



Two-stream network with viewport selection for blind omnidirectional video quality assessment

Junhao Chen¹ · Yuzhen Niu¹

Received: 28 November 2022 / Revised: 30 March 2023 / Accepted: 25 April 2023 /
Published online: 22 June 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

In omnidirectional images or videos, the viewer receives an interactive and immersive experience from the viewport by changing the viewing angle. Due to the wide application of omnidirectional videos, the visual quality assessment for omnidirectional videos is becoming an urgent issue. Due to the large resolution of an omnidirectional video, regions with object motions usually catch the viewers' attention, so the motion regions have great influences on the visual quality perception. Since the number of potential viewports is huge and the viewer spends varying amounts of time for different viewports, viewport selection is a critical yet not resolved problem for omnidirectional video quality assessment (VQA). In this paper, we propose a two-stream network with viewport selection for blind omnidirectional VQA to incorporate the influences of motion regions and viewport selection. Firstly, we propose a two-stream multi-task convolutional neural network (TSMT) for VQA at any viewport, which uses video frame sequences and motion sequences as inputs. The motion sequences are represented as horizontal and vertical optical flows. Based on the observation that the low latitude regions, the front view, and the moving objects have higher possibilities that appearing in the viewport, we propose a viewport selection method based on a fusion-based saliency map that considers those regions. Experimental results on two datasets demonstrated that the proposed model outperforms state-of-the-art omnidirectional VQA methods.

Keywords Omnidirectional video · Video quality assessment · Viewport selection · Two-stream convolutional neural network

1 Introduction

Omnidirectional/360° images or videos provide an omnidirectional interactive experience. The viewer changes the viewing angles to get a viewport through a headset, mobile device,

✉ Yuzhen Niu
yuzhenniu@gmail.com

Junhao Chen
fz.junhao.chen@gmail.com

¹ Fujian Key Laboratory of Network Computing and Intelligent Information Processing, Fuzhou University, 350108 Fuzhou, China

or standard computer screen, thus achieving the experience of being in a virtual reality (VR) space. Omnidirectional data is widely used in interactive applications such as tourism, education, and sports. Since the omnidirectional image or video reproduces the entire visual world in the projection plane, the resolution of the omnidirectional video and image requires a resolution of 4K or higher to meet the viewer experience. And the bandwidth consumption of the omnidirectional video is 80 times that of the 2D video [21], bringing huge challenges to omnidirectional video compression and transmission [7]. To evaluate the visual quality affected by processes such as compression, the need for omnidirectional video quality assessment (VQA) methods has become increasingly urgent.

In recent years, with the introduction of the omnidirectional VQA dataset [18, 33], omnidirectional VQA attracts a lot of attention from both research and industry communities. The VQA methods of omnidirectional video are proposed along with the video datasets. Some works analyze various essential omnidirectional video viewing characteristics, such as the low latitude regions [38], and the front view angles [33] are more attractive to users. These viewing characteristics can guide the design of omnidirectional VQA. Compared with subjective VQA methods of omnidirectional video, which require large labor and time costs, the demand for objective VQA methods becomes urgent. However, the existing objective omnidirectional VQA methods have the following two problems.

Firstly, temporal cues of videos have not been well utilized for omnidirectional VQA. Common temporal artifacts [24], including ghosting, jitter, etc., can greatly affect the viewers' viewing experience. In addition, exceptional motion [12] and judder effect [20], can also affect the immersive experience in omnidirectional video. However, the above-mentioned temporal cues have not been well considered in existing omnidirectional VQA methods. In recent years, some methods consider both temporal cues and spatial cues using the two-stream networks [6, 26, 32, 41] or pseudo-3d residual (P3D) networks [23] for video classification, action recognition, and feature representation tasks. These methods validate the effectiveness of temporal cues for these computer vision tasks. In Fig. 1, we show two examples of omnidirectional video frames and the corresponding moving object maps. We can see that the temporally changing moving objects have high probability values that catch viewers' attention.

Secondly, for data-driven deep learning methods, the sampling method for view angles or crop positions is a critical and not well-addressed problem. Existing sampling methods mainly

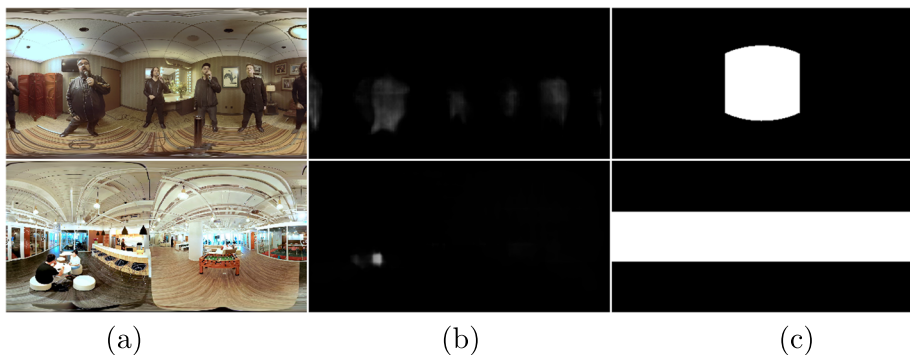


Fig. 1 Examples of omnidirectional video frames. (a) Video frames from omnidirectional videos in ERP. (b) The moving object maps correspond to the videos. (c) The top figure shows the mask of the front view; the bottom figure shows the mask of the low latitude region

include viewport predication and image patch selection in Equirectangular Projection (ERP). The viewport-based methods need to predict the viewer's head movement [19] or specify the viewing point [29]. For patch-based-ERP methods, the patch crop strategy with a certain space and time interval that is commonly used for 2D video quality assessment is not a good option for omnidirectional VQA, because the omnidirectional video owns different viewing characteristics. Specifically, the viewer can only see part of the omnidirectional video from the viewing angle, while the viewer can obtain the entire viewing plane in the 2D video, therefore omnidirectional video sampling method should consider this significant difference. For example, subjective omnidirectional VQA found that viewers tend to watch the front view [33], and the front view is more correlated to the subjective score than the back view. And in statistics, the viewers' visual attention is more frequently located in the low latitude regions [38]. As shown in Fig. 1, image regions within the front view or low latitude (as indicated by Fig. 1(c)) also have high probability values that catch viewers' attention.

In this paper, we propose a blind two-stream multi-task convolutional neural network (TSMT) for the omnidirectional VQA based on a viewport selection method that comprehensively considers the low latitude, front view, and motion regions. Inspired by works [6, 26, 32], we propose using the two-stream network to extract features from both the spatial domain and the temporal domain, then combining those features for predicting the quality score and distortion type of the omnidirectional video. Specifically, the RGB frame carries visual information including scenes, objects, spatial artifacts, etc. The motion information represented by the optical flow includes motion, temporal artifacts, etc. Both of these serve as input to the two-stream network. Since omnidirectional video distortion types, including projection type, compression type, etc., directly affect the visual experience, we regard the classification of distortion type as one of the goals of the network.

Furthermore, we present a viewport selection method that comprehensively considers the low latitude, front view, and motion regions. We first calculate a saliency map that comprehensively considers the low latitude, front view, and motion regions, which usually catch viewers' attention. Then viewport positions are gradually selected according to the saliency map. To avoid selecting redundant viewports, we update the saliency map after each viewport selection. Experimental results show that the performance of our proposed TSMT is superior to the state-of-the-art (SOTA) blind omnidirectional video quality assessment methods, and our proposed TSMT even achieves comparable results with the full-reference (FR) omnidirectional VQA methods.

