Adaptive Hypergraph Convolutional Network for No-Reference 360-degree Image Quality Assessment

Jun Fu
University of Science and Technology of China
Hefei, Anhui, China
fujun@mail.ustc.edu.cn

Chen Hou
University of Science and Technology of China
Hefei, Anhui, China
houchen@mail.ustc.edu.cn

Wei Zhou*
University of Waterloo
Waterloo, Canada
weichou@mail.ustc.edu.cn

Jiahua Xu
University of Science and Technology of China
Hefei, Anhui, China
xujiahua@mail.ustc.edu.cn

Zhibo Chen*
University of Science and Technology of China
Hefei, Anhui, China
chenzhibo@ustc.edu.cn

ABSTRACT

In no-reference 360-degree image quality assessment (NR 360IQA), graph convolutional networks (GCNs), which model interactions between viewports through graphs, have achieved impressive performance. However, prevailing GCN-based NR 360IQA methods suffer from three main limitations. First, they only use high-level features of the distorted image to regress the quality score, while the human visual system scores the image based on hierarchical features. Second, they simplify complex high-order interactions between viewports in a pairwise fashion through graphs. Third, in the graph construction, they only consider the spatial location of the viewport, ignoring its content characteristics. Accordingly, to address these issues, we propose an adaptive hypergraph convolutional network for NR 360IQA, denoted as AHGCN. Specifically, we first design a multi-level viewport descriptor for extracting hierarchical representations from viewports. Then, we model interactions between viewports through hypergraphs, where each hyperedge connects two or more viewports. In the hypergraph construction, we build a location-based hyperedge and a content-based hyperedge for each viewport. Experimental results on two public 360IQA databases demonstrate that our proposed approach has a clear advantage over state-of-the-art full-reference and no-reference IQA models.

CCS CONCEPTS

• **Computing methodologies** → *Computer vision.*

KEYWORDS

360-degree, image quality assessment, adaptive hypergraph

*Corresponding author

MM '22, October 10–14, 2022, Lisboa, Portugal

ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3548337

ACM Reference Format:
Jun Fu, Chen Hou, Wei Zhou, Jiahua Xu, and Zhibo Chen. 2022. Adaptive Hypergraph Convolutional Network for No-Reference 360-degree Image Quality Assessment. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3503161.3548337

1 INTRODUCTION

360-degree images/videos have become increasingly popular with the boost of virtual reality (VR) technology [7]. Compared to 2D images/videos, 360-degree ones allow users to interactively manipulate the perspective through head-mounted displays (HMDs), thereby bringing users an immersive experience. In real-world VR broadcasting systems, 360-degree images/videos typically undergo three stages, i.e., acquisition, compression, and transmission, before reaching end users. Among these stages, 360-degree images/videos may be degraded by various distortions such as white noise, blurring, and compression artifacts [9, 25]. The quality degradation of 360-degree content may significantly impair the user's quality of experience [40]. Therefore, it is important to study 360-degree image quality assessment (360IQA), which can guide the optimization of VR broadcasting systems.

Compared to 2D image quality assessment (2DIQA), 360IQA encounters more challenges. First, 360IQA needs to consider geometric deformation and pixel redundancy, as they are often introduced when storing 360-degree content. Second, 360IQA needs to consider the gap between the viewport image seen in the HMD and its corresponding 360-degree image in the equirectangular format. Third, 360IQA needs to consider interactions between viewports. Specifically, since users only watch a small portion of the 360-degree image at a time, they need to browse multiple viewports for accurate quality assessment. During this viewing process, the visual information of different viewports is interacted [32].

In general, 360IQA can be divided into two categories: full-reference (FR) 360IQA and no-reference (NR) 360IQA. FR 360IQA assesses the distorted 360-degree image by comparing it with the original 360-degree image. Due to the geometric deformation and pixel redundancy, it is inappropriate for FR 360IQA to directly use FR 2DIQA metrics such as PSNR and SSIM [30]. Accordingly,

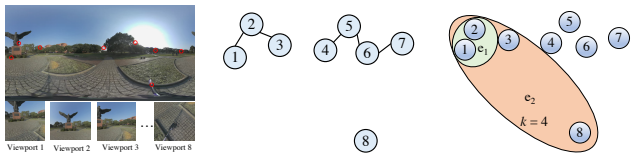


Figure 1: Comparison of graphs and hypergraphs in modeling interactions between viewports. (a) The distorted 360-degree image and 8 key viewports. (b) The constructed graph based on spatial locations of viewports. (c) Two hyperedges constructed for the node 1. $\mathbf{e}_1$ and $\mathbf{e}_2$ contain spatial and semantic neighborhoods of the node 1, respectively. k denotes the number of semantic neighborhoods.

researchers design various variants of FR 2DIQA metrics, e.g., WS-PSNR [28] and WS-SSIM [43]. NR 360IQA first trains a 360IQA model using distorted 360-degree images with labeled quality scores, and then uses it to assess the unseen distorted 360-degree image without the original 360-degree image. Since the original 360-degree image is typically unavailable in many real-world applications, we focus on NR 360IQA instead of FR 360IQA. Early NR 360IQA methods use patches cropped from the distorted 360-degree image to predict the quality score. As patches are not the actual content viewed by users, viewport-oriented methods are thus developed. Recently, the viewport-based graph convolutional neural network (VGCN [32]) considers interactions between viewports and models them through graphs, achieving remarkable performance. As shown in Figure 1 (b), VGCN defines the graph structure based on spatial locations of viewports, and uses high-level features of viewports as node features.

However, there are three main limitations in existing GCN-based NR 360IQA methods. First, only high-level features of viewports are used for quality evaluation, which is inconsistent with the human visual perception process. Specifically, it is known that the human brain hierarchically processes the perceived image and the human visual system comprehends the image based on the obtained hierarchical features [29]. As intrinsically related to image understanding, quality evaluation also relies on hierarchical features. Second, since each edge only connects two viewports, graphs have limited capabilities for modeling complicated interactions between three or more viewports. Third, the graph structure only represents spatial relations between viewports. However, it is also important for quality evaluation to consider semantic correlations between viewports. For example, as shown in Figure 1 (b), although far away from the node 1, the node 8 can offer guidance to assess the quality of the road in the node 1.

To address these issues, in this paper, we propose an adaptive hypergraph convolutional network for NR 360IQA, denoted as AHGCN. Specifically, we first develop a multi-level viewport descriptor, which combines low-level, mid-level, and high-level features of viewports to produce hierarchical representations. Then, we model interactions between viewports through hypergraphs instead of graphs. For each viewport, we construct a location-based hyperedge based on the angular distance between viewports, and a content-based hyperedge according to the content similarity between viewports. Experimental results on two public 360IQA databases demonstrate that the proposed AHGCN has a clear advantage over state-of-the-art 360IQA models. The main contributions of the proposed method are listed as follows:

- We design a multi-level viewport descriptor and verify the effectiveness of hierarchical representations for 360IQA.
- We present the first attempt to use hypergraphs to capture interactions between viewports and validate the superiority of hypergraphs over graphs.
- We propose an adaptive hyperedge construction method, which considers both the locations and content features of viewports.

