sensors

Article

RAVA: Region-Based Average Video Quality Assessment

Xuanyi Wu [1,*], Irene Cheng [1], Zhenkun Zhou [2] and Anup Basu [1]

[1] Department of Computing Science, University of Alberta, Edmonton, AB T6G 2R3, Canada; locheng@ualberta.ca (I.C.); basu@ualberta.ca (A.B.)
[2] Huawei Fields Laboratory, Hangzhou 310000, China; kfzle@126.com
* Correspondence: xuanyi@ualberta.ca

Abstract: Video has become the most popular medium of communication over the past decade, with nearly 90 percent of the bandwidth on the Internet being used for video transmission. Thus, evaluating the quality of an acquired or compressed video has become increasingly important. The goal of video quality assessment (VQA) is to measure the quality of a video clip as perceived by a human observer. Since manually rating every video clip to evaluate quality is infeasible, researchers have attempted to develop various quantitative metrics that estimate the perceptual quality of video. In this paper, we propose a new region-based average video quality assessment (RAVA) technique extending image quality assessment (IQA) metrics. In our experiments, we extend two full-reference (FR) image quality metrics to measure the feasibility of the proposed RAVA technique. Results on three different datasets show that our RAVA method is practical in predicting objective video scores.

Keywords: video quality assessment; objective score; human visual system (HVS); image quality assessment

Citation: Wu, X.; Cheng, I.; Zhou Z.; Basu, A. RAVA: Region-Based Average Video Quality Assessment. *Sensors* **2021**, *21*, 5489. https://doi.org/10.3390/s21165489

Academic Editor: Zhe-Ming Lu

Received: 28 June 2021
Accepted: 11 August 2021
Published: 15 August 2021

1. Introduction

Video is widely used in our daily lives. TV shows, computer games, online meetings, all rely on the quality of video. As mentioned in [1], video sensor networks (VSNs) are communication infrastructures that involve video coding, transmission, and display/storage. Via VSNs, the dense visual information is captured and transmitted to applications on different devices for users to view. The size of video clips is many times larger than that of images and texts. To reduce bandwidth usage and save storage space, video coding is widely used. Recent technologies such as that in [2] have attempted to transmit only the difference in data between the past and current images, while the difference data are calculated by a MPEG-4 Visual video encoder. Some technologies are integrated into the encoders for video compression. For example, a denoising algorithm is combined with a high-efficiency video coding (HEVC) encoder to improve the compression efficiency in [3]. More importantly, if the quality of the network is poor, the perceptual quality continues to deteriorate as fewer packets are received [4]. This is another problem we face during transmission. Before displaying to users, the decoder restores the data to a video. However, the quality of the video is often degraded after such a long process. The question is, "How can we measure quality?".

In the real world, objects are three-dimensional (3D). To measure the perceptual quality, many aspects need to be considered. Resolutions of texture and mesh are combined to measure 3D perceptual quality in [5]; optimized linear combinations of accurate geometry and texture quality measurements are used in [6]; and a multi-attribute computational model optimized by machine learning is used in [7]. For videos, all 3D objects are projected onto a 2D plane. Thus, estimating the quality for videos is simpler than quality estimation for 3D objects. Quality assessment can be performed considering two types of scores—subjective and objective. Users need to manually rate videos to allow the computation of precise subjective scores; however, this process can be very time-consuming.

Thus, researchers have tried to develop objective scores that can estimate subjective scores automatically. Many aspects affect objective scores, such as contrast, frequency, pattern, and color perception. Thus, some metrics are developed to analyze specific features. Objective metrics can be further divided into three categories: full-reference (FR) methods; reduced-reference (RR) methods; and no-reference (NR) methods.

There are many image quality assessment (IQA) methods. For instance, SSIM [8] and PSNR [9] are FR IQA methods, [10,11] and NIMA [12] are NR IQA methods. Since videos are composed of image frames, we consider extending IQA methods to assess video quality. Some VQA methods are extended from IQA methods, including PSNR [9], the extension of PSNR based on HVS (PSNR-HVS) [13], PSNR based on between-coefficient contrast masking (PSNR-HVS-M) [14]), structural similarity image metrics (SSIM) [8], and the extension of SSIM to video structural similarity (VSSIM) [15]). However, existing full-reference VQA methods do not have high correlation with human perception while the video content varies considerably. In addition, some methods simply average the IQA scores for all the frames. By doing this, they only consider spatial features such as color and illumination, but neglect the temporal features. Our goal is to combine motion features and 2D spatial features together. In our work, we extend two IQA methods, namely SSIM [8] and PSNR [9], to obtain the RAVA scores. The reason we choose these IQA metrics are outlined below. SSIM and PSNR are the most widely used IQAs. We want to apply them to video quality evaluation and compare their performance with existing VQA methods, especially with the VQA methods extended by them. We divide each frame into foreground and background regions, calculate the IQA scores for those regions, and then assign them different weights based on the motion features. We also notice that if we linearly combine foreground and background scores as the VQA scores, the range and mean value differ considerably for videos with different content. To generalize the VQA scores for videos with various types of content, we introduced a self-supervised video content distinction network. Finally, the foreground, background, and content distinction features are passed to a support vector regression (SVR) [16] to obtain the final VQA score.