The main contributions of the proposed method are in three folds:

1. We propose a two-stream multi-task blind omnidirectional video quality assessment method that can extract and combine the features from both spatial and temporal domains.
2. We present a viewport selection method that comprehensively considers the low latitude, front view, and motion regions, which usually catch viewers' attention.
3. Experimental results demonstrate that our method outperforms the SOTA blind omnidirectional VQA methods, and achieves comparable results with the FR VQA methods.

2 Related work

In this section, we describe some objective omnidirectional VQA methods that are most related to our proposed method. For more methods, please refer to the review [34].

Objective VQA methods can be divided into FR, reduced-reference (RR), and no-reference (NR, a.k.a, blind) according to the use of reference videos. FR-VQA, RR-VQA, and NR-

VQA methods predict the video quality score by regarding all, partial, and no reference videos, respectively.

The classic FR-VQA methods, such as the method based on PSNR, were first introduced into the omnidirectional VQA. S-PSNR-NN [38] calculates a certain number of sampling points that are uniformly distributed on the sphere, then maps the sampling points from the projection surface through the nearest neighbors, and calculates the PSNR of the sampling points. CPP-PSNR [40] projects the reference video and the impaired video to the parabolic projection, and the PSNR in parabolic projection is calculated. WS-PSNR [28] utilizes the region stretching ratio of the projected surface to the sphere of some uniformly distributed sampling points, and calculates the PSNR on the projection surface through the ratio without projecting back to the sphere. Xu et al. [33] proposed NCP-PSNR, which weights the viewing regions by the distribution of viewing directions and uses the content-based CP-PSNR to predict the viewing direction by region of interest. SSIM has also been introduced into the omnidirectional VQA method. S-SSIM [3] is an SSIM method that is calculated in the sphere domain. WS-SSIM [5] extends the traditional SSIM method by combining the region stretching ratio between various projection planes and spheres. S-PSNR-NN, CPP-PSNR, and WS-PSNR are all recommended as omnidirectional VQA of JVET [42].

The deep learning technique further improves the performance of the omnidirectional VQA methods. According to the network input, omnidirectional VQA methods follow either of the two categories: patch-based and viewport-based methods. Patch-based methods generally sample the image patches directly from the projection plane like ERP. Chen et al. [18] proposed a full reference method for improving quality prediction through head position and viewport map (VQA-HMEM). Lim et al. [17] proposed a no-reference Generative adversarial network (GAN) (VR-IQA) to predict the quality score, using human visual perception to distinguish the actual quality score from the predicted quality score for adversarial learning. Kim et al. [13] further proposed a GAN-based omnidirectional image quality assessment method using the location and visual features of image patches.

The viewport-based methods need to project the omnidirectional videos to a 2D plane according to the prediction of angles or specified angles. Chen et al. [19] proposed predicting the position of the head/viewport by sphere convolution and predicting the saliency map for the viewport as an auxiliary task of the VQA (V-CNN). Sun et al. [29] proposed to predict the quality score of omnidirectional images by viewing planes from multiple angles (MC-CNN). Xu et al. [36] proposed a multi-viewport omnidirectional image quality assessment method based on angle sampling. Kim et al. [12] proposed a deep GAN to predict VR fatigue. Xu et al. [35] proposed a graph convolutional network to simulate the viewing interaction of omnidirectional image (VGCN), and then predict the score of the viewport. Chai et al. [4] proposed to fuse the quality of single-frame and inter-frame information for blind VQA (NR-OVQA).

3 Proposed TSMT method

The proposed viewport selection method and the framework of the proposed two-stream multi-task network are shown in Figs. 2 and 3, respectively. The proposed model takes the omnidirectional video frames and the two directions' optical flows corresponding to the frames as input. In this way, the temporal motion cues carried by the optical flows are used

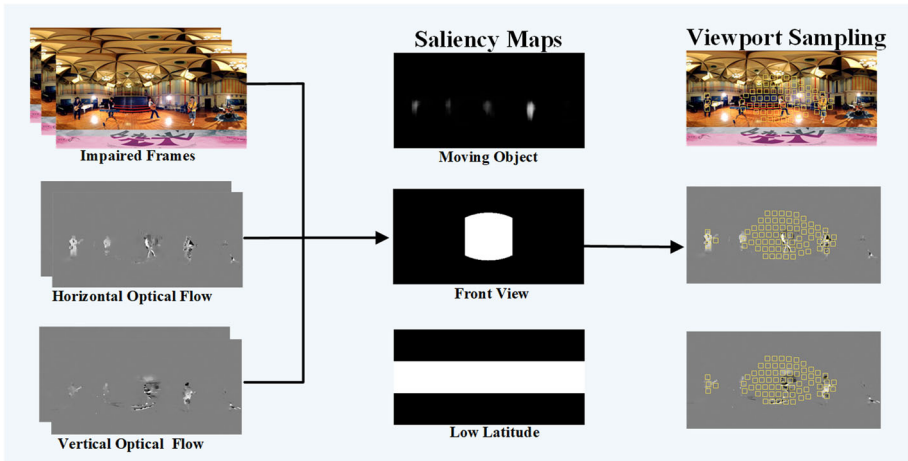


Fig. 2 Proposed viewport selection for our TSMT method. The input is some impaired video frames and the corresponding optical flows of horizontal and vertical directions. The proposed viewport selection method fuses three cues from moving objects, front view, and low latitude to locate the viewports. Then the proposed two-stream multi-task network predicts a quality score and a distortion type via the RGB image patch and optical flows of each selected viewport

as a supplement to the RGB patches, as shown in the left part of Fig. 2. The saliency value of each pixel in the saliency map is calculated by fusing the moving object, the front view, the low latitude, and the smoothing constraints, as shown in the middle part of Fig. 2. The saliency value of each pixel represents the probability that the pixel is sampled or selected as the center of the viewport. Moving object regions, low latitudes regions, and front view regions are more frequently to be sampled because they usually affect the viewing experience, while regions in stationary backgrounds, high latitudes, and other views are less viewed and therefore are less sampled, as shown in the right part of Fig. 2. The sampled RGB patches and the corresponding optical flow patches are fed to the two-stream multi-task network, as shown in Fig. 3. The goals of this multi-task network include quality-score prediction and distortion-type classification.

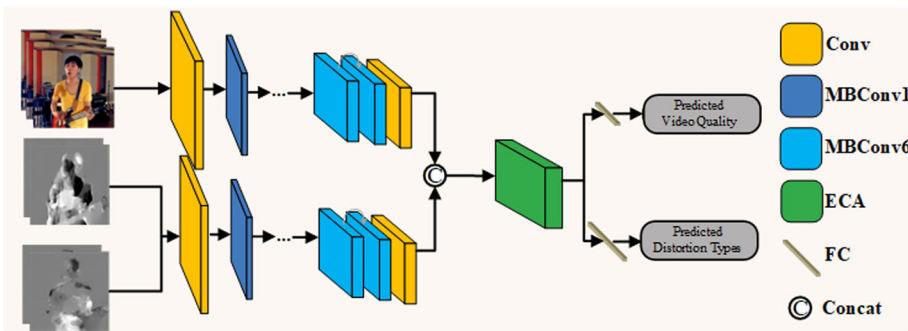


Fig. 3 Our proposed two-stream multi-task network predicts quality score and distortion type via the RGB and optical flows of viewports

3.1 Two-stream framework of omnidirectional VQA

3.1.1 Architecture

The architecture of the proposed two-stream multi-task network is shown in Fig. 3. Compared to the two-stream classification models [6, 26, 32] using VGG [27] or Resnet [9] as the backbone, we use Efficient-Net [30] as the backbone, because Efficient-Net has fewer parameters and runs faster on many existing tasks. We take the first 4 layers of EfficientNet-B0 to extract features.

