarly, but the difference between the DCT coefficients is further multiplied by a contrast masking metric for every 8x8 block.

MS-SSIM. MultiScale-SSIM [24] is an extension of SSIM [25] for multiple scales. At every scale, from 1 to M, MS-SSIM iteratively applies a low pass filter to the reference and distorted images and downsample the filtered images by a factor of two. At the mth scale, contrast and structural comparisons are computed, respectively, $c_m(x,y)$ and $s_m(x,y)$. The luminance comparison is performed only at scale M and denoted as $l_M(x,y)$. The overall, MS-SSIM is then computed as:

$$MS{-}SSIM(\mathbf{x},\mathbf{y}) = [l_M(\mathbf{x},\mathbf{y})]^{\alpha_M} \cdot \prod_{m=1}^{M} [c_m(\mathbf{x},\mathbf{y})]^{\beta_m} \cdot [s_m(\mathbf{x},\mathbf{y})]^{\gamma_m} \tag{4}$$

where α_M, β_m, and γ_m adjust the relative importance of the different components.

GMSD. Gradient Magnitude Similarity Deviation (GMSD) [26] is based on the standard deviation of the gradient magnitude similarity map, GMS, which is computed as:

$$GMS(u,v) = \frac{2 \cdot m(u) \cdot m(v) + c}{m(u)^2 + m(v)^2 + c}, \tag{5}$$

where u and v are respectively the SRC and PVS frame; c is a positive constant that guarantees stability; and $m(z)$ is:

$$m(z) = \sqrt{(z * G_2)^2 + (z * G_2^T)^2}, \tag{6}$$

where $*$ denotes the convolution operator, G_2 represents the vertical Prewitt filter:

$$G_2 = \begin{bmatrix} \frac{1}{3} & 0 & -\frac{1}{3} \\ \frac{1}{3} & 0 & -\frac{1}{3} \\ \frac{1}{3} & 0 & -\frac{1}{3} \end{bmatrix}. \tag{7}$$

G_2^T is the transpose of G_2, i.e, the horizontal Prewitt filter. The GMSD index is then computed as:

$$GMSD(u,v) = \sqrt{\frac{1}{NM} \sum_{i,j} (GMS(u,v) - \overline{GMS(u,v)})^2} \tag{8}$$

where

$$\overline{GMS(u,v)} = \frac{1}{MN} \sum_{i,j} GMS(u,v) \tag{9}$$

Relative change in temporal information. Temporal information (TI) characterizes the amount of motion in a video and is defined as the standard deviation of the difference between two frames: $TI[F_n] = std(\Delta F_n)$, where $\Delta F_n = F_n - F_{n-1}$. Here, we define the relative change in TI as:

$$TI_{rel}[F_n] = \frac{|TI_{ref}[F_n] - TI_{dist}[F_n]|}{TI_{ref}[F_n]} \tag{10}$$

where $TI_{ref}[F_n]$ and $TI_{dist}[F_n]$ are respectively the TA for the frame F_n in the reference and distorted videos.

3.3. HVS-based temporal pooling

The per-viewports metrics $\mathbf{Q}_f^n$ for each frame, $f \in \{1, ..., F\}$ and viewport $n \in 1, ..., N$, are integrated to yield the overall quality of each viewport: $\mathbf{Q}_{pool}^n$. This integration is performed by the temporal pooling method [27] considering the characteristics of the HVS, in particular: (i) *the smooth effect*, i.e., the subjective ratings of the whole video sequence typically demonstrate far less variation than the frame-level quality scores; (ii) *the assymetric effect*, i.e., HVS is more sensitive to frame-level quality degradations than to improvement; and (iii) *recency effect*, i.e, subjects tend to put a higher weight on what they have seen most recently. More precisely, for each viewport n, we compute:

$$Q_{LP}^n(f) = \begin{cases} Q_{LP}^n(f-1) + \alpha \cdot \Delta Q(f), \text{if } \Delta Q^n \leq 0 \\ Q_{LP}^n(f-1) + \beta \cdot \Delta Q(f), \text{if } \Delta Q^n > 0 \end{cases} \tag{11}$$

$$Q_{pool}^n = \frac{1}{F} \sum_{f=1}^{F} (Q_{LP}^n(f) \cdot ln(\gamma \cdot f + 1)) \tag{12}$$

where $\Delta Q^n = Q_{frame}^n(f) - Q_{LP}^n(f-1)$ and $Q_{LP}^n = Q_{frame}^n(1)$, α and β controls the asymetric weights, and γ is a positive constant for adjusting the time-related weight. Similar to [27], in our experiments we use $\alpha = 0.03$, $\beta = 0.2$, and $\gamma = 1000$.