For evaluation, we used the LIVE Mobile VQA database [17], the MCL-V database [18], and the Netflix Public Dataset [19]. The video categories in these datasets are quite varied. These datasets are widely used to analyze video quality. The LIVE Mobile VQA database [17] and the Netflix Public Dataset [19] provide the subjective differential mean opinion score (DMOS), while the MCL-V database [18] provides the mean opinion score (MOS). As mentioned in [20], MOS is a typical subjective quality of experience (QoE) assessment score; for instance, users rate quality from 1 (bad quality) to 5 (excellent quality). It is considered the most accurate way to measure QoE since actual users were involved in developing the metric. DMOS is calculated as first getting the difference scores from the raw quality scores, and then converting the scores into Z-scores with outliers removed [21]—here, lower is better. To evaluate the results, we analyzed Pearson's (linear) correlation coefficient (PCC) [22] and Spearman's rank correlation coefficient (SCC) [23] of the RAVA scores and the subjective scores.

The contributions of our paper are: (1) proposing a region-based VQA method that estimates video quality by extracting and processing the information of background regions and moving objects in the foreground regions; (2) integrating a self-supervised video content distinction network to generalize the VQA scores for videos with different content; (3) extending two full-reference IQA metrics to VQA metrics in the experiments, which shows the possibility of applying the RAVA technique to other FR IQA methods.

2. Related Work

Objective video quality assessment techniques can be categorized into three types: full reference (FR) methods, reduced reference (RR) methods and no reference (NR) methods. FR methods utilize the entire original video to determine the quality score. Nevertheless, their performance is relatively poor in terms of accuracy. Thus, perceptual factors in the human visual system (HVS) need to be incorporated to develop reliable video quality assessment techniques [24]. Some FR VQA methods are as follows.

Netflix proposed video multi-method assessment fusion (VMAF) [19] in 2016. VMAF calculates the visual information fidelity (VIF) [25], detail loss metric (DLM) [26], and a motion feature, which is defined as the average absolute pixel difference for the luminance component between adjacent frames. A support vector regressor (SVR) is subsequently used to fuse these elementary metrics together. In 2018, Netflix posted another blog saying that they added AVX optimization and frame-level multi-threading, which accelerates its execution three times and improves its prediction accuracy [27]. Liu et al. [28] proposed a new VQA metric using space–time slice mappings. They first use spatial temporal slices (STS) [29] to obtain some STS maps. Then, on each of the reference-distorted STS map pairs, they calculated the IQA scores via a full-reference IQA algorithm. Finally, they apply feature pooling on the IQA scores on those maps to obtain the final score. Aabed et al. [30] proposed power spectral density (PSD) [30]. It is a perceptual video quality assessment (PVQA) metric that analyzes the power spectral density of a group of pictures. The authors built 2D time-aggregated PSD (or tempospatial PSD) planes for several sets of frames for both the original and distorted videos to capture spatio-temporal changes in the pixel domain. Following this, they built a local cross-correlation map. The perceptual quality score is the average of the values in the correlation map, with a higher value implying better quality.

RR methods extract some outstanding features from both the original and acquired videos, compare these features and obtain the objective score. For example, the Institute for Telecommunication Science (ITS) proposed the video quality metric (VQM) [31]. It was adopted as the standard by the American National Standards Institute (ANSI) and the International Telecommunication Union (ITU) [31]. VQM is defined in (1):

$$\begin{aligned} VQM = & -0.2097 * si\_loss + 0.5969 * hv\_loss \\ & + 0.2483 * hv\_gain + 0.0192 * chroma\_spread \\ & - 0.3416 * si\_gain + 0.0431 * ct\_ati\_gain \\ & + 0.0076 * chroma\_extreme \end{aligned} \tag{1}$$

Here, h and v represent the horizontal and vertical axes, respectively; *si_loss* detects the loss of or decrease in spatial information; *si_gain* detects edge sharpening or enhancement; *hv_loss* captures the shift of edges from vertical and horizontal orientations to a diagonal orientation; *hv_gain* finds the shift of edges from diagonal to horizontal; *chroma_spread* finds changes in the spread of the distribution of 2D color samples; *chroma_extreme* measures serious localized color impairments; and *ct_ati_gain* is the product of a contrast feature [32].