In Fig. 4, we show the stream taking RGB image patch as input, and the other stream uses a same structure. After RGB image patches and optical flow patches are fed into the two-stream efficient model, they go through the first 4 layers of the EfficientNet-B0 baseline network. Specifically, one stream network includes a convolutional layer, four mobile inverted bottleneck convolution (MBConv) layers [30], namely an MBConv1 and three MBConv6, and finally a convolutional layer. After these layers, the output feature maps of the image patches and the optical flows are concatenated and fed into an Efficient Channel Attention (ECA), including a convolution layer, batch normalization, dropout, and average pooling. Finally, the output of ECA is used to predict the quality score and distortion type through two fully connected layers.

It is worth noting that, different from RGB image patch that have three channels, the optical flow has only two channels, so the channel of the corresponding convolution kernel is set to 2, and the weights are initialized using the average of the RGB convolutional kernel.

3.1.2 Loss function

According to the uncertainty among multiple tasks [11], we use task-dependent uncertainty to determine the weights of our multi-task loss, specifically through the Gaussian distribution likelihood estimation of the predicted quality score and the softmax likelihood estimation of the predicted distortion type. For the video quality score prediction task, the predicted score

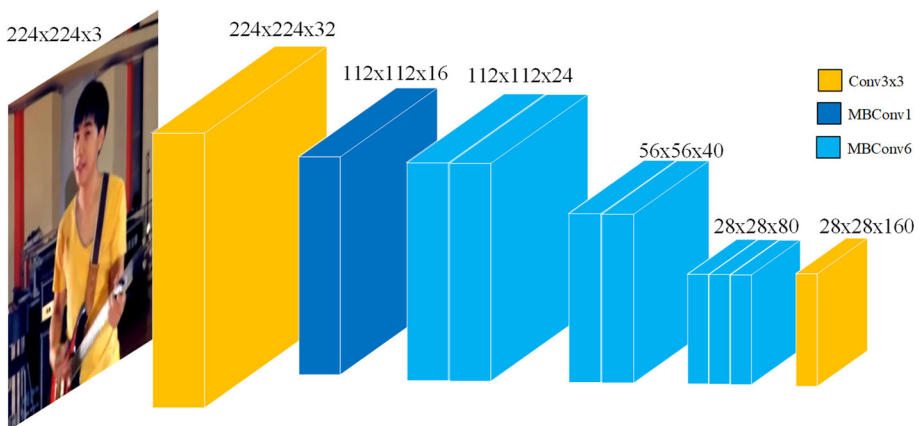


Fig. 4 Single-stream of two-stream network

\hat{y}_1 for the video sample x is obtained through the fully connected layer FC_{sc} . The l_2 norm loss between \hat{y}_1 and the ground truth video quality score y_1 is calculated as follows,

$$L_1(W) = \sum |\hat{y}_1 - y_1|^2, \tag{1}$$

where W corresponds to the model parameters related to the video quality score prediction task.

For the distortion type classification task, we use the last fully connected layer FC_{cls} and a softmax function to predict the output probability \hat{y}_2 that the video sample x is distorted by the c distortion type. The cross-entropy loss is defined as follows,

$$L_2(W) = -\sum y_2 \log \hat{y}_2, \tag{2}$$

where y_2 is the ground truth probability for the video sample, and W are the model parameters related to the distortion type classification task.

We follow the task-dependent uncertainty [11], and the likelihood is defined as follows:

$$p(y_1, y_2 = c | f^W(x)) = \mathcal{N}(y_1; f^W(x), \sigma_1^2) \cdot \text{Softmax}(y_2 = c; f^W(x), \sigma_2), \tag{3}$$

where y_1 is the target score, y_2 is the target category, and $f^W(x)$ is the output of images x under parameter W . For the score regression task we assume that the likelihood is a Gaussian distribution with a mean of $f^W(x)$ and a variance of σ_1 . The classification task’s likelihood is defined as a Boltzmann distribution with a scaling factor of σ_2 .

The derivation of (3) leads to the loss function, specifically we maximize the likelihood function in (3) by minimizing the negative logarithm of it as follows:

$$\begin{aligned} & -\log(p(y_1, y_2 = c | f^W(x))) \\ &= -\log(\mathcal{N}(y_1; f^W(x), \sigma_1^2) \cdot \text{Softmax}(y_2 = c; f^W(x), \sigma_2)) \\ &\propto \frac{1}{2\sigma_1^2} |y_1 - f^W(x)|^2 + \log \sigma_1 - \log p(y_2 = c | f^W(x), \sigma_2). \end{aligned} \tag{4}$$

According to the task-dependent uncertainty [11], the log likelihood of a softmax loss can be written as follows:

$$\begin{aligned} \log p(y_2 = c | f^W(x), \sigma_2) &= \frac{1}{\sigma_2} f_c^W(x) - \log \sum_{c'} \exp(\frac{1}{\sigma_2} f_{c'}^W(x)) \\ &= \frac{1}{\sigma_2} f_c^W(x) - \log \frac{\sum_{c'} \exp(\frac{1}{\sigma_2} f_{c'}^W(x))}{(\sum_{c'} \exp(f_{c'}^W(x)))^{\frac{1}{\sigma_2}}} \\ &\approx \frac{1}{\sigma_2} f_c^W(x) - \log \sigma_2, \end{aligned} \tag{5}$$

substituting (1), (2) and (5) into (4), the loss function can be obtained as follows:

$$\begin{aligned} p(y_1, y_2 = c | f^W(x)) &= \frac{1}{2\sigma_1^2} L_1(W) + \frac{1}{\sigma_2} L_2(W) + \log \sigma_1 + \log \frac{\sum_{c'} \exp(\frac{1}{\sigma_2} f_{c'}^W(x))}{(\sum_{c'} \exp(f_{c'}^W(x)))^{\frac{1}{\sigma_2}}} \\ &\approx \frac{1}{2\sigma_1^2} L_1(W) + \frac{1}{\sigma_2} L_2(W) + \log \sigma_1 + \log \sigma_2. \end{aligned} \tag{6}$$

By the derived loss function, these two learnable parameters σ_1 and σ_2 come to be able to adjust the weights between multi-tasking. For example, a large σ_1 reduces the distribution of $L_1(W)$, while the $\log \sigma_1$ part penalizes an oversized σ_1 . According to work [11], in the implementation, the learnable parameter is set as $\log \sigma^2$ to avoid zeroes when the σ is divided.

3.1.3 Network input

First of all, the optical flow of the video is obtained through a re-implemented optical flow algorithm TVL1 [39] based on OpenCV and CUDA [32]. We calculated the full-resolution horizontal and vertical optical flow maps on the GPU, and then normalized the optical flow maps to discrete values from 0 to 255. After obtaining the position of the viewport, we crop the viewport projected from the video frame to obtain RGB channels and optical flow channels. The input of the network is the three-channel color image patch and the spliced two-channel optical flow patch.