3.4. Random forest regression

After the temporal pooling, we end up with M features for each viewport, which are then concatenated as a feature vector, $\mathbf{Q} = [Q_0^0, ..., Q_{m-1}^0, Q_0^1, ..., Q_{m-1}^1, Q_0^{n-1}, ..., Q_{m-1}^{n-1}]$. Such vector is used for learning a non-linear mapping between the computed per-viewport features and the subjective DMOS scores of 360-degree videos. In our framework, we have tested both Support Vector Regression (SVR) [28] and Random forest regression (RFR) [29], from which we have chosen RFR because it significantly outperformed SVR in our preliminary tests. Next we detail the hyper-parameter tuning, training, and test processes of our experiments.

4. EXPERIMENTAL RESULTS AND ANALYSIS

We validate our proposal based on the VQA-ODV dataset [12], the largest available at this date. It is composed of 3 types of projections and 3 levels of H.265 distortions, quantization parameters (QP) = 27, 37, and 42. In total, there are 60 reference sequences (12 in raw format and others download from YouTube VR channel) and 180 distorted sequences that were rated by 221 participants. Both MOS (Mean Opinion Scores) and DMOS (Differential Mean Opinion Scores) are available. Without lack of generality, we focus only on the ERP sequences of the dataset in the following experiments.

We compare our method to PSNR, S-PSNR, WS-PSNR MS-SSIM, and VMAF, using common criteria for the evaluation of objective quality metrics: Pearson Linear Correlation Coefficient (PLCC) Spearman Rank Order Correlation Coefficient (SROCC), and Root Mean Squared Error (RMSE). SROCC measures the prediction monotonicity while PLCC and RMSE measure the prediction accuracy. Higher SROCC, PLCC and lower RMSE indicate good correlation with subjective scores. Moreover, we compare the performance of our method and the one of the other objective metrics when the features are computed in: i) the projection domain ("Proj."); ii) all viewports merged in a collage frame ("VP-Collage") (see Fig. 4); and iii) the viewports considered individually ("VP"), i.e., the metrics are computed independently for each viewports (as discussed in Section 3). Computing the objective metrics in the viewport collage frame is similar to averaging the quality of all the viewports. Based on the above 3 different modes, we performed two experiments: 1) using the same fixed train/test subset of the VQA-ODV dataset used in [12, 16]; and 2) a cross-validation approach on the whole VQA-ODV dataset. In both experiments, we use a uniform sampling with a 40-deg field of view for the viewports, which resolution matches the HMD resolution used in the dataset (an HTC Vive).

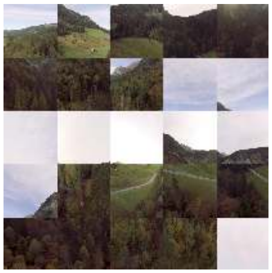

Fig. 4: Example of a collage frame used to compute the "VP-collage" mode of the objective metrics.

Fixed train/test sets. Table 1 shows the results of our method using the same (fixed) train/test data separation of VQA-ODV as in [12, 16]. Such a test/train sets are composed of a pre-defined subset of 80% of the sequences for training and 20% for testing. Given such a pre-defined subsets, we first run a group shuffle cross-validation only on the training set to find the best random forest hyper-parameters. Based on the found hyper-parameters, we then train the model with the training set and test it with the test set. For the individual objective metrics, the training phase is composed of fitting a 4-parameter logistic function with the train set and then computing its performance with the test set. As an example of the trained models for the different modes, Fig. 5 shows the average viewports features importance of our method ("VP").

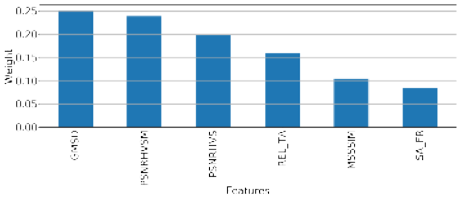

Fig. 5: Average features importance over viewports of our "VP" model.

Cross-validation. To avoid bias on the specific train/test set used above, we also performed a full cross-validation on the VQA-ODV dataset. In the cross-validation experiments, we performed a 1000x randomly group selection of 80%/20% train/test splitting of the dataset, and then computed the average LCC, SROCC, and RMSE of the models. To avoid bias, the group selection ensures that there is no overlap between content in the training and test sets. Table 2 shows the average PLCC, SROCC, and RMSE results for the cross-validation experiments, and Fig. 6 depicts the distribution of the correlation scores through a violin plot.