The rest of this paper is organized as follows. Section 2 introduces works related to our approach. We detail the proposed AHGCN for NR 360IQA in Section 3, followed by experimental results presented in Section 4. Section 5 concludes the paper.

2 RELATED WORK

In this section, we first overview FR 360IQA metrics, then introduce NR 360IQA models, and finally present the progress of the hypergraph learning.

2.1 FR 360IQA

FR 360IQA aims to evaluate the quality of the distorted 360-degree image with reference to the original 360-degree image. Compared to FR 2DIQA, FR 360IQA needs to consider geometric deformation and pixel redundancy issues brought in storing 360-degree content. Therefore, diverse variants of FR 2DIQA models are proposed. Yu et al. [34] develop a spherical PSNR (S-PSNR), which calculates PSNR for the set of points uniformly distributed on a sphere instead of the rectangular plane. Sun et al. [28] design a weighted spherical PSNR (WS-PSNR), which weights the error of points sampled on the 2D plane according to their stretch degree. Zakharchenko et al. [35] propose the Craster Parabolic Projection PSNR (CPP-PSNR), which calculates PSNR on the Craster Parabolic Projection domain. Xu et al. [33] put forward a non-content-based PSNR (NCP-PSNR), which weights different regions based on the distribution of viewing directions. Like PSNR, SSIM [30] also derives various versions for FR 360IQA. Chen et al. [2] propose a spherical SSIM (S-SSIM), which computes the similarity between distorted and original 360-degree images on the sphere. Zhou et al. [43] develop weighted-to-spherically-uniform SSIM (WS-SSIM). Facebook [5] designs SSIM360, which overcomes the warping issue of 360-degree images by a weighting mechanism.

2.2 NR 360IQA

NR 360IQA aims to assess the distorted 360-degree image without referring to the original 360-degree image. Early NR 360IQA methods usually leverage patch-level features to perform quality prediction. Kim et al. [12, 19] propose a deep-learning-based VR image quality assessment framework (DeepVR-IQA) with adversarial learning. It firstly predicts quality scores of patches sampled from the distorted 360-degree image and then fuses them to obtain the final quality score using a position-aware weighting mechanism. Similarly, Li et al. [17] predict quality scores of patches and weight them based on the estimated head movement (HM) and eye movement (EM) maps.

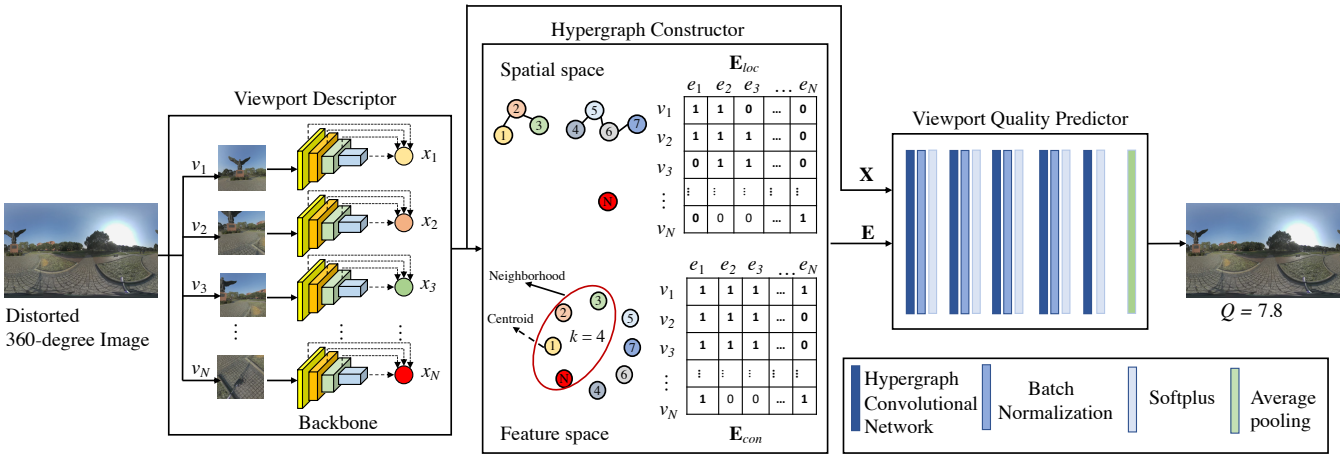


Figure 2: Illustration of the proposed adaptive hypergraph convolutional network. E and X denote the hypergraph structure and hierarchical features of viewports. E consists two parts: location-based hyperedges $\mathbf{E}_{loc}$ and content-based hyperedges $\mathbf{E}_{con}$. Q is the predicted quality score. k represents the number of semantic neighborhoods.

Considering the inconsistency between the viewport image and the patch, viewport-based NR 360IQA methods are thus proposed. Li et al. [18] design a multi-task framework, which simultaneously predicts quality scores of viewports and performs HM and EM map prediction. Zhou et al. [42] present a NR 360IQA framework, which predicts the quality score of the distorted 360-degree image based on multi-frequency information and local-global naturalness. Sun et al. [27] put forward a multi-channel convolution neural network for blind 360-degree image quality assessment (MC360IQA). It uses a shared ResNet-34 network to parallelly extract high-level features from six viewports, and concatenate them for quality score regression. Recently, considering the importance of interactions between viewports, Xu et al. [32] develop a viewport-oriented graph convolutional network (VGCN). Despite achieving remarkable performance, GCN-based NR 360IQA methods still suffer from several limitations as aforementioned.

2.3 Hypergraph Learning

Unlike conventional graphs, where each edge only connects two nodes, hypergraphs allow edges to connect three or more nodes. As such, hypergraphs are widely used to model complicated systems with high-order interactions. Zhou et al. [41] firstly introduce hypergraphs into clustering, embedding, and classification tasks, and achieve superior performance than graph-based methods. Feng et al. [6] propose a hypergraph neural network (HGNN), which extends hypergraph-based representation learning to learn-based models. Jiang et al. [11] develop a dynamic hypergraph neural network (DHGNN), which uses k-NN and k-means clustering methods to construct hyperedges without a pre-defined hypergraph structure. Zhang et al. [38] put forward a self-attention-based graph neural network for hypergraphs (Hyper-SAGNN), which learns an aggregation function for each hyperedge.

Most of the existing studies on hypergraphs focus on classification tasks but pay less attention to other tasks. In this paper, we are the first to introduce hypergraphs to NR 360IQA for modeling interactions between viewports. Moreover, we develop an adaptive hyperedge construction method, which considers not only spatial locations of viewports but also their content characteristics.