3.2 Fusion-based saliency map

As shown in Fig. 5, we observe that heatmaps of all viewers' head positions (Fig. 5(c)) have a strong correlation with the front view, low latitude regions, and moving object regions (Fig. 5(b)). In this paper, we propose to calculate the saliency map for viewport selection based on the correlation between the viewer's behavior and these regions. In order to combine the moving object regions, the front view, and the low latitude regions and get a smooth saliency map, we propose calculating the saliency map using an energy function. Specifically, we introduce a smoothness constraint to the energy function to get a smooth saliency map.

We formulate the problem of calculating pixel saliency values as optimizing pixel weights. A saliency value is designed to be a continuous value within the range of [0, 1] in the energy function. The saliency map of an omnidirectional video can be computed by minimizing an energy function composed of moving region constraint, front view constraint, low latitude constraint, and the smoothness constraint, as follows:

$$E = \sum_{i=1} w_i^m (s_i - 1)^2 + \lambda_f \sum_{i=1} w_i^f (s_i - 1)^2$$

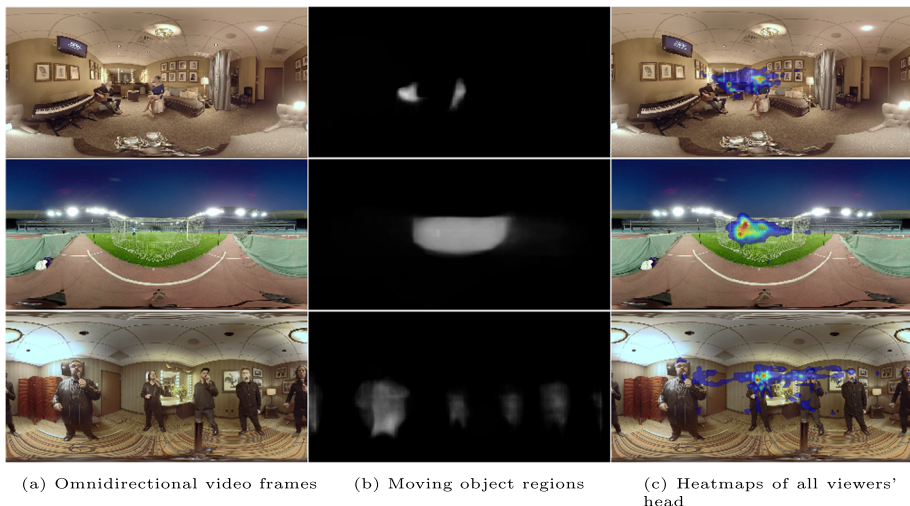


Fig. 5 A demonstration of the relationship between viewer behavior and the regions of moving objects, low latitude, and the front view. The moving object regions shown in (b) are obtained by accumulating the moving object segmentation results [10] of the video frames

$$+\lambda_l \sum_{i=1} w_i^l s_i^2 + \lambda_n \sum_{i,j \in N_i} w_{ij} (s_i - s_j)^2, \quad (7)$$

where s_i and s_j are the saliency values of pixels p_i and p_j , λ_f , λ_l , and λ_n are balancing weights for different constraints, w_m , w_f , w_l are the motion map, front view map, and low latitude map respectively, w_i^m , w_i^f , and w_i^l are the motion value, the front view value, and the low-latitude value of pixel p_i , respectively. The last term in the energy function is the smoothness constraint, and w_{ij} is the smooth-constraint value between pixels p_i and p_j , and p_j is a neighboring pixel of p_i , which is indicated by p_i 's neighboring pixel set N_i .

For the moving region constraint, we first compute the object segmentation results from the optical flows by [10], then accumulate the segmentation results and obtain the moving object region map w^m . The motion value w_i^m is the value in motion map w^m for pixel p_i . The moving region constraint encourages a pixel p_i with a larger w_i^m to have a larger saliency value s_i , which is close to 1.

Likewise, the front view constraint also increases the saliency values of the pixels in the front view regions. Specifically, the front view map w^f is computed by setting the center of the viewport's longitude and latitude as $(0^\circ, 0^\circ)$ for the front viewing angle and the viewport's size as 90° .

The low latitude constraint encourages pixels p_i in low latitude regions to have a saliency value that is not limited by the latitude, while encourages pixels p_i in high latitude regions to have a saliency value that is small and close to 0. The low latitude map w^l is calculated as the distance from the pixel to the two boundaries b_d and b_u of the low-latitude regions, and is defined as follows:

$$w_i^l = \frac{|l_i - b_d| + |l_i - b_u| - (b_u - b_d)}{z}, \quad (8)$$

where b_d and b_u correspond to latitudes of -30° and 30° , following work [38], respectively, z is a parameter used to normalize w_i^l to the range of $[0,1]$, and $z = 2 \times (90^\circ - b_u)$. l_i is the latitude of pixel p_i , define as $l_i = \frac{r_i}{W} \times 180^\circ$, r_i corresponds to the abscissa of p_i , and W corresponds to the width of the image. For the pixel p_i belonging to the low-latitude regions $[b_d, b_u]$, the value of w_i^l is 0, and the w_i^l of the pixels in the remaining regions increases as the distance between the latitude and the low-latitude regions increases.

With the smoothness term, neighboring pixels obtain saliency values that are similar each another. The smooth-constraint value w_{ij} between pixels p_i and its neighbor p_j is computed using their motion values as follows:

$$w_{ij} = \exp\left(-\left(w_i^m - w_j^m\right)^2\right) + u, \quad (9)$$

where w_i^m and w_j^m correspond to the motion values of pixels p_i and p_j , and u is a constant which is set to 0.1.

The above four terms are all quadratic errors, which can be solved by the least quadratic calculation. In our implementation, λ_f , λ_l , and λ_n are set to 0.7, 0.5 and 0.5, respectively. In Fig. 6, we demonstrate some input video frames (a), the moving object region maps (b), and the saliency maps that combine the four constraints using heatmaps (c).

3.3 Viewport selection

We propose a viewport method based on the greedy algorithm to locate the viewport positions in accordance with the saliency map obtained by solving the energy function. We prefer the viewport position at the current moment to be close to the previous viewport position to