Discussion. In both fixed train/test set and group cross-validation results, the best correlation is achieved by our method using features computed separated for each viewport ("VP"). Besides having a better average, Fig. 6 also highlights the density of the group cross-validation dataset, showing a better performance for such a method. The above results can be explained by both viewports being closer to what the users see and that the model can learn the most important viewports, which is not the case when using the "VP-Collage" mode. It is also interesting to note that our method (on both "Proj." and "VP-Collage") also outperforms VMAF, which can be explained by the choice of objective metrics, the temporal pooling, and regression methods we are using. Finally, our results also show that the use of viewports (even when using the "VP-collage mode) improve the results when compared to the projection domain.

Table 1: Fixed train/test set results.

Metric	PLCC	SROCC	RMSE
PSNR	0.72495	0.73797	8.17600
PSNR (VP-Collage)	0.76222	0.76345	7.58240
S-PSNR	0.75138	0.77040	7.75570
WS-PSNR	0.74328	0.56056	7.95010
MS-SSIM (Proj.)	0.76005	0.78867	7.87410
MS-SSIM (VP-Collage)	0.81719	0.84144	7.00240
VMAF (Proj.)	0.79657	0.79382	7.24810
VMAF (VP-Collage)	0.84483	0.85637	6.27100
Ours (Proj.)	**0.85629**	**0.86873**	**6.35880**
Ours (VP-Collage)	**0.89867**	**0.87439**	**5.72560**
Ours (VP)	**0.92575**	**0.91712**	**4.99540**

Table 2: Average of GroupShuffle cross validation (80%/20%) performance on VQA-ODV.

Metric	PLCC	SROCC	RMSE
PSNR (Proj.)	0.57156	0.61873	9.8249
PSNR (VP-Collage)	0.64746	0.68579	9.1224
S-PSNR	0.62460	0.66731	9.3461
WS-PSNR	0.59803	0.64501	9.5983
MS-SSIM (Proj.)	0.75004	0.77535	7.9351
MS-SSIM (VP-Collage)	0.76405	0.79113	7.758
VMAF	0.74692	0.76673	7.9631
VMAF (VP-Collage)	0.78085	0.79802	7.5147
Ours (Proj.)	**0.81728**	**0.82901**	**6.8716**
Ours (VP-Collage)	**0.82676**	**0.82647**	**6.7376**
Ours (VP)	**0.86778**	**0.86769**	**5.9367**

5. CONCLUSION

We propose the use of viewport-based multi-metrics fusion for 360-degree VQA. Computation of features in viewports implies that our metric can be applied on a variety of projec-

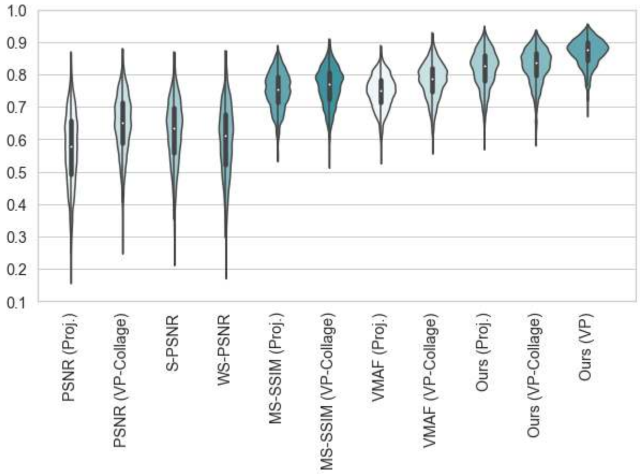

Fig. 6: Violin plots for the GroupShuffle (80%x20%) cross-validation PLCC performance on VQA-ODV dataset.

tions, and our experiments demonstrate that the multi-metrics fusion is capable of achieving state-of-the-art results while requiring much less training data than deep learning techniques. As future work, we plan to consider color and visual attention and test our method on different datasets.

6. REFERENCES

[1] R. Azevedo et al., “Visual distortions in 360-degree videos,” *IEEE Trans. on Circuits and Syst. for Video Technol.*, 2019.

[2] “State-of-the-art in 360deg Video/Image Processing: Perception, Assessment and Compression,” Apr. 2019, arXiv: 1905.00161.