3 PROPOSED METHOD

In this section, we first define the problem of viewport-based NR 360IQA. Then, we introduce the proposed AHGCN in detail. Finally, we describe the implementation and training details.

3.1 Problem Formulation

Given N viewports $\mathbf{V} = \{v_1, ..., v_N\}$ sampled from a distorted 360-degree image I, we aim to predict the quality score of I using a mapping function F:

$$Q = F(\mathbf{V}; \theta), \tag{1}$$

where θ represents all learnable parameters of the model F.

In this paper, we propose an adaptive hypergraph convolutional network (AHGCN) to model F. As shown in Figure 2, AHGCN consists of three parts: a viewport descriptor, a hypergraph constructor, and a viewport quality predictor. Next, we will detail these parts in sequence.

3.2 Viewport Descriptor

The viewport descriptor aims to extract hierarchical features from the input viewport. Recently, deep convolutional neural networks have exhibited impressive power in representing perceptual image distortions [1, 22, 37]. Inspired by this, we bulid the viewport descriptor on backbones, pre-trained for object recognition [16] on the ImageNet database [3]. As shown in Figure 3, the viewport descriptor first feeds the input viewport v_i into a backbone network f:

$$\mathbf{L}_i = \{l_{i,1}, ..., l_{i,m}\} = f(v_i; \theta_f), \forall i \in [1, N], \tag{2}$$

where $\mathbf{L}_i$ denotes multi-level features of the i-th viewport, $l_{i,m}$ represents the feature map at the m-th level, and θ_f represents pre-trained weights of the backbone. Then, it individually compacts

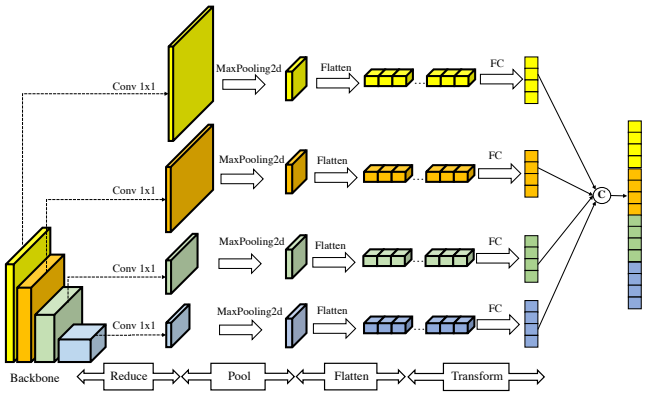


Figure 3: The framework of the proposed viewport descriptor. Conv 1x1 means the convolutional layer with the kernel size of 1, and FC denotes the fully connected network.

each feature map into a vector:

$$c_{i,j} = g(l_{i,j}), \forall j \in [1, m], \tag{3}$$

where the function $g(\cdot)$ denotes the principle of "*reduce-pool-flatten-transform*", $c_{i,j} \in \mathbb{R}^d$ is the compacted result of the feature map at j-th level. Finally, it concatenates compacted multi-level features to obtain hierarchical representations of the i-th viewport $x_i \in \mathbb{R}^{md}$:

$$x_i = c_{i,1} \cup c_{i,2} \cup ... \cup c_{i,m}. \tag{4}$$

3.3 Hypergraph Constructor

To assess the distorted 360-degree image, subjects will browse the landscape of the sphere for a while. In this viewing process, the visual information of different viewports is interacted and aggregated for local quality evaluation [32]. In this paper, we model interactions between viewports using hypergraphs, which can capture high-order interaction by connecting multiple viewports through one hyperedge.

The hypergraph constructor aims to discover a location-based hyperedge and a content-based hyperedge for each viewport. The motivation of this design is two-fold. On the one hand, when evaluating one viewport, the user tends to refer to viewports close to it rather than those far away from it. This is mainly because dramatic head movements rarely occur in the user's viewing process. On the other hand, the visual information of viewports with same or similar content to the evaluated viewport may be also useful. For example, as shown in Figure 2, the visual information of the road in N-th viewport could offer guidance to assess the quality of the first viewport.

Location-based Hyperedges. Motivated by the success of adding an edge between two viewports whose angular distance is lower than a threshold [32], we construct location-based hyperedges as follows. The location-based hypergraph $\mathbf{E}_{loc} = \{e^1_{loc}, ..., e^N_{loc}\} \in \mathbb{R}^{N\times N}$ consists of N hyperedges, and the hyperedge $e^i_{loc} \in \mathbb{R}^{1\times N}$ collects spatial neighborhoods of the i-th viewport :

$$e^{i,p}_{loc} = \begin{cases} 1, \text{if } AngularDist[(\phi_i, \Theta_i), (\phi_p, \Theta_p)] \le \delta \\ 0, \text{otherwise} \end{cases}, \tag{5}$$

where ϕ_i and Θ_i denote the longitude and latitude of the i-th viewport, respectively. The function $AngularDist(\cdot)$ is defined as $arccos(cos\phi_i cos\phi_p + sin\phi_i sin\phi_p + cos\Theta_i cos\Theta_p)$, which computes the angular distance between the i-th viewport and the p-th viewport. δ is a pre-defined angular distance threshold.

Content-based Hyperedges. Motivated by the success of using k-nearest neighborhood to construct a hyperedge based on cosine similarity [11], we construct content-based hyperedges as follows. The content-based hypergraph $\mathbf{E}_{con} = \{e^1_{con}, ..., e^N_{con}\} \in \mathbb{R}^{N\times N}$ also consists of N hyperedges, and the hyperedge $e^i_{con} \in \mathbb{R}^{1\times N}$ collects semantic neighborhoods of the i-th viewport :

$$e^{i,p}_{con} = \begin{cases} 1, \text{if } v_p \in \mathcal{N}(v_i) \\ 0, \text{otherwise} \end{cases}, \tag{6}$$

where $\mathcal{N}(v_i)$ represents the neighborhood of the i-th viewport in the feature space, and the size of the neighborhood is k. For simiplicity, we measure the distance between viewport i and p using the feature similarity:

$$s_{i,p} = \frac{x_i \cdot x_p}{\max\{||x_i||_2 \cdot ||x_p||_2, \epsilon\}}, \tag{7}$$

where ϵ is set as $1e{-}12$ to avoid division by zero.