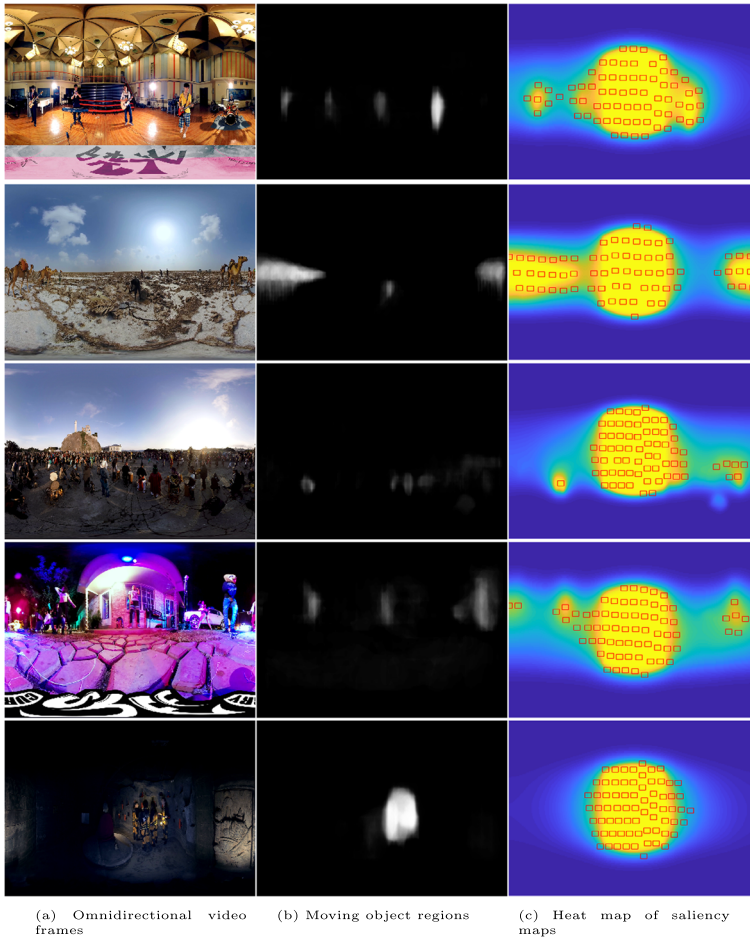


Fig. 6 Demonstration of video frames (a), the moving object region maps (b), and the saliency maps that combine the four constraints using heatmaps and the selected viewpoints (c)

simulate the continuous viewport shifting when watching an omnidirectional video, so our viewport selection algorithm prefers the current viewport to be near the previous viewport. Meanwhile, if all the viewports close to the previous viewport have low saliency values, we choose the viewport globally. In this way, our viewport selection method can simulate viewers' behavior.

At the very beginning, we select the first viewport position as the pixel that has the maximum saliency value in saliency map w by (7), denoted as viewport V_1 , which is centered at the maximum value pixel and with width VW and height VH . The global saliency map w^g for global search and the local saliency map w^l for local search is initialize by the saliency map w . Then we select the viewport iteratively until we select N viewports. For example, at the k_{th} iteration, to avoid selecting a viewport position within the previously selected viewport, we set the pixel values in the saliency map of the viewport selected in the $(k-1)_{th}$ iteration, denoted as viewport V_{k-1} . Then we select two viewport positions, including one global viewport position candidate cg from the updated global saliency map w^g and one

Algorithm 1 Viewport selection

Input: The saliency map w from (7),
The number of viewports to be select N .

Output: The set of selected viewports C .

- 1: Initialize iteration number k as 1;
- 2: Initialize global saliency map w^g and local saliency map w^l by the saliency map w ;
- 3: Initialize the set of selected viewports C as \emptyset ;
- 4: **while** $k \leq N$ **do**
- 5: $cl \leftarrow \operatorname{argmax}_{\{cl|w^l(cl) \in w^l\}} w^l$; // viewport position candidate from local search
- 6: $cg \leftarrow \operatorname{argmax}_{\{cg|w^g(cg) \in w^g\}} w^g$; // viewport position candidate from global search
- 7: $V^k \leftarrow \max(w_{cl}^l + \Delta w, w_{cg}^g)$; // Select viewport position V^k from cl and cg
- 8: $w^l \leftarrow w^g(\operatorname{Neighborhood}(V^k))$; // Set w^l as the neighborhood of V^k in w^g
- 9: $w^g(\operatorname{Viewport}(V^k)) \leftarrow -1$; // Update the pixel values in the viewport centered at w^g as -1
- 10: $C \leftarrow C \cup V^k$;
- 11: $k \leftarrow k + 1$;
- 12: **end while**

local viewport position candidate cl from the neighborhood of viewport V_{k-1} in the updated local saliency map w^l . The width and height of the neighborhood are set to be larger than the width and height of the viewport and smaller than the width and height of the video frame. Then if the difference between cg and cl is less than a predefined threshold Δw , we choose the viewport centered at cl as viewport V_k , otherwise, we choose the viewport centered at cg as viewport V_k according to Algorithm 1. The neighborhood that corresponds to the VH and VW ranges of w^g yields the new local saliency map. Set viewport corresponding to V_k in the global saliency map to -1, so that the new viewport position will not return to the original viewport position. Finally, the obtained image patches and optical flow patches corresponding to the selected viewports are then fed into the two-stream network.

4 Experiments

In order to evaluate the performance of our proposed two-stream networks for omnidirectional VQA, we conducted extensive experiments on two omnidirectional VQA datasets, including VQA-ODV [18] and VR-VQA48 [33]. On VQA-ODV dataset, we compare and evaluate our proposed model with some SOTA methods. The VR-VQA48 dataset is smaller than the VQA-ODV dataset, and the distortion type of VR-VQA48 belongs to the distortion type of VQA-ODV, so we only test on VR-VQA48 to verify the generalization ability of the model trained on the trainset of VQA-ODV dataset. In order to verify the effectiveness of the proposed fusion-based saliency map, comparison experiments are conducted, including random dicing, using only moving objects, front view, and low latitude regions as the saliency map. Additionally, comparisons between local and global greedy searches are carried out, along with comparisons to SOTA methods.

4.1 Datasets

VR-VQA48 contains 12 reference omnidirectional videos from YouTube and VRCun, and collects subject scores from 40 subjects. H.265 with 3 different quantization parameters (QP) 27, 37, 42 are used to generate 36 impaired videos. The duration of the reference video and the

impaired video is 12 seconds, the frame rate is 25 fps, and the resolution of all videos under ERP projection is 4096×2048 . The dataset provides the subjective score and the differential mean opinion score (DMOS) of each viewer. We use reverse DMOS (rDMOS) [19] as the ground truth of the predicted score.

VQA-ODV is a large omnidirectional VQA dataset. VQA-ODV contains 60 reference omnidirectional videos from the YouTube virtual reality channel. The impaired video of VQA-ODV is generated through 3 compression levels and 3 types of projection, so each reference video corresponds to 9 different impaired videos. Compared with VR-VQA48, which only considers the compression level, VQA-ODV further considers projection formats, including ERP, RCMP [22], and TSP [8]. For the compression level of VQA-ODV, the impaired video is encoded as quantization parameters 27, 37, and 42. According to QP, the bit rates are at high, medium, and low levels, respectively. Since the subject experiments of the 540 impaired videos in VQA-ODV are divided into 10 groups, we follow [19] to use the average value of the valid DMOS of each group as the DMOS. Same to VR-VQA48, rDMOS is used as the ground truth of the predicted score.

4.2 Performance indicators

After the network predicts the score results of all selected viewpoints of the same video, the average pooling is further adopted for computing the final video quality score. rDMOS and objective VQA scores were evaluated using five metrics, including Spearman Rank Order Correlation Coefficient (SROCC), Pearson Linear Correlation Coefficient (PLCC), root mean square error (RMSE), and mean absolute error (MAE). SROCC, KROCC, and SROCC measure rank correlation, while PLCC, MAE, and RMSE assess prediction accuracy. For SROCC, KROCC, and PLCC, the higher the value is, the closer to the subjective score. Whereas for RMSE and MAE, the lower the value is, the closer to the subjective score. SROCC and KROCC can be directly calculated by objective VQA and rDMOS. Before calculating PLCC, RMSE and MAE, we fit objective VQA to rDMOS according to a 4-parameter logistic function [25] as follows:

$$f(x) = \beta_2 + \frac{\beta_1 - \beta_2}{1 + e^{-\left(\frac{x - \beta_3}{|\beta_4|}\right)}}, \quad (10)$$

where β_i is the fitting parameter from the objective VQA method to rDMOS, β_i is initialized according to [1], x is score predicted by the objective VQA method, and $f(x)$ is the fitted score.