NR methods access the quality of a new video without referring to the original video. Li et al. [33] proposed VSFA (quality assessment of in-the-wild videos). It integrates two eminent effects of the human visual system: content-dependency and temporal-memory effects. Content-dependency effects are obtained by extracting features from a pre-trained image classification neural network on ImageNet; temporal-memory effects are integrated by adding a gated recurrent unit and a subjectively inspired temporal pooling layer to the neural network. The method does not refer to the original video when predicting the video quality. Zadtootaghaj et al. [34] proposed an NR VQA method DEMI. DEMI first uses the scores predicted by the pre-trained VMAF model [19] for training. Then, it is fine-tuned on a small image quality dataset. Finally, the authors apply random forest for feature pooling.

The problem for existing FR and RR methods is that they do not working well if the content of videos in a dataset varies a lot. The correlation values of many existing FR and RR VQA scores with human perception are low. For the NR methods, the predicted quality tends to be more affected by the content than the distortions. NR methods are often trained and tested with in-the-wild video datasets. The videos there are collected from real-world video sequences. The content does vary significantly, but we are not sure how much distortion is involved.

Since we want to extend some image quality assessment metrics to video, we will introduce the two full-reference IQA metrics we used below.

- PSNR
  Peak signal-to-noise ratio (PSNR) [9] is the ratio between the maximum possible power of a signal and the power of the corrupting noise that affects the fidelity of its representation. A higher value of PSNR is better:

$$PSNR = 20\log_{10}\left(\frac{MAX_f}{\sqrt{MSE}}\right) \tag{2}$$

  where MAX_f is the highest value in the two input variables (it is normally 255 for RGB images) and MSE is the mean squared error of the two inputs.

- SSIM
  SSIM [8] is a perception-based model that considers image degradation as perceived change in structural information, while also incorporating important perceptual phenomena, including both luminance and contrast masking. Structural information is the idea of pixels having strong inter-dependencies, especially when they are spatially close. For SSIM, a higher value is better:

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)} \tag{3}$$

  where μ_x is the average of x, μ_y is the average of y, σ_x^2 is the variance of x, σ_y^2 is the variance of y, σ_{xy} is the covariance of x and y, c_1 and c_2 are two variables to stabilize the division with weak denominator, with $c_1 = (k_1L)^2$, $c_2 = (k_2L)^2$. L is the dynamic range of the pixel-values (typically $L = 2^{\#bits\ per\ pixel} - 1$). $k_1 = 0.01$, $k_2 = 0.03$ by default.

3. Proposed Method

Figure 1 illustrates the main procedure of the proposed method. The key idea is to compute a region-based weighted average, combining spatial and temporal features while integrating content distinction features. At the beginning, each frame is divided into several regions. Motion features are extracted by optical flow [35]. Later, they can be used to define the weights for each region. Following this, larger weights can be assigned to regions with larger motion change and smaller weights can be assigned to regions with smaller motion change. Furthermore, videos with similar content tend to have similar VQA scores. How to assess the quality of videos with different contents has become a problem. Motivated by this, we add a content distinction neural network to generalize the VQA scores for videos with varying content.

3.1. Foreground Features

The first step is to define and find the foreground regions. Humans tend to pay more attention to the foreground objects than the background, so we want to define the regions based on this observation. In our implementation, for every consecutive pair of frames, I_i and I_{i+1}, we first locate the objects in frame I_i with bounding boxes. Then, the boxes are used to approximate those objects' positions in the next frame I_{i+1} and also to define the regions used to calculate the optical flows. Each region in I_i can be represented as

$$R_{i,k} = \begin{cases} x_L \le x \le x_L + w \\ y_L \le y \le y_L + h \end{cases} \tag{4}$$

where k is the kth region in frame I_i, (x_L, y_L) is the top left corner of the bounding box, w is the width and h is the height.

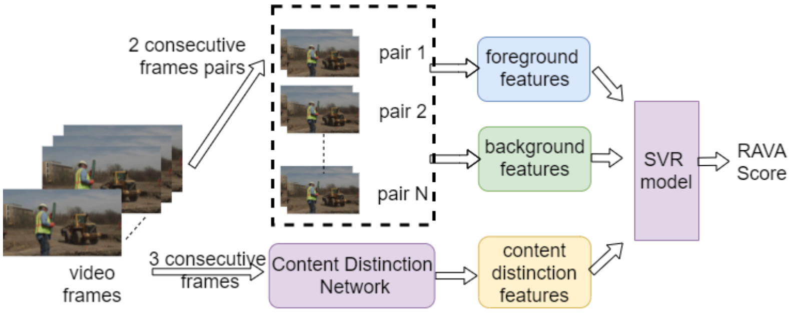


Figure 1. Main procedure.