3.4 Viewport Quality Predictor

Let us denote hierarchical representations of N viewports as $\mathbf{X} = \{x_1, x_2, ..., x_N\} \in \mathbb{R}^{N\times md}$ and the hypergraph structure as $\mathbf{E} = \{\mathbf{E}_{loc}, \mathbf{E}_{con}\} \in \mathbb{R}^{N\times 2N}$. The viewport quality predictor aims to derive the quality score of the distorted 360-degree image through hypergraph convolutional neural networks (HGCNs). The single-layer HGCN [6] is calculated as follows:

$$\mathbf{H}^{(t+1)} = \sigma[\mathrm{BN}_{\gamma,\beta}(\hat{\mathbf{E}}\mathbf{H}^{(t)}\mathbf{W}^{(t)})], \tag{8}$$

where $\mathrm{BN}_{\gamma,\beta}(\cdot)$ is the batch normalization with trainable parameters of γ and β, σ is the Softplus activation function. $\mathbf{H}^{(t)}$ and $\mathbf{H}^{(t+1)}$ are the input and output of the t-th HGCN, respectively. The input of the first HGCN is $\mathbf{X}$, i.e. $\mathbf{H}^{(0)} = \mathbf{X}$. $\mathbf{W}^{(t)}$ is the set of learnable parameters of the t-th HGCN. $\hat{\mathbf{E}} \in \mathbb{R}^{N\times N}$ is the normalized hypergraph structure, calculated as follows:

$$\hat{\mathbf{E}} = \mathbf{D}_v^{-1/2}\mathbf{E}\mathbf{D}_e^{-1}\mathbf{E}^T\mathbf{D}_v^{-1/2}, \tag{9}$$

where $\mathbf{D}_v = diag\{\sum_j \mathbf{E}_{1,j}, ..., \sum_j \mathbf{E}_{N,j}\} \in \mathbb{R}^{N\times N}$ is the node degree matrix and $\mathbf{D}_e = diag\{\sum_j \mathbf{E}^T_{1,j}, ..., \sum_j \mathbf{E}^T_{2N,j}\} \in \mathbb{R}^{2N\times 2N}$ is the edge degree matrix. The output of the last HGCN is $\mathbf{H}^{(n)} \in \mathbb{R}^{N\times 1}$, which contains the quality scores of all viewports. The quality score of the entire 360-degree image is obtained by an average pooling layer:

$$Q = \mathrm{Mean}(\mathbf{H}^n). \tag{10}$$

Table 1: Details of the ResNet-18 architecture [10].

Layer name	Output size	Layer
conv1	$112 \times 112 \times 64$	7×7, 64, stride 2
conv2_x	$56 \times 56 \times 64$	3×3 max-pool, stride 2 $\begin{bmatrix} 3 \times 3, & 64, & 1 \\ 3 \times 3, & 64, & 1 \end{bmatrix} \times 2$
conv3_x	$28 \times 28 \times 128$	$\begin{bmatrix} 3 \times 3, & 128, & 1 \\ 3 \times 3, & 128, & 1 \end{bmatrix} \times 2$
conv4_x	$14 \times 14 \times 256$	$\begin{bmatrix} 3 \times 3, & 256, & 1 \\ 3 \times 3, & 256, & 1 \end{bmatrix} \times 2$
conv5_x	$7 \times 7 \times 512$	$\begin{bmatrix} 3 \times 3, & 512, & 1 \\ 3 \times 3, & 512, & 1 \end{bmatrix} \times 2$
average pool	$1 \times 1 \times 512$	7×7 average pool
fully connected	1000	512×1000 fully connections
softmax	1000	

[1] Convolutional layer: kernel size, channel, stride.
[2] Output size: height, width, channel.
[3] x: the number of the layer.

3.5 Training Setup

Implementation Details. For the viewport descriptor, following the literature [32], we choose ResNet-18 architecture [10] as the backbone network, and select features from conv2_5, conv3_9, conv4_13, and conv5_17 layers. The details of the ResNet-18 network are presented in Table 1. In the principle of "Reduce-Pool-Flatten-Transform", each feature map is first converted to a tensor with the size of $8 \times 8 \times 16$ using convolutional neural networks with kernel size of 1 and the operation of MaxPooling2d, which is later tranformed into a 256-dim vector by a fully connected network (FC). In constructing location-based hyperedges, we set the angular distance threshold as half of the viewport size, i.e., 45°. In the construction of content-based hyperedges, through parameter tuning experiments, we adopt the nearest neighbor with $k = 15$ on the OIQA dataset and the farthest neighbor with $k = 5$ on the CVIQD dataset. For the viewport quality predictor, following the literature [32], five HGCN layers are adopted and the viewport feature dimensions after each HGCN layer are [256, 128, 64, 32, 1].
Loss Function. We use mean square error (MSE) as the training objective:

$$L = \frac{1}{B} \sum_{b=1}^{B} (Q_b - G_b)^2, \tag{11}$$

where B is the batch size. Q_b and G_b are the estimated and ground-truth MOS values of the b-th distorted 360-degree image, respectively.
Training Settings. We initialize the ResNet-18 network with pre-trained weights provided by the literature [32]. All trainable parameters are optimized by the Adam optimizer [14]. The learning rate of the ResNet-18 network is initialized to $1e{-}6$ and fixed during the training phase, the one for remaining trainable parameters is initialized to $1e{-}3$, scaled by 0.25 every 40 epochs. The maximum of training epoch is set to 80. During training, the mini-batch size is set to 16 and the input viewport is resized to 256×256. In addition, to avoid overfitting, we use dropout [24] in the first 4 HGCN layers, and the dropout rate is set to 0.5.

4 EXPERIMENTS

In this section, we first introduce databases, performance metrics, and baselines. Then, we compare AHGCN with existing competitive IQA metrics on two public datasets. Finally, we conduct an ablation study to verify the effectiveness of each component in AHGCN.

4.1 Databases

We evaluate the effectiveness of the proposed method on two public datasets, i.e., OIQA database [4] and CVIQD database [26]. Following the literature [32], we split the database into the training and testing set, and crop 20 viewports for each distorted 360-degree image.

OIQA Database. It contains 320 distorted 360-degree images, which are obtained by applying 4 distortion types with 5 levels to 16 raw 360-degree images. The distortion types are JPEG compression (JPEG), JPEG2000 compression (JP2K), Gaussian blur (BLUR), and Gaussian white noise (WN). Mean opinion score (MOS) values of distorted 360-degree images range from 1 to 10.

CVIQD Database. It collects 528 compressed 360-degree images generated from 16 lossless source images. It considers three compression distortion types, i.e., JPEG, H.264/AVC, and H.265/HEVC. The MOS values of distorted 360-degree images are normalized and rescaled to the range [0, 100].

4.2 Evaluation Metrics and Baselines

Standard measures [8] and the Krasula methodology [15] are used to measure the performance of IQA metrics.

Standard measures. We choose three standard measures, i.e., Spearmans rank order correlation coefficient (SROCC), Pearsons linear correlation coefficient (PLCC), and root mean squared error (RMSE). PLCC and RMSE reflect the prediction accuracy, while SROCC reflects the prediction monotonicity. A better 360IQA model should have lower RMSE values while higher SROCC and PLCC values. Before calculating the PLCC and RMSE, a five-parameter logistic function [23] is applied on predictions of IQA metrics:

$$\hat{q} = \beta_1 \left[\frac{1}{2} - \frac{1}{1 + \exp(\beta_2 (q - \beta_3))} \right] + \beta_4 q + \beta_5, \tag{12}$$

where $\hat{q}$ is the mapping result of the predicted score q. $\beta_1, \beta_2, \beta_3, \beta_4$ and β_5 are the parameters to be fitted.