[3] Z. Chen, Y. Li, and Y. Zhang, “Recent advances in omnidirectional video coding for virtual reality: Projection and evaluation,” *Signal Process.*, May 2018.

[4] M. Yu, H. Lakshman, and B. Girod, “A Framework to Evaluate Omnidirectional Video Coding Schemes,” in *2015 IEEE ISMAR*, Sept. 2015.

[5] Y. Sun, A. Lu, and L. Yu, “Weighted-to-Spherically-Uniform Quality Evaluation for Omnidirectional Video,” *IEEE Signal Process. Lett.*, 2017.

[6] N. Birkbeck, C. Brown, and R. Suderman, “Quantitative evaluation of omnidirectional video quality,” in *9th QoMEX*, May 2017.

[7] R. Azevedo et al., “Subjective and viewport-based objective quality assessment of equiangular cubemap 360 videos,” in *Electronic Imaging*, 2020.

[8] T. Liu, W. Lin, and C. . J. Kuo, “Image quality assessment using multi-method fusion,” *IEEE Trans. on Image Process.*, May 2013.

[9] R. Rassool, “VMAF reproducibility: Validating a perceptual practical video quality metric,” in *IEEE BMSB*, 2017.

[10] M. Rousselot, X. Ducloux, O. L. Meur, and R. Cozot, “Quality metric aggregation for HDR/WCG images,” in *2019 IEEE ICIP*, Sep. 2019.

[11] T.-J. Liu et al., “A multi-metric fusion approach to visual quality assessment,” in *3rd QoMEX*, Mechelen, Belgium, Sept. 2011.

[12] C. Li, M. Xu, X. Du, and Z. Wang, “Bridge the Gap Between VQA and Human Behavior on Omnidirectional Video: A Large-Scale Dataset and a Deep Learning Model,” in *ACM MM*, Seoul, Republic of Korea, 2018.

[13] V. Zakharchenko, K. P. Choi, and J. H. Park, “Quality metric for spherical panoramic video,” Sept. 2016.

[14] S. Croci et al., “Voronoi-based objective quality metrics for omnidirectional video,” .

[15] H. G. Kim, H.-t. Lim, and Y. M. Ro, “Deep Virtual Reality Image Quality Assessment with Human Perception Guider for Omnidirectional Image,” *IEEE Trans. on Circuits and Syst. for Video Technol.*, 2019.

[16] C. Li et al., “Viewport proposal CNN for 360deg video quality assessment,” , June 2019.

[17] M. Pinson and S. Wolf, “A new standardized method for objectively measuring video quality,” *IEEE Trans. on Broadcast.*, Sept. 2004.

[18] K. Seshadrinathan and A. C. Bovik, “Motion-tuned spatio-temporal quality assessment of natural videos,” *IEEE Trans. on Image Process.*, Feb. 2010.

[19] P. V. Vu and D. M. Chandler, “ViS3: an algorithm for video quality assessment via analysis of spatial and spatiotemporal slices,” *J. of Electronic Imaging*, Feb. 2014.

[20] F. Pearson, *Map Projections: Theory and Applications*, 1990.

[21] F. De Simone et al., “Geometry-driven quantization for omnidirectional image coding,” in *2016 PCS*, Dec 2016.

[22] P. G. Freitas et al., “Using multiple spatio-temporal features to estimate video quality,” *Signal Process. Image Commun.*, May 2018.

[23] N. Ponomarenko et al., “On between-coefficient contrast masking of DCT basis functions,” in *3rd Intern. Workshop on Video Process. and Quality Metrics*, 2007.

[24] Z. Wang et al., “Multiscale structural similarity for image quality assessment,” in *The 37th Asilomar Conf. on Signals, Systems & Computers, 2003*, 2003.

[25] Z. Wang et al., “Image Quality Assessment: From Error Visibility to Structural Similarity,” *IEEE Trans. on Image Process.*, Apr. 2004.

[26] W. Xue et al., “Gradient magnitude similarity deviation: A highly efficient perceptual image quality index,” *IEEE Trans. on Image Process.*, Feb 2014.

[27] Y. Lu, M. Yu, and G. Jiang, “Low-Complexity Video Quality Assessment Based on Spatio-Temporal Structure,” in *Information and Software Technologies*. 2019.

[28] D. Basak et al., “Support vector regression,” *Neural Information Processing-Letters and Reviews*, 2007.

[29] L. Breiman, “Random forests,” *Machine learning*, 2001.