After finding the regions, we need to determine their weights. Since we are dealing with video quality, we cannot only consider the spatial features. To address the relationship between frames, we use optical flow to calculate the weights, as shown in Figure 2a. Optical flow shows the pattern of apparent motion changes of objects, surfaces, and edges caused by the relative motion between an observer and the scene [36,37]. In addition, it gives us information about the rate of the change of the observer. The rate of change is an important factor in the votes of subjects. If this rate between frames is large, it may cause blurring and affect the viewing experience. Thus, we believe that using optical flow to assign different weights for various regions can reflect the user's attention in each area to some extent. Flow is represented in polar coordinates, using magnitude and angle. The optical flow for the k^{th} region in frame I_i can be represented as

$$mag_{i,k}, \alpha_{i,k} = f_{R_{i,k} \to R_{i+1,k}} \tag{5}$$

where $R_{i+1,k}$ is the same region in the next frame I_{i+1}. The average magnitude for this region is:

$$avg\_opt_{i,k} = \frac{\sum_{j=1}^{K} mag_{i,k,j}}{K} \tag{6}$$

where j is the iterator to go through the values in $mag_{i,k}$ and K is the number of pixels in it.

We do not consider the foreground regions and weights in the last frame, since optical flow calculation needs two frames. After we obtain the average magnitudes for all the foreground regions, the weight assigned for region $R_{i,k}$ is:

$$w_{i,k} = \frac{avg\_opt_{i,k}}{\sum_{m=1}^{M-1} \sum_{n=1}^{N} avs.g\_opt_{m,n}} \tag{7}$$

where M is the number of frames and N is the number of regions detected in frame m. The foreground feature can be calculated as a weighted average of the foreground region IQAs:

$$FG\ feature = \frac{\sum_{m=1}^{M-1} \sum_{n=1}^{N} w_{m,n} \cdot IQA(R_{m,n})}{\sum_{m=1}^{M-1} \sum_{n=1}^{N} 1} \tag{8}$$

3.2. Background Features

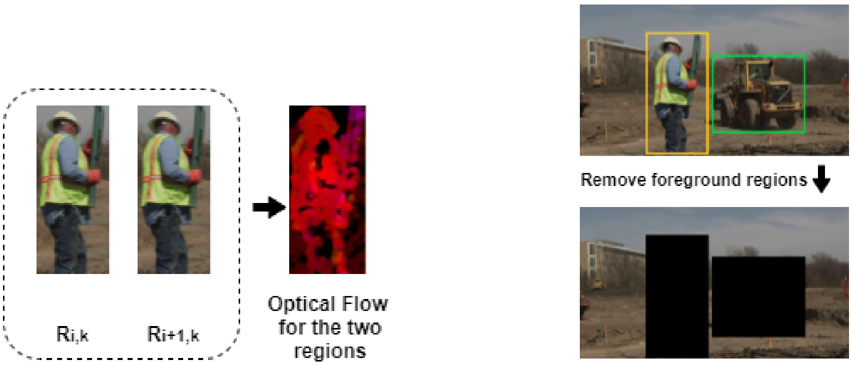
We cannot ignore the background regions, especially for those videos with small or no foreground objects. To extract the background only, we mask out the foreground regions with zeros, as shown in Figure 2b. The background region for frame I_i, which we call BG_i, is defined by the following equations:

$$BG_i = I_i \ominus R_{i,k} \qquad for\ k \in [1, number\ of\ regions] \tag{9}$$

where $\ominus$ is defined as region-based pixelwise manipulation, with the restriction: pixel at location (x, y) in image I_i, namely $I_{i,x,y} = max(0, I_{i,x,y} - R_{i,k,x,y})$.

The background feature is the simple average IQA for background regions:

$$BG\ feature = \frac{\sum_{m=1}^{M} IQA(BG_i)}{M} \tag{10}$$



(**a**) Computing optical flow for a region

(**b**) Removing foreground regions to determine background region

Figure 2. (**a**) Shows how to get the motion feature for the foreground region; and (**b**) defines the background region.

Algorithm 1 shows the pseudocode for getting the foreground and background features.

Algorithm 1 Proposed RAVA algorithm.