Krasula methodology [15]. The Krasula methodology is used to evaluate the reliability of IQA metrics from three aspects, i.e., the area under the ROC curve of Different vs. Similar (AUC-DS), the area under the ROC curve of Better vs. Worse categories (AUC-BW), and the percentage of correct classification (C_0). Specifically, AUC-BW and AUC-DS represent the capacity of IQA metrics for distinguishing better/worse and different/similar pairs. A better IQA metric should have higher AUC-DS, AUC-BW, and C_0.

Baselines. We select thirteen representative IQA methods for performance comparison. The competitive approaches include five

FR 2DIQA metrics, i.e., PSNR, SSIM [30], MS-SSIM [31], FSIM [36], and DeepQA [13]; three learning-based NR 2DIQA metrics, i.e., BRISQUE [21], BMPRI [20], and DB-CNN [39]; three FR 360IQA metrics, i.e., S-PSNR [34], WS-PSNR [28], and CPP-PSNR [35]; two viewport-oriented NR 360IQA metrics, i.e., MC360IQA [27] and VGCN [32]. Compared to MC360IQA, VGCN considers interactions between viewports. It is worth noting that this paper only uses the local branch of the original VGCN for a fair comparison.

4.3 Performance Comparison

Table 2 and Table 3 present the results of performance comparison. According to these tables, we can draw the following conclusions:

- PSNR and its variants are inferior to SSIM-based IQA metrics. This is because PSNR can only reflect the pixel error between the distorted panorama and the undistorted one, while SSIM can reflect the structural distortion related to the human visual system.
- Viewport-oriented NR 360IQA metrics have a clear advantage over FR and NR 2DIQA metrics. This confirms the gap between 2DIQA and 360IQA, and points out the importance of viewport-level information to 360IQA.
- VGCN significantly outperforms MC360IQA in terms of PLLC, SROCC, and RMSE. This shows that it is essential to consider interactions between viewports in 360IQA.
- Compared to VGCN (local), the proposed AHGCN achieves better performance on both OIQA and CVIQD datasets. This verifies the effectivenss of AHGCN.

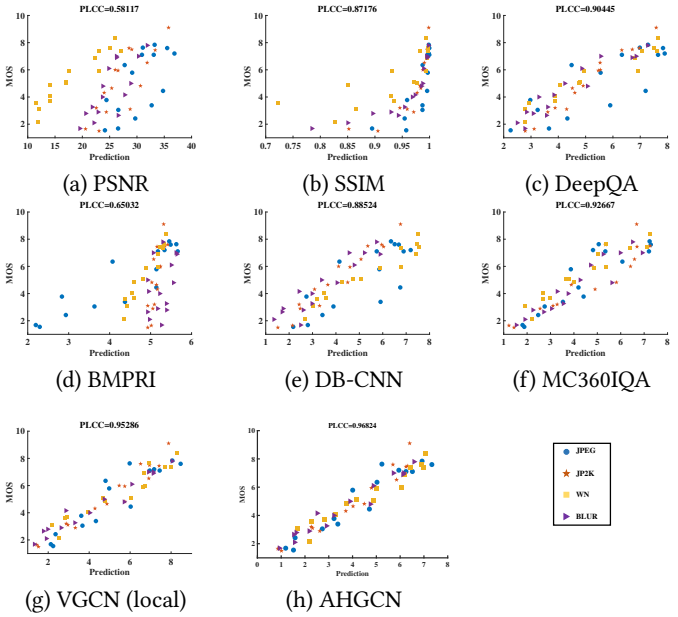
From the performance of each method in individual distortion type, we have the following observations:

- Compared with existing competitive IQA metrics, our proposed AHGCN achieves comparable or superior performance across a broad of distortion types.
- AHGCN exhibits outstanding power in evaluating compressed 360-degree images, especially for ones encoded by AVC.
- The development of image compression technology poses more challenges to IQA. As shown in Table 3, the performance of all IQA metrics significantly drops from JPEG to HEVC. One possible reason is that artifacts introduced by new encoders are more indistinguishable compared with blockiness and tonal distortion brought by JPEG.

Figure 4 and Figure 5 show the scatter plots of ground-truth MOS values versus predictions of IQA models for individual distortion types. We have the following findings:

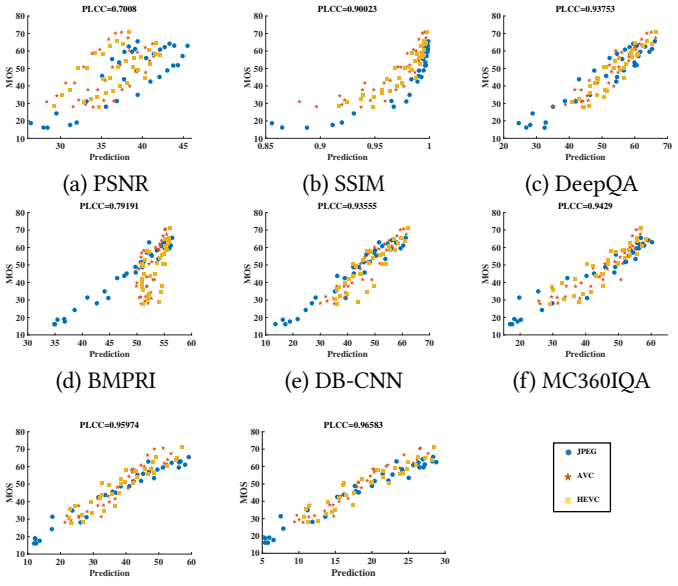
- As shown in Figure 4, compared with the other three distortions, predictions of JPEG compression have a lower linear correlation with subjective scores. This reveals that the quality assessment of JPEG compression distortion is more challenging.
- According to Figure 5, although the difficulty of 360IQA increases with the development of coding technology, our predictions have comparable or higher linear correlation with subjective scores than existing NR IQA metrics.

Figure 6 presents the results of Krasula criteria on the CVIQD database. In the significance plot, black (white) boxes mean that the metric in the row is significantly worse (better) than the metric in the column, and gray boxes indicate that two metrics are evenly matched. As shown in Figure 6, our approach achieves the best performance in terms of AUC-DS, AUC-BW, and C_0. This reveals that our approach is more reliable than existing IQA metrics.



(a) PSNR (b) SSIM (c) DeepQA
(d) BMPRI (e) DB-CNN (f) MC360IQA
(g) VGCN (local) (h) AHGCN

Figure 4: Scatter plots of MOS values versus predictions of IQA metrics on the testing set of OIQA database.



(a) PSNR (b) SSIM (c) DeepQA
(d) BMPRI (e) DB-CNN (f) MC360IQA
(g) VGCN (local) (h) AHGCN

Figure 5: Scatter plots of MOS values versus predictions of IQA metrics on the testing set of CVIQD database.