4.3 Implement details

All experiments are conducted on the PyTorch deep learning framework [2], using the stochastic gradient descent algorithm with Adam optimizer [16] to update the parameters. We set the learning rate as 0.0001, the weight decay as 0.0005 for regularization, and the batch size as 64.

In order to train the two-stream network at the same time, we use EfficientNet-B0, which occupies less GPU memory, and use part of the convolutional layer of the model for feature extraction. For the RGB branch, the complete EfficientNet-B0 parameters are adopted for initialization. Since the optical flow is stored as a single-channel image in two directions, the first convolutional layer parameters for the flow stream are initialized as the average of the first convolutional layer parameters of EfficientNet-B0, and the subsequent convolutional

layers are initialized using the EfficientNet-B0 parameters. In a single-channel experiment, the RGB stream using the pre-trained parameters performs better than the randomly initialized model. The flow stream using the pre-trained parameters is slightly better than the randomly initialized model.

For the VQA-ODV dataset, we sample one frame from every few frames to obtain the sampled video frames and sampled optical flows, following method [14], we sample every eight frames for videos with 24 FPS, and every ten frames for videos with 30 FPS. For each video frame and optical flow, we select 70 viewports and follow V-CNN [19] to use the center crop method to generate each viewport with size of 224×224 . Following method [19], we separate the dataset into a training set, validation set, and testing set. Specifically, a total of 387, 45, and 108 impaired omnidirectional videos were used as the training set, validation set, and testing set. Because the VQA-ODV dataset has three compression parameters and mapping types, and a total of 9 combinations of compression and mapping, we use the classification of distortion types as one of the tasks of the model. The target score for the regression task is set to rDMOS. For the RGB and optical flow single-channel experiments, we use the complete Efficient-B0 as the feature extraction part, because a deeper single-branch network in the experiment performs better than a shallower network. For VR-VQA48, we use the same sampling method and use 36 impaired videos as the test set. The network is trained for a total of 36 epochs. For the VR-VQA48 dataset, we use the model trained on the VQA-ODV dataset to test 36 impaired videos.

4.4 Performance comparison

In order to validate the performance of our proposed model, we compare it with S-PSNR, WS-PSNR, CPP-PSNR, BP-QAVR [37], VR-IQA-NET, VQA-HMEM [18], DeepQA [15], WaDOQaM-FR [2], V-CNN [19], VGCN [35] and NR-OVQA [4] on the VQA-ODV dataset. S-PSNR, WS-PSNR, and CPP-PSNR are classic omnidirectional image/video visual quality assessment methods. The remaining methods are based on deep neural networks. DeepQA

Table 1 Performance comparison between our proposed method and other SOTA methods in performance indicators including PLCC, SROCC, KROCC (larger is better), RMSE and MAE (smaller is better) on VQA-ODV dataset

	PLCC	SROCC	KROCC	RMSE	MAE
S-PSNR	0.6929	0.6976	0.4981	8.5407	6.6810
WS-PSNR	0.6721	0.6839	0.4860	8.7707	6.9089
CPP-PSNR	0.6812	0.6896	0.4912	8.6718	6.7932
BP-QAVR	0.6588	0.6801	0.4780	8.9112	7.0823
VQA-HMEM	0.7821	0.7953	0.5902	7.3817	5.7793
VR-IQA-NET	0.3713	0.3379	0.2260	10.9984	9.1010
DeepQA	0.6936	0.7296	0.5213	8.5325	6.7720
WaDOQaM-FR	0.6207	0.6162	0.4206	9.2868	7.4574
V-CNN	0.8740	0.8962	0.7137	5.7551	4.4893
VGCN	0.8032	0.8122	0.6144	7.0562	5.4088
NR-OVQA	0.7598	0.7972	0.6286	7.7006	4.9496
TSMT	0.8946	0.9011	0.7214	5.2922	4.2443

Table 2 Performance comparison on VR-VQA48 dataset

	PLCC	SROCC	RMSE	MAE
S-PSNR	0.589	0.639	9.518	7.692
WS-PSNR	0.556	0.596	9.938	8.097
CPP-PSNR	0.575	0.632	9.592	7.782
BP-QAVR	0.659	0.680	8.911	7.082
TSMT	0.940	0.932	4.215	3.291

and WaDOQaM-FR are full-reference (FR) 2D IQA methods. BP-QAVR, VR-IQA-NET, VQA-HMEM, and V-CNN are omnidirectional VQA methods. VR-IQA-NET, VGCN, and NR-OVQA are NR methods, and the others are FR methods. For methods based on deep neural networks, we do not retrain the VR-IQA-NET and WaDOQaM-FR on the VQA-ODV dataset.

Table 1 shows the comparison between the proposed method and the existing objective VQA methods for omnidirectional VQA performance. In Table 1, our proposed TSMT method, VGCN, NR-OVQA, and VR-IQA-NET are no-reference methods. VR-IQA-NET is a method based on the generative adversarial network. We believe that the lack of retraining is the reason for its lower performance. As shown in Table 1, among the five performance indicators, our proposed TSMT method achieves the best performance.

Table 2 shows the omnidirectional VQA performance on VR-VQA48. We do not train the model using the VR-VQA48 dataset, all the images in the VR-VQA48 dataset are used as testing images. For the experiments on the VR-VQA48 dataset, we use the model trained on the training set of the VQA-ODV dataset. Since the performance values of other methods are not available, we only compare our method with four methods on the VR-VQA48 dataset. As shown in Table 2, our proposed TSMT model achieves the best performance among the comparison methods, indicating a good generalization capability of our proposed TSMT model.

4.5 Ablation study

To investigate the effectiveness of our two-stream architecture, we compare the quantitative performances between the two-stream model and two single-stream models. In Table 3, the single-stream model B0-R uses the color channel only and the model B0-F only uses the optical flow channel on the VQA-ODV dataset. As shown in Table 3, model B0-R achieves better performance than model B0-F. And the two-stream model TSMT achieves the best performance among these three models, indicating the effectiveness of combining color information and temporal motion information.

In order to compare the performance of our proposed saliency map fusion method described in subsection 3.3, in Table 4, we compare the random viewport positioning method

Table 3 Quantitative comparison single-stream and two-stream network

	PLCC	SROCC	KROCC	RMSE	MAE
B0-R	0.8659	0.8446	0.6334	5.8497	4.7213
B0-F	0.8224	0.8083	0.6178	6.6529	5.0012
TSMT	0.8946	0.9011	0.7214	5.2922	4.2443

Table 4 Quantitative comparison among different constraints, the viewpoints produced by different constraints are used as the inputs to our proposed TSMT model

	PLCC	SROCC	KROCC	RMSE	MAE
Random	0.8851	0.8839	0.6819	5.5131	4.4535
Moving object	0.8688	0.8641	0.6674	5.8660	4.5321
Front view	0.8808	0.8874	0.6964	5.6094	4.4294
Low latitude	0.8655	0.8654	0.6594	5.9338	4.6874
TSMT	0.8946	0.9011	0.7214	5.2922	4.2443

(Random), viewport selected using a saliency map only considering the moving object (Moving object), viewport selected using a saliency map only considering the front view (Front view), and viewport selected using a saliency map only considering the low latitude (Low Latitude). As shown in Table 4, all different viewports achieve reasonably good performance, but are inferior to our proposed model. The viewports with the front view and the random viewports achieve better results than the viewports with moving objects and low latitudes. The viewports that are constrained by the front view achieve the second-best SROCC performance. Our model that selects viewports using a fusion-based saliency map outperforms all the other models significantly.