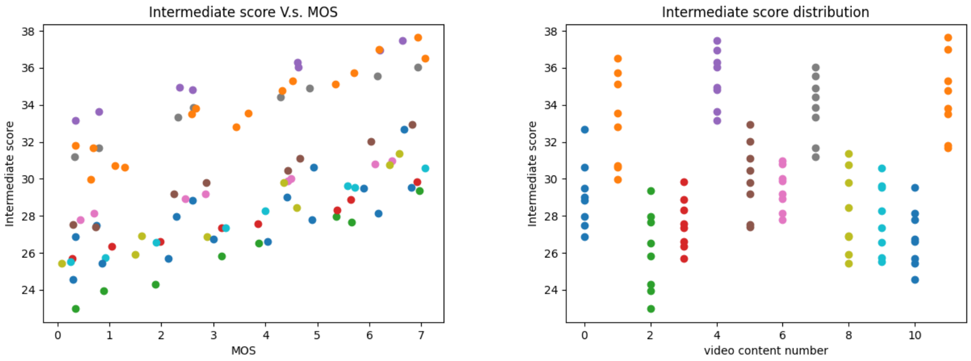
RAVA(I, M): I represents list of all the frames of a video, and M is the number of frames.

```
$i \leftarrow 1$
for all $i \leftarrow 1\ to\ M-1$ do
    $Bboxes_i \leftarrow locate\ objects\ in\ frame\ I_i$
    $BG_i \leftarrow I_i$
    for all $k \leftarrow 1\ to\ len(Bboxes_i)$ do
        $R_{i,k} \leftarrow Bboxes_i[k]$
        $BG_i \leftarrow I_i \ominus R_{i,k}$                    {Mask out foreground regions}
        $mag_{i,k},\ \alpha_{i,k} \leftarrow Optical\ flow(\ R_{i,k} \rightarrow R_{i+1,k})$
        $avg\_opt_{i,k} \leftarrow \frac{\sum_{j=1}^{K} mag_{i,k,j}}{K}$
                                   {K is the number of pixels in this region $R_{i,k}$}
    end for
end for
for all $i \leftarrow 1\ to\ M-1$ do
    for all $k \leftarrow 1\ to\ N$ do
                                   {N is the number of regions in the frame}
        $w_{i,k} = \frac{avg\_opt_{i,k}}{\sum_{m=1}^{M-1}\sum_{n=1}^{N} avs.g\_opt_{m,n}}$

    end for
end for
$FG\ feature = \frac{\sum_{m=1}^{M-1}\sum_{n=1}^{N} w_{m,n} \cdot IQA(R_{m,n})}{\sum_{m=1}^{M-1}\sum_{n=1}^{N} 1}$

$BG\ feature = \frac{\sum_{m=1}^{M} IQA(BG_i)}{M}$
                                   {IQA can be PSNR or SSIM}
```

3.3. *Content Distinction Features and Self-Supervised Learning*

3.3.1. Statistical Analysis

We first attempted to linearly combine the foreground and background features as the intermediate RAVA score. However, the result was not good. Figure 3a shows the plot for the intermediate RAVA$_{PSNR}$ scores vs. MOS on the MCL-V [18] database, where dots with the same colors represent videos distorted from the same raw HD video. We can see that videos with the same content fit a line, however, videos with different contents are scattered. Thus, we performed a statistical analysis to determine the causes. Figure 3b shows the intermediate RAVA$_{PSNR}$ scores for the distorted videos in the MCL-V database. Scores for videos with the same content are in the same bin. There are 12 bins, since there are 12 raw videos. We can notice that even under the same distortions, the distribution for the intermediate scores are different. They have distinct means and ranges. Motivated by this finding, we chose the mean and range to be the content distinction features. There are also some existing works such as MaD-DLS [38], a full-reference image quality assessment method, which analyzes the mean and range (deviation) when designing the metric.



(**a**) Intermediate scores vs. MOS (**b**) Intermediate scores distribution

Figure 3. Statistical analysis.

3.3.2. Self-Supervised Learning for Content Distinction Features

To predict the aforementioned features, we applied transfer learning on ResNet50 [39]. We modified the architecture somewhat. Since we were dealing with the content for videos, the number of layers for the inputs was set to nine instead of three. We used three consecutive frames in a video to predict its content distinction features. We also modified the final fully connected layer to keep only two values, representing the mean and range. The middle layers are the same as ResNet50. Their weights are loaded from the pre-trained ResNet50 on ImageNet, and kept unchanged. Only the weights for the input layer, the average pooling layer, and the final fully connected layer are tuned by learning on the video datasets, as shown in Figure 4. By doing this, the content distinction network can learn faster.

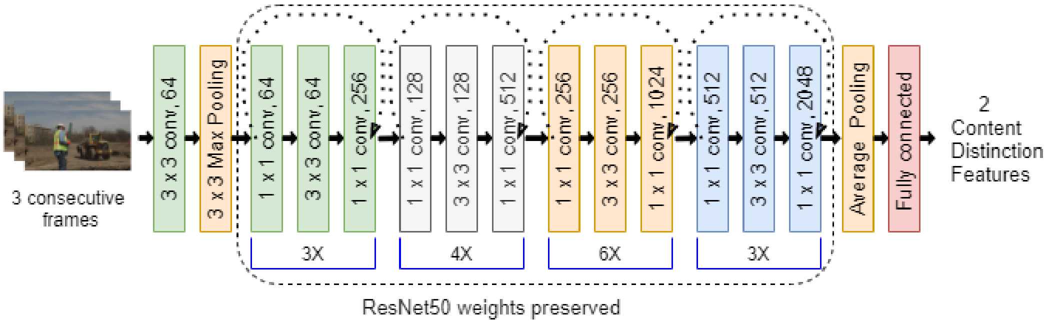


Figure 4. Self-supervised content distinction network.