4.4 Cross-dataset Validation

To verify the generalization ability of the proposed AHGCN, we conduct cross-dataset experiments. The results are listed in the

Table 2: Performance comparison on OIQA database. The best FR and NR metrics are highlighted in bold.

		JPEG			JP2K			WN			BLUR			ALL		
		PLCC	SROCC	RMSE	PLCC	SROCC	RMSE	PLCC	SROCC	RMSE	PLCC	SROCC	RMSE	PLCC	SROCC	RMSE
FR	PSNR	0.6941	0.7060	1.6141	0.8632	0.7821	1.1316	0.9547	0.9500	0.5370	0.9282	0.7417	0.8299	0.5812	0.5226	1.7005
	S-PSNR [34]	0.6911	0.6148	1.6205	0.9205	0.7250	0.8757	0.9503	0.9357	0.5620	0.8282	0.7525	1.0910	0.5997	0.5399	1.6721
	WS-PSNR [28]	0.7133	0.6792	1.5713	0.9344	0.7500	0.9128	0.9626	0.9500	0.4890	0.8190	0.7668	1.1172	0.5819	0.5263	1.6994
	CPP-PSNR [35]	0.6153	0.5362	1.7693	0.8971	0.7250	0.9904	0.9276	0.9143	0.6739	0.7969	0.7185	1.1728	0.5683	0.5149	1.7193
	SSIM [30]	0.9077	**0.9008**	0.9406	0.9783	0.9679	0.4643	0.8828	0.8607	0.8474	0.9926	0.9777	0.2358	0.8718	0.8588	1.0238
	MS-SSIM [31]	**0.9102**	0.8937	**0.9288**	0.9492	0.9250	0.7052	0.9691	0.9571	0.4452	0.9251	0.8990	0.7374	0.7710	0.7379	1.3308
	FSIM [36]	0.8938	0.8490	1.0057	0.9699	0.9643	0.5454	0.9170	0.8893	0.7197	**0.9914**	**0.9902**	**0.2544**	0.9014	0.8938	0.9047
	DeepQA [13]	0.8301	0.8150	1.2506	**0.9905**	**0.9893**	**0.3082**	**0.9709**	**0.9857**	**0.4317**	0.9623	0.9473	0.5283	**0.9044**	**0.8973**	**0.8914**
NR	BRISQUE [21]	0.9160	0.9392	0.8992	0.7397	0.6750	1.5082	**0.9818**	0.9750	**0.3427**	0.8663	0.8508	0.9697	0.8424	0.8331	1.1261
	BMPRI [20]	0.9361	0.8954	0.7886	0.8322	0.8214	1.2428	0.9673	0.9821	0.4572	0.5199	0.3807	1.6584	0.6503	0.6238	1.5874
	DB-CNN [39]	0.8413	0.7346	1.2118	0.9755	0.9607	0.4935	0.9772	0.9786	0.3832	0.9536	0.8865	0.5875	0.8852	0.8653	0.9717
	MC360IQA [27]	0.9459	0.9008	0.7272	0.9165	0.9036	0.8966	0.9718	0.9464	0.4251	0.9526	0.9580	0.5907	0.9267	0.9139	0.7854
	VGCN (local) [32]	0.9508	0.8972	0.6949	0.9793	0.9439	0.4541	0.9682	0.9714	0.4515	**0.9838**	**0.9759**	**0.3479**	0.9529	0.9444	0.6340
	AHGCN	**0.9669**	**0.9348**	**0.5722**	**0.9884**	**0.9607**	**0.3402**	0.9706	**0.9893**	0.3944	0.9833	**0.9759**	0.3538	**0.9682**	**0.9647**	**0.5225**

Table 3: Performance comparison on CVIQD database. The best FR and NR metrics are highlighted in bold.

		JPEG			AVC			HEVC			ALL		
		PLCC	SROCC	RMSE	PLCC	SROCC	RMSE	PLCC	SROCC	RMSE	PLCC	SROCC	RMSE
FR	PSNR	0.8682	0.6982	8.0429	0.6141	0.5802	10.5520	0.5982	0.5762	9.4697	0.7008	0.6239	9.9599
	S-PSNR [34]	0.8661	0.7172	8.1008	0.6307	0.6039	10.3760	0.6514	0.6150	8.9585	0.7083	0.6449	9.8564
	WS-PSNR [28]	0.8572	0.6848	8.3465	0.5702	0.5521	10.9841	0.5884	0.5642	9.5473	0.6729	0.6107	10.3283
	CPP-PSNR [33]	0.8585	0.7059	8.3109	0.6137	0.5872	10.5615	0.6160	0.5689	9.3009	0.6871	0.6265	10.1448
	SSIM [30]	0.9822	0.9582	3.0468	0.9303	0.9174	4.9029	0.9436	0.9452	3.9097	0.9002	0.8842	6.0793
	MS-SSIM [31]	0.9636	0.9047	4.3355	0.7960	0.7650	8.0924	0.8072	0.8011	6.9693	0.8521	0.8222	7.3072
	FSIM [36]	**0.9839**	**0.9639**	**2.8928**	**0.9534**	**0.9439**	**4.0327**	**0.9617**	**0.9532**	**3.2385**	0.9340	0.9152	4.9864
	DeepQA [13]	0.9526	0.9001	4.9290	0.9477	0.9375	4.2683	0.9221	0.9288	4.5694	**0.9375**	**0.9292**	**4.8574**
NR	BRISQUE [21]	0.9464	0.9031	5.2442	0.7745	0.7714	8.4573	0.7548	0.7644	7.7455	0.8376	0.8180	7.6271
	BMPRI [20]	0.9874	0.9562	2.5597	0.7161	0.6731	9.3318	0.6154	0.6715	9.3071	0.7919	0.7470	8.5258
	DB-CNN [39]	0.9779	0.9576	3.3862	0.9564	0.9545	3.9063	0.8646	0.8693	5.9335	0.9356	0.9308	4.9311
	MC360IQA [27]	0.9698	**0.9693**	3.9517	0.9487	0.9569	4.2281	0.8976	0.9104	5.2557	0.9429	0.9428	4.6506
	VGCN (local) [32]	**0.9857**	0.9666	**2.7310**	0.9684	0.9622	3.3328	0.9367	0.9422	4.1329	0.9597	0.9539	3.9220
	AHGCN	0.9841	0.9602	2.8778	**0.9826**	**0.9786**	**2.4834**	**0.9513**	**0.9512**	**3.6409**	**0.9658**	**0.9617**	**3.6184**

Table 4. As shown in Table 4, the proposed AHGCN is superior to other state-ofthe-art IQA methods by a large margin. This indicates that the proposed AHGCN is well-generalized.

4.5 Ablation study

To validate each component of the proposed method, we conduct ablation experiments. The experimental results are shown in Table 5.