Comparison experiments with other viewport selection methods are also carried out, and the experimental results are shown in Table 5. Specifically, we use the viewport selection methods in MC360 [29] and VGCN [35] to select the viewports, which serve as input of our two-stream network. As shown in Table 5, our proposed model outperforms MC360 and VGCN by large margins.

We also validate the effectiveness of our viewport selection method described in subsection 3.3. Specifically, we compared our viewport selection method with a local saliency search method and a global saliency search method, and show the experimental results in Table 6. The local saliency search method always selects the viewport that is locally close to the previous viewport. While the global saliency search method always selects the viewport globally. As shown in Table 6, the performance of the proposed TSMT framework is significantly better than the local method and the global method.

To investigate the long-range temporal information, a model with 3D convolutional kernels (Cov3D), some models with mixtures of 2D and 3D convolutional kernels (MC x) are also experimented and compared. The experimented model with 3D convolutional kernels use the same architecture as our proposed TSMT, only substituting the 2D convolutional kernels and pooling layers with 3D convolutional kernels and pooling layers. Since the architecture needs to be the same for Cov3D, the sequence length after 3D convolution needs to be greater than 32 so that the temporal dimensions can be continuously halved. We pad the sequence to meet this requirement, with the sequence length set to 32, and the batch size set to 15. Unlike EfficientNet, the 3D convolution does not have a trained model, so the 2D convolution

Table 5 Quantitative verification of the effectiveness of our viewport selection method

	PLCC	SROCC	KROCC	RMSE	MAE
MC360	0.7821	0.7953	0.5902	7.3817	5.7793
VGCN	0.8669	0.8547	0.6514	5.9056	4.6961
TSMT	0.8946	0.9011	0.7214	5.2922	4.2443

Compared viewport selection methods include MC360 [29] and VGCN [35]

Table 6 Quantitative verification of the effectiveness of the saliency map search method

	PLCC	SROCC	KROCC	RMSE	MAE
Local	0.8583	0.8541	0.6504	6.0777	4.9457
Global	0.8815	0.8804	0.6905	5.5935	4.5147
TSMT	0.8946	0.9011	0.7214	5.2922	4.2443

weights are repeatedly and used as the initial parameters for 3D convolution. The comparison results are shown Table 7. The score of the model with 3D convolution is lower than that of 2D convolution.

For the models with mixtures of 2D and 3D convolutional kernels, we adopt the MC_x given in the work [31], the 2D convolutional kernels in the first layers were replaced by 3D convolutional kernels. We experimented with MC1 and MC2, which use 3D convolutional kernels in the first layer and the first two layers, respectively. For the input of the models with MC_x , we experimented with sequences with lengths of 5, 10, and 15 frames, which are denoted as $MC_x/5$, $MC_x/10$, and $MC_x/15$ in Table 7, respectively. As shown in Table 7, MC2 performed better than MC1, and MC2/5 achieved the second highest score. As the input sequence length increased, the performance of the models with MC_x decreased. The inferior performance may be caused by the larger number of parameters and a smaller number of training samples. On the contrary, our proposed TSMT is able to obtain sufficient temporal information through the optical flow on this dataset and achieves the best performance among all experimented models.

In summary, our proposed TSMT model outperforms the SOTA omnidirectional VQA methods. Also, the two-stream network, the saliency map fusion method, and the viewport selection method are validated to be effective for omnidirectional VQA.

5 Conclusion

In this paper, we propose a two-stream multi-task (TSMT) model to assess the quality of omnidirectional video. In the proposed model, a viewport selection method that uses a saliency map fuses the low latitude, front view, and moving object regions is proposed. The image patches and optical flow patches corresponding to the selected viewports are used as the input to our two-stream network. Optical flow is introduced into the video quality assessment to explicitly represent the temporal motion information, which is an effective complement to the

Table 7 Quantitative comparison with models with 3D convolutional kernels and mixtures of 2D and 3D convolutional kernels (MC_x)

	PLCC	SROCC	KROCC	RMSE	MAE
Cov3D	0.5225	0.4984	0.3516	10.0991	8.0502
MC1/5	0.8367	0.8413	0.6317	6.4880	5.6700
MC1/10	0.7357	0.7106	0.5244	8.0227	4.7971
MC1/15	0.7187	0.7132	0.5199	8.2357	5.9622
MC2/5	0.8751	0.8873	0.6964	5.7322	4.4183
MC2/10	0.8468	0.8609	0.6577	6.3015	4.7885
MC2/15	0.8266	0.8464	0.6452	6.7002	4.8122
TSMT	0.8946	0.9011	0.7214	5.2922	4.2443

color information. Experimental results show that the performance of our proposed TSMT model outperforms the state-of-the-art omnidirectional VQA methods.

Acknowledgements This work is supported by the National Natural Science Foundation of China (No.61972097) and the Natural Science Foundation of Fujian Province (No.2020J01494).

Data Availability Date and code will be made available on reasonable request.

Declarations

Conflicts of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Antkowiak J, Jamal Baina T, Baroncini FV, Chateau N, FranceTelecom F, Pessoa ACF, Stephanie Colonese F, Contin IL, Caviedes J, Philips F (2000) Final report from the video quality experts group on the validation of objective models of video quality assessment March 2000
2. Bosse S, Maniry D, Müller K-R, Wiegand T, Samek W (2017) Deep neural networks for no-reference and full-reference image quality assessment. *IEEE Trans Image Process* 27(1):206–219. <https://doi.org/10.1109/TIP.2017.2760518>
3. Boyce J, Alshina E, Abbas A, Yan Y (2018) JVET-E1030: JVET common test conditions and evaluation procedures for 360° video
4. Chai X, Shao F (2021) Blind quality assessment of omnidirectional videos using spatio-temporal convolutional neural networks. *Optik* 226:165887. <https://doi.org/10.1016/j.ijleo.2020.165887>
5. Chen S, Zhang Y, Li Y, Chen Z, Wang Z (2018) Spherical structural similarity index for objective omnidirectional video quality assessment. In: 2018 IEEE International Conference on Multimedia and Expo (ICME), pp 1–6. IEEE
6. Chi L, Tian G, Mu Y, Tian Q (2019) Two-stream video classification with cross-modality attention. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, pp 0–0. <https://doi.org/10.1109/ICCVW.2019.00552>
7. Choi B, Wang Y, Hannuksela M, Lim Y, Murtaza A (2017) Information technology-coded representation of immersive media (mpeg-i)-part 2: Omnidirectional media format. ISO/IEC 23090–2
8. der Auwera GV, Coban M, Karczewicz M (2016) AHG8: Truncated square pyramid projection (tsp) for 360 video. JVET Doc 0071
9. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proc IEEE Conf Comput Vis Pattern Recognit, pp 770–778. <https://doi.org/10.1109/CVPR.2016.90>
10. Jain SD, Xiong B, Grauman K (2017) Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos. In: 2017 IEEE Conf Comput Vis Pattern Recognit (CVPR), pp 2117–2126. <https://doi.org/10.1109/CVPR.2017.228>. IEEE
11. Kendall A, Gal Y, Cipolla R (2018) Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proc IEEE Conf Comput Vis Pattern Recognit, pp 7482–7491. <https://doi.org/10.1109/CVPR.2018.00781>
12. Kim HG, Lim H-T, Lee S, Ro YM (2018) Vrsa net: Vr sickness assessment considering exceptional motion for 360 vr video. *IEEE Trans Image Process* 28(4):1646–1660. <https://doi.org/10.1109/TIP.2018.2880509>
13. Kim HG, Lim H-T, Ro YM (2019) Deep virtual reality image quality assessment with human perception guider for omnidirectional image. *IEEE Transactions on Circuits and Systems for Video Technology* 30(4):917–928. <https://doi.org/10.1109/TCSVT.2019.2898732>
14. Kim W, Kim J, Ahn S, Kim J, Lee S (2018) Deep video quality assessor: From spatio-temporal visual sensitivity to a convolutional neural aggregation network. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 219–234. https://doi.org/10.1007/978-3-030-01246-5_14
15. Kim J, Lee S (2017) Deep learning of human visual sensitivity in image quality assessment framework. In: Proc IEEE Conf Comput Vis Pattern Recognit, pp 1676–1684. <https://doi.org/10.1109/CVPR.2017.213>
16. Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)