This is a self-supervised learning process, meaning that we do not train the network with the ground truth (DMOS or MOS). Instead, as mentioned above, we use the means and ranges of the intermediate scores to be the content distinction features. The implementation details are discussed in Section 4.2.2.

3.4. Feature Pooling with the SVR Model

In recent years, machine-learning algorithms became popular for feature pooling, as they can consider the strength of those features and assign different weights. For instance, both the full-reference VQA method VMAF [19,27] and the no-reference VQA method TLVQM [40] use SVR for feature pooling to develop a final metric. In our work, the foreground, background, and content distinction features are passed to a support vector regression (SVR) with an RBF kernel, which allows non-linear mapping.

4. Experimental Results

4.1. Description of Datasets

The proposed RAVA methods were evaluated on two existing datasets: the LIVE Mobile Video Quality Assessment (VQA) Database [17] and the MCL-V database [18]. The LIVE Mobile Video Quality Assessment (VQA) Database consists of 10 RAW HD reference videos and 200 distorted videos (four compression, four wireless packet-loss, four frame-freezes, three rate-adapted and five temporal dynamics per reference). Each video has a resolution of $1280 * 720$ at a frame rate of 30 fps and a duration of 15 s [17]. The study involved over 50 subjects, resulting in 5300 summary subjective scores and time-sampled subjective traces of video quality [17]. The dataset involves two different evaluation approaches to obtain DMOS: mobile and tablet. We analyzed the results for both of these approaches in our experiments. Figure 5 shows some snapshots from this dataset.

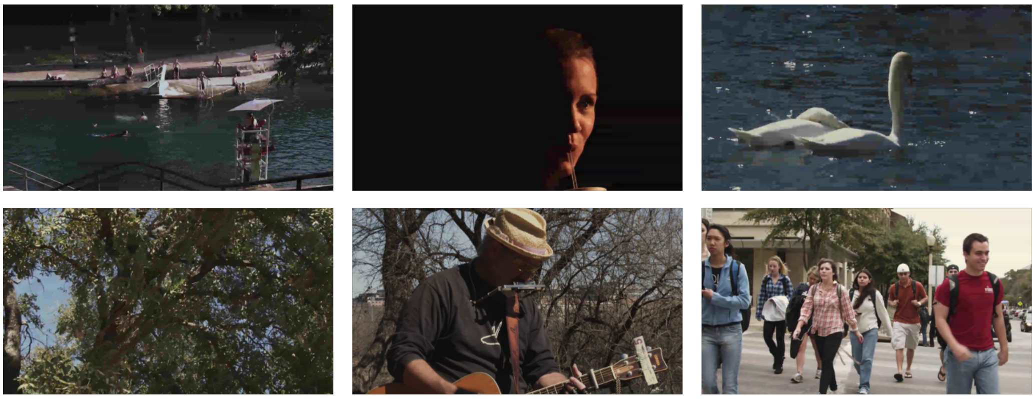


Figure 5. Snapshots from the LIVE database.

The MCL-V database contains 12 uncompressed HD ($1920 * 1080$) source video clips, as shown in Figure 6. Its resolution is higher than the previous dataset. In addition, this dataset captures two typical video distortion types—compression and image size scaling. For each distortion type, four distortion levels are adopted, resulting in 96 distorted video clips in total. Furthermore, its contents are quite varied. There are not only real-life video clips, but also some cartoons and animations. It also provides the mean opinion scores (MOSs) along with the videos.

The Netflix Public Dataset is a full-reference video quality assessment dataset published by Netflix together with their work [19,27]. It consists of nine source video clips of resolution 1920 × 1080 with frame rates ranging from 24 to 30 fps. The source clips are encoded in multiple resolution–bitrate pairs. The bitrates go from 375 to 5800 kbps while the resolution goes from 288 to 1080 p. They also provide the differential mean opinion scores (DMOSs) for the 70 distorted videos. Note that lower DMOS values are normally better, as mentioned in [21]. However, for the DMOSs provided in the Netflix Public

Dataset, higher values are better. Their range varies from 10 (impairments are annoying) to 100 (impairments are imperceptible) [19]. We used this dataset for cross-library evaluation.