Effectiveness of hierachical features. B_a and B_b use adaptive hypergraph convolutional network (AHGCN) to model interactions between viewports, but they use different features of viewports. Compared with B_b, B_a performs better on OIQA and CVIQD datasets. This confirms that hierachical features are more informative than high-level ones.

Effectiveness of AHGCN. B_a, B_c and B_d use AHGCN, fully-connected network (FC) and GCN to model interactions between viewports, respectively. Compared with B_c, B_a and B_d perform better on OIQA and CVIQD datasets. This is because FC cannot capture dependencies between viewports. Compared to B_d, B_a has higher PLCC and SROCC and lower RMSE. This shows that AHGCN is superior to GCN in modeling interactions between viewports.

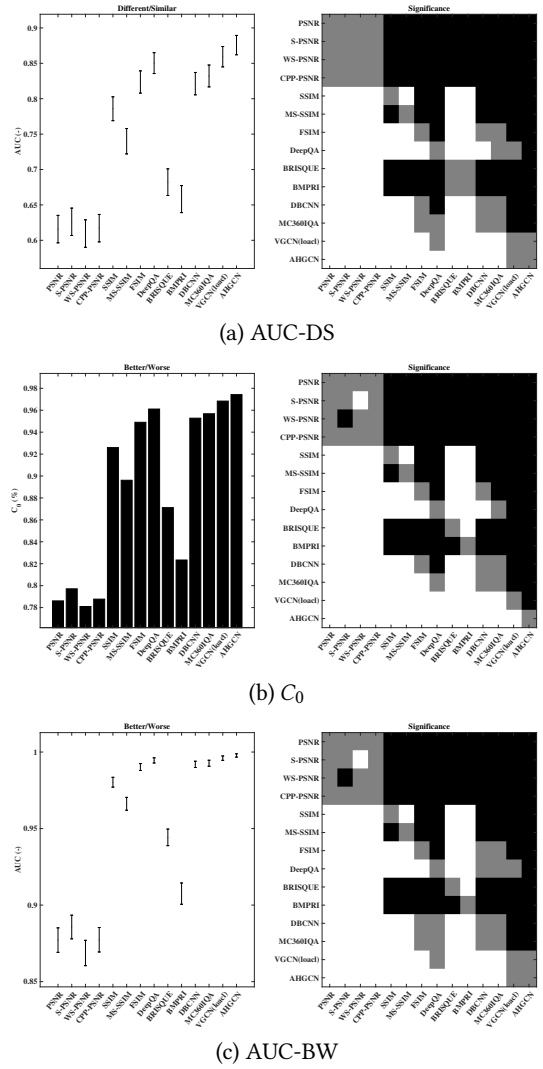


Figure 6: The results of Krasula methodology on the CVIQD database.

Effectiveness of content-based hyperedges. Table 6 shows the impact of hyperparameter k on the performance of the proposed method. $k = 0$ indicates that content-based hyperedges are not considered. As shown in Table 6, the proposed method achieves the best performance at the settings of $k = 5$ and $k = 15$ on CVIQD and OIQA datasets. This indicates that content-based hyperedges are useful for NR 360IQA.

5 CONCLUSION

In this paper, we propose an adaptive hypergraph convolutional neural network for NR 360IQA, dubbed as AHGCN. It designs a multi-level viewport descriptor to extract hierarchical representations from viewports, and models interactions between viewports through hypergraphs instead of graphs. In the hypergraph construction, it considers both the locations and content features of viewports. Experimental results demonstrate that the proposed AHGCN achieves state-of-the-art performance and shows an impressive generalization capability across a broad of distortion types.

Table 4: Cross-dataset Validation. The best performance are highlighted in bold.

Method	Train OIQA/Test CVIQD			Train CVIQD/Test OIQA		
	PLCC	SROCC	RMSE	PLCC	SROCC	RMSE
BRISQUE [21]	0.7543	0.6891	9.3805	0.6816	0.5238	1.5471
BMPRI [20]	0.8007	0.7492	8.5600	0.6483	0.5890	1.6097
DBCNN [39]	0.7896	0.7684	8.7669	0.5817	0.5299	1.7198
MC360IQA [27]	0.8886	0.8629	6.5526	0.4375	0.3329	1.9012
VGCN [32]	0.9241	0.9050	5.4616	0.7911	0.7832	1.2934
AHGCN	**0.9436**	**0.9366**	**4.6223**	**0.8258**	**0.7867**	**1.1785**

Table 5: Ablation study on each component of the proposed method.

Method		B_a	B_b	B_c	B_d
Hierarchical features		✓		✓	✓
High-level features			✓		
FC				✓	
GCN					✓
AHGCN		✓	✓		
CVIQD	PLCC	0.9658	0.9559	0.9607	0.9647
	SROCC	0.9617	0.9489	0.9535	0.9574
	RMSE	3.6184	4.1017	3.8774	3.6770
OIQA	PLCC	0.9682	0.9301	0.9425	0.9594
	SROCC	0.9647	0.9164	0.9306	0.9516
	RMSE	0.5225	0.7677	0.6985	0.5891

Table 6: The impact of k on the performance of AHGCN. The best performance are highlighted in bold.

k	CVIQD			OIQA		
	PLCC	SROCC	RMSE	PLCC	SROCC	RMSE
0	0.9647	0.9578	3.6755	0.9607	0.9557	0.5797
5	**0.9658**	**0.9617**	**3.6184**	0.9612	0.9580	0.5767
10	0.9624	0.9561	3.7936	0.9659	0.9638	0.5412
15	0.9619	0.9540	3.8176	**0.9682**	**0.9647**	**0.5225**
20	0.9614	0.9547	3.8398	0.9631	0.9604	0.5625

6 ACKNOWLEDGEMENTS

This work was supported in part by NSFC under Grant U1908209, 62021001 and the National Key Research and Development Program of China 2018AAA0101400.

REFERENCES

[1] Sebastian Bosse, Dominique Maniry, Klaus-Robert Müller, Thomas Wiegand, and Wojciech Samek. 2017. Deep neural networks for no-reference and full-reference image quality assessment. *IEEE Transactions on image processing* 27, 1 (2017), 206–219.

[2] Sijia Chen, Yingxue Zhang, Yiming Li, Zhenzhong Chen, and Zhou Wang. 2018. Spherical structural similarity index for objective omnidirectional video quality assessment. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1–6.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[4] Huiyu Duan, Guangtao Zhai, Xiongkuo Min, Yucheng Zhu, Yi Fang, and Xiaokang Yang. 2018. Perceptual quality assessment of omnidirectional images. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 1–5.

[5] Facebook. [n.d.]. Quality assessment of 360 video view sessions. ([n. d.]). https://engineering.fb.com/video-engineering/quality-assessment-of-360-video-view-sessions/

[6] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. 2019. Hypergraph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 3558–3565.

[7] Jun Fu, Zhibo Chen, Xiaoming Chen, and Weiping Li. 2020. Sequential Reinforced 360-Degree Video Adaptive Streaming With Cross-User Attentive Network. *IEEE Transactions on Broadcasting* (2020).