17. Lim H-T, Kim HG, Ra, YM (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). pp 6737–6741. <https://doi.org/10.1109/ICASSP.2018.8461317>. IEEE
18. Li C, Xu M, Du X, Wang Z (2018) Bridge the gap between vqa and human behavior on omnidirectional video: A large-scale dataset and a deep learning model. In: Proceedings of the 26th ACM International Conference on Multimedia. pp 932–940. <https://doi.org/10.1145/3240508.3240581>
19. Li C, Xu M, Jiang L, Zhang S, Tao X (2019) Viewport proposal cnn for 360° video quality assessment. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp 10169–10178. <https://doi.org/10.1109/CVPR.2019.01042>. IEEE
20. Mahmoudpour S, Schelkens P (2019) Visual quality analysis of judder effect on head mounted displays. In: 2019 27th European Signal Processing Conference (EUSIPCO). pp 1–5. IEEE
21. Mangiante S, Klas G, Navon A, Guanhua Z, Ran J, Silva MD (2017) VR is on the edge: How to deliver 360° videos in mobile networks. In: Proceedings of the Workshop on Virtual Reality and Augmented Reality Network . <https://doi.org/10.1145/3097895.3097901>
22. Ng K-T, Chan S-C, Shum H-Y (2005) Data compression and transmission aspects of panoramic videos. IEEE Transactions on Circuits and Systems for Video Technology 15(1):82–95. <https://doi.org/10.1109/TCSVT.2004.839989>
23. Qiu Z, Yao T, Mei T (2017) Learning spatio-temporal representation with pseudo-3d residual networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp 5533–5541. <https://doi.org/10.1109/ICCV.2017.590>
24. Seshadrinathan K, Bovik AC (2009) Motion tuned spatio-temporal quality assessment of natural videos. IEEE Trans Image Process 19(2):335–350. <https://doi.org/10.1109/TIP.2009.2034992>
25. Seshadrinathan K, Soundararajan R, Bovik AC, Cormack LK (2010) Study of subjective and objective quality assessment of video. IEEE Trans Image Process 19(6):1427–1441. <https://doi.org/10.1109/TIP.2010.2042111>
26. Shi L, Zhang Y, Cheng J, Lu H (2019) Two-stream adaptive graph convolutional networks for skeletonbased action recognition. In: Proc IEEE/CVF Conf Comput Vis Pattern Recognit. pp 12026–12035. <https://doi.org/10.1109/CVPR.2019.01230>
27. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
28. Sun Y, Lu A, Yu L (2017) Weighted-to-spherically-uniform quality evaluation for omnidirectional video. IEEE signal processing letters 24(9):1408–1412. <https://doi.org/10.1109/LSP.2017.2720693>
29. Sun W, Min X, Zhai G, Gu K, Duan H, Ma S (2019) Mc360iqa: A multi-channel cnn for blind 360-degree image quality assessment. IEEE Journal of Selected Topics in Signal Processing 14(1):64–77
30. Tan M, Le Q (2019) Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning. pp 6105–6114. PMLR
31. Tran D, Wang H, Torresani L, Ray J, LeCun Y, Paluri M (2018) A closer look at spatiotemporal convolutions for action recognition. In: Proc IEEE Conf Comput Vis Pattern Recognit. pp 6450–6459. <https://doi.org/10.1109/CVPR.2018.00675>
32. Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L (2016) Temporal segment networks: Towards good practices for deep action recognition. In: European Conference on Computer Vision. pp 20–36. Springer. https://doi.org/10.1007/978-3-319-46484-8_2
33. Xu M, Li C, Chen Z, Wang Z, Guan Z (2018) Assessing visual quality of omnidirectional videos. IEEE transactions on circuits and systems for video technology 29(12):3516–3530. <https://doi.org/10.1109/TCSVT.2018.2886277>
34. Xu M, Li C, Zhang S, Le Callet P (2020) State-of-the-art in 360° video/image processing: Perception, assessment and compression. IEEE Journal of Selected Topics in Signal Processing 14(1):5–26. <https://doi.org/10.1109/JSTSP.2020.2966864>
35. Xu J, Zhou W, Chen Z (2020) Blind omnidirectional image quality assessment with viewport oriented graph convolutional networks. IEEE Transactions on Circuits and Systems for Video Technology 31(5):1724–1737. <https://doi.org/10.1109/TCSVT.2020.3015186>
36. Xu J, Luo Z, Zhou W, Zhang W, Chen Z (2019) Quality assessment of stereoscopic 360-degree images from multi-viewports. In: 2019 Picture Coding Symposium (PCS). pp 1–5. IEEE
37. Yang S, Zhao J, Jiang T, Wang J, Rahim T, Zhang B, Xu Z, Fei Z (2017) An objective assessment method based on multi-level factors for panoramic videos. In: 2017 IEEE Visual Communications and Image Processing (VCIP). pp 1–4. IEEE
38. Yu M, Lakshman H, Girod B (2015) A framework to evaluate omnidirectional video coding schemes. In: 2015 IEEE International Symposium on Mixed and Augmented Reality. pp 31–36. <https://doi.org/10.1109/ISMAR.2015.12>. IEEE

39. Zach C, Pock T, Bischof H (2007) A duality based approach for realtime tv-l 1 optical flow. In: Joint Pattern Recognition Symposium, pp. 214–223 . Springer. https://doi.org/10.1007/978-3-540-74936-3_22
40. Zakharchenko V, Choi KP, Park JH (2016) Quality metric for spherical panoramic video. In: Optics and Photonics for Information Processing X, vol 9970. p 99700 . <https://doi.org/10.1117/12.2235885>. International Society for Optics and Photonics
41. Zhou T, Li J, Wang S, Tao R, Shen J (2020) Matnet: Motion-attentive transition network for zero-shot video object segmentation. *IEEE Trans Image Process* 29:8326–8338. <https://doi.org/10.1109/TIP.2020.3013162>
42. Zhou Y, Yu M, Ma H, Shao H, Jiang G (2018) Weighted-to-spherically-uniform ssim objective quality evaluation for panoramic video. In: 2018 14th IEEE International Conference on Signal Processing (ICSP), pp 54–57. <https://doi.org/10.1109/ICSP.2018.8652269>. IEEE

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.