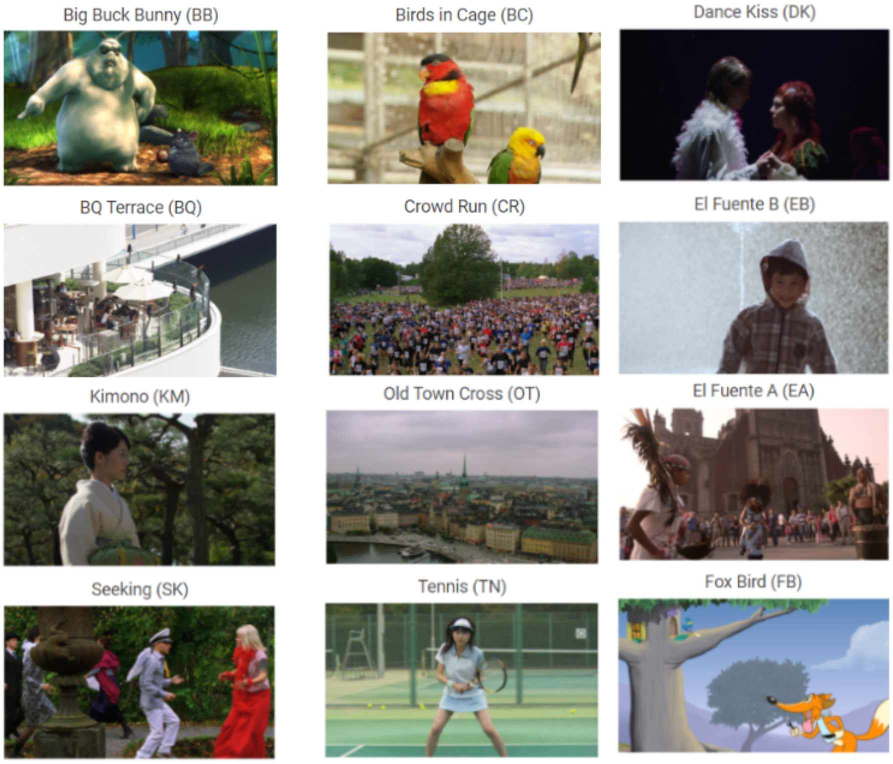


Figure 6. Snapshots from the MCL-V database.

4.2. Implementation Details

4.2.1. Packages

There do exist some advanced object detection and tracking methods such as AdaMM [41] and content-aware focal plane selection [42]. They can handle complex situations such as occlusion. However, in our work, an object does not affect the viewing experience if it is not visible. Moreover, the position of an object does not greatly differ for two consecutive frames, so we simply used YOLOv3 [43] to locate an object with a bounding box and used that location to approximate its position in the next frame. The bounding boxes' widths and heights are offsets from the anchor boxes' centroids. YOLOv3 uses anchor boxes of nine different sizes. The smallest anchor box is of size 10 × 13 so that it can track small objects. If the object is even smaller than this size, then it is viewed as a background, as discussed in Section 3.2. We tried two implementations to calculate the optical flow: dense optical flow provided by OpenCV [44] and FlowNet2 [45]. Their performances are very similar. In the following paragraphs, we showed the result with the first implementation.

4.2.2. Training on the Content Distinction Network

The features are called "content distinction"; thus, regardless of the distortion type, the video with the same content should obtain the same features. Figures 7 and 8 show how we prepare the training data. To obtain the training data for Video 1, we randomly generate 20 groups of frames from each video distorted from it. Each group contains three consecutive frames. For example, if we have eight videos distorted from Video 1, we will have $8 \times 20 = 160$ groups. To save some training time, each frame is resized to 240×240. All the groups will have the same content distinction features, so we expected that they would have the same output from the network. The features we used for training are the mean and range of all the intermediate scores for a video. We repeated the process for all the videos with different content.

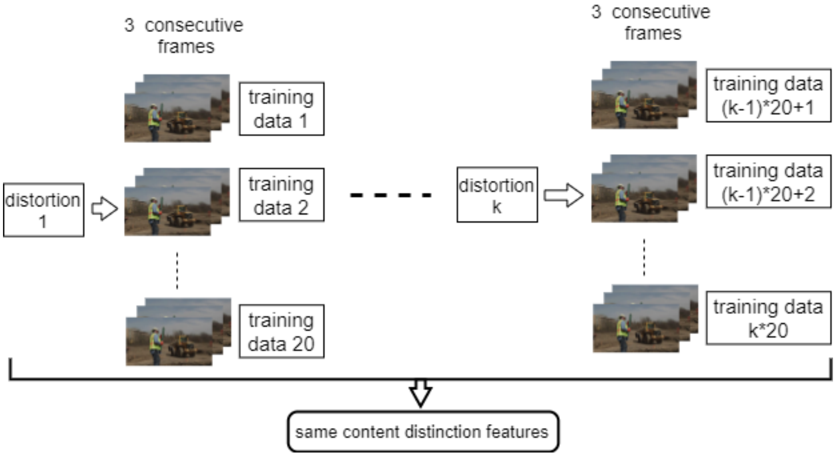


Figure 7. Preparing training data for videos with the same content (distorted from the same RAW video).