[8] Video Quality Experts Group et al. 2003. Final report from the video quality experts group on the validation of objective models of video quality assessment, phase II. *2003 VQEG* (2003).

[9] Ke Gu, Guangtao Zhai, Xiaokang Yang, and Wenjun Zhang. 2014. Hybrid no-reference quality metric for singly and multiply distorted images. *IEEE Transactions on Broadcasting* 60, 3 (2014), 555–567.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[11] Jianwen Jiang, Yuxuan Wei, Yifan Feng, Jingxuan Cao, and Yue Gao. 2019. Dynamic Hypergraph Neural Networks.. In *IJCAI*. 2635–2641.

[12] Hak Gu Kim, Heoun-taek Lim, and Yong Man Ro. 2019. Deep Virtual Reality Image Quality Assessment with Human Perception Guider for Omnidirectional Image. *IEEE Transactions on Circuits and Systems for Video Technology* (2019).

[13] Jongyoo Kim and Sanghoon Lee. 2017. Deep learning of human visual sensitivity in image quality assessment framework. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1676–1684.

[14] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[15] Lukáš Krasula, Karel Fliegel, Patrick Le Callet, and Miloš Klíma. 2016. On the accuracy of objective image and video quality models: New methodology for performance evaluation. In *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 1–6.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* 25 (2012), 1097–1105.

[17] Chen Li, Mai Xu, Xinzhe Du, and Zulin Wang. 2018. Bridge the gap between VQA and human behavior on omnidirectional video: A large-scale dataset and a deep learning model. (2018), 932–940.

[18] Chen Li, Mai Xu, Lai Jiang, Shanyi Zhang, and Xiaoming Tao. 2019. Viewport Proposal CNN for 360deg Video Quality Assessment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 10177–10186.

[19] Heaun-Taek Lim, Hak Gu Kim, and Yang Man Ra. 2018. VR IQA NET: Deep virtual reality image quality assessment using adversarial learning. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 6737–6741.

[20] Xiongkuo Min, Guangtao Zhai, Ke Gu, Yutao Liu, and Xiaokang Yang. 2018. Blind image quality estimation via distortion aggravation. *IEEE Transactions on Broadcasting* 64, 2 (2018), 508–517.

[21] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. 2012. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing* 21, 12 (2012), 4695–4708.

[22] Ekta Prashnani, Hong Cai, Yasamin Mostofi, and Pradeep Sen. 2018. Pieapp: Perceptual image-error assessment through pairwise preference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1808–1817.

[23] Hamid R Sheikh, Muhammad F Sabir, and Alan C Bovik. 2006. A statistical evaluation of recent full reference image quality assessment algorithms. *IEEE Transactions on image processing* 15, 11 (2006), 3440–3451.

[24] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* 15, 1 (2014), 1929–1958.

[25] Simeng Sun, Tao Yu, Jiahua Xu, Jianxin Lin, Wei Zhou, and Zhibo Chen. 2021. GraphIQA: Learning Distortion Graph Representations for Blind Image Quality Assessment. *arXiv preprint arXiv:2103.07666* (2021).

[26] Wei Sun, Ke Gu, Siwei Ma, Wenhan Zhu, Ning Liu, and Guangtao Zhai. 2018. A large-scale compressed 360-degree spherical image database: From subjective quality evaluation to objective model comparison. In *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 1–6.

[27] Wei Sun, Xiongkuo Min, Guangtao Zhai, Ke Gu, Huiyu Duan, and Siwei Ma. 2019. MC360IQA: A Multi-channel CNN for Blind 360-degree Image Quality Assessment. *IEEE Journal of Selected Topics in Signal Processing* (2019).

[28] Yule Sun, Ang Lu, and Lu Yu. 2017. Weighted-to-spherically-uniform quality evaluation for omnidirectional video. *IEEE signal processing letters* 24, 9 (2017), 1408–1412.

[29] David C Van Essen and John HR Maunsell. 1983. Hierarchical organization and functional streams in the visual cortex. *Trends in neurosciences* 6 (1983), 370–375.

[30] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on image processing* 13, 4 (2004), 600–612.

[31] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. 2003. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, Vol. 2. IEEE, 1398–1402.

[32] Jiahua Xu, Wei Zhou, and Zhibo Chen. 2020. Blind omnidirectional image quality assessment with viewport oriented graph convolutional networks. *IEEE Transactions on Circuits and Systems for Video Technology* (2020).

[33] Mai Xu, Chen Li, Zhenzhong Chen, Zulin Wang, and Zhenyu Guan. 2018. Assessing visual quality of omnidirectional videos. *IEEE Transactions on Circuits and Systems for Video Technology* (2018).

[34] Matt Yu, Haricharan Lakshman, and Bernd Girod. 2015. A framework to evaluate omnidirectional video coding schemes. In *2015 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 31–36.

[35] Vladyslav Zakharchenko, Kwang Pyo Choi, and Jeong Hoon Park. 2016. Quality metric for spherical panoramic video. In *Optics and Photonics for Information Processing X*, Vol. 9970. International Society for Optics and Photonics, 99700C.

[36] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. 2011. FSIM: A feature similarity index for image quality assessment. *IEEE Transactions on Image Processing* 20, 8 (2011), 2378–2386.

[37] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 586–595.

[38] Ruochi Zhang, Yuesong Zou, and Jian Ma. 2019. Hyper-SAGNN: a self-attention based graph neural network for hypergraphs. *arXiv preprint arXiv:1911.02613* (2019).

[39] Weixia Zhang, Kede Ma, Jia Yan, Dexiang Deng, and Zhou Wang. 2018. Blind image quality assessment using a deep bilinear convolutional neural network. *IEEE Transactions on Circuits and Systems for Video Technology* (2018).

[40] Yingxue Zhang, Yingbin Wang, Feiyang Liu, Zizheng Liu, Yiming Li, Daiqin Yang, and Zhenzhong Chen. 2018. Subjective panoramic video quality assessment database for coding applications. *IEEE Transactions on Broadcasting* 64, 2 (2018), 461–473.

[41] Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. 2006. Learning with hypergraphs: Clustering, classification, and embedding. *Advances in neural information processing systems* 19 (2006), 1601–1608.

[42] Wei Zhou, Jiahua Xu, Qiuping Jiang, and Zhibo Chen. 2021. No-Reference Quality Assessment for 360-degree Images by Analysis of Multi-frequency Information and Local-global Naturalness. *arXiv preprint arXiv:2102.11393* (2021).

[43] Yufeng Zhou, Mei Yu, Hualin Ma, Hua Shao, and Gangyi Jiang. 2018. Weighted-to-Spherically-Uniform SSIM Objective Quality Evaluation for Panoramic Video. In *2018 14th IEEE International Conference on Signal Processing (ICSP)*. IEEE, 54–57.