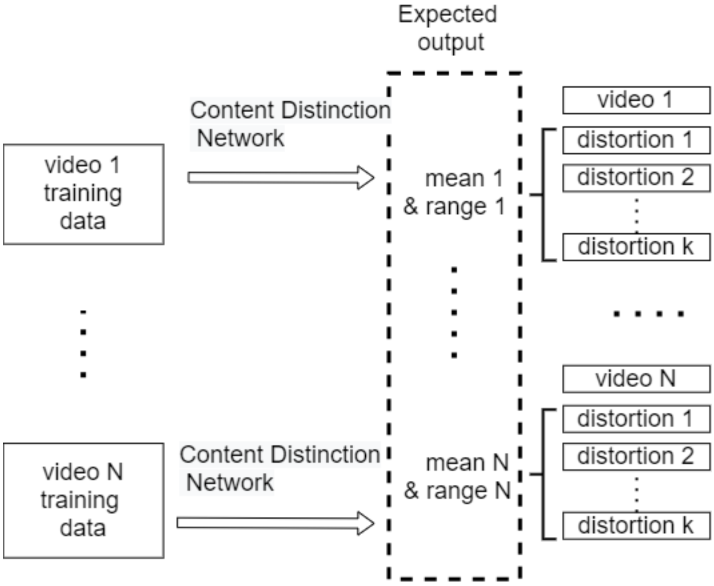


Figure 8. Content distinction network learning procedure.

We used the Adam optimizer with a fixed learning rate of 0.0005 for 20 epochs to train the models. We perform cross-dataset prediction when we generated the content distinction features. This means that we trained the content distinction network on one dataset and predicted features on another dataset. In this case, the contents that we learned and tested were quite different. Our experimental results show that this is a promising direction.

4.2.3. SVR Model Parameter Tuning

The SVR models on the LIVE mobile database and the MCL-V database are separately tuned since one dataset provides MOS and the other provides DMOS. For each dataset, we perform a random train–test split: 80% of the data train the SVR model, and the remaining data are used for testing. We calculated the average PCC and SCC over 100 runs. Each run has a different random seed. This can reduce the effect of some special cases and show off the overall performance. γ and C are the two parameters to be tuned. γ is the inverse of the standard deviation of the Gaussian function. C is used to control the regularization term. There are two common techniques for parameter tuning, namely grid search and random search. Both can help tune the hyper-parameters by trying different values and picking the one with the best performance. However, since random search randomly tries potential values, it can miss the best values. Thus, we used grid search.

4.3. Evaluation Criteria

We used Pearson's (linear) correlation coefficient (PCC) [22] and Spearman's rank correlation coefficient (SCC) [23] to see the correlation of the two RAVA scores with DMOS or MOS. PCC and SCC are the most popular methods for measuring the dependence of two variables X and Y. PCC evaluates the linear relationship while SCC evaluates the monotonic relationship. Mathematically, PCC can be written as

$$\mathrm{PCC}(X,Y) = \frac{\mathrm{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\bar{X})(Y-\bar{Y})]}{\sigma_X \sigma_Y} \tag{11}$$

where $\bar{X}$ and $\bar{Y}$ are the average values of X and Y, respectively; σ_X and σ_Y are the standard deviations.

For SCC, given two samples of size n for both X and Y, R_{X_i} denotes the rank of X_i in the ascending sorted X sample. Similarly, R_{Y_i} denotes the rank of Y_i. When several observations have the same rank, an average rank will be assigned to them. Mathematically, SCC can be written as

$$SCC = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)} \tag{12}$$

where $d_i = R_{X_i} - R_{Y_i}$.

RAVA scores and MOS or DMOS may increase by very different factors. In this case, the PCC value will be affected. However, since SCC is calculated based on the ranks, the value for SCC will remain the same even if they change at different rates. Taking this into consideration, we also reported the SCC scores. In addition, the correlation coefficients are in range of -1–1 since the RAVA scores and the ground truth scores may go in different directions. We only want to measure how strongly the two variables are correlated. Thus, we will only compare the absolute value of PCC and SCC values. If the absolute value is large, then their correlation is high; otherwise, they are less likely to be correlated. The new metric is still considered to be valuable if the negative correlation is strong.

4.4. Experimental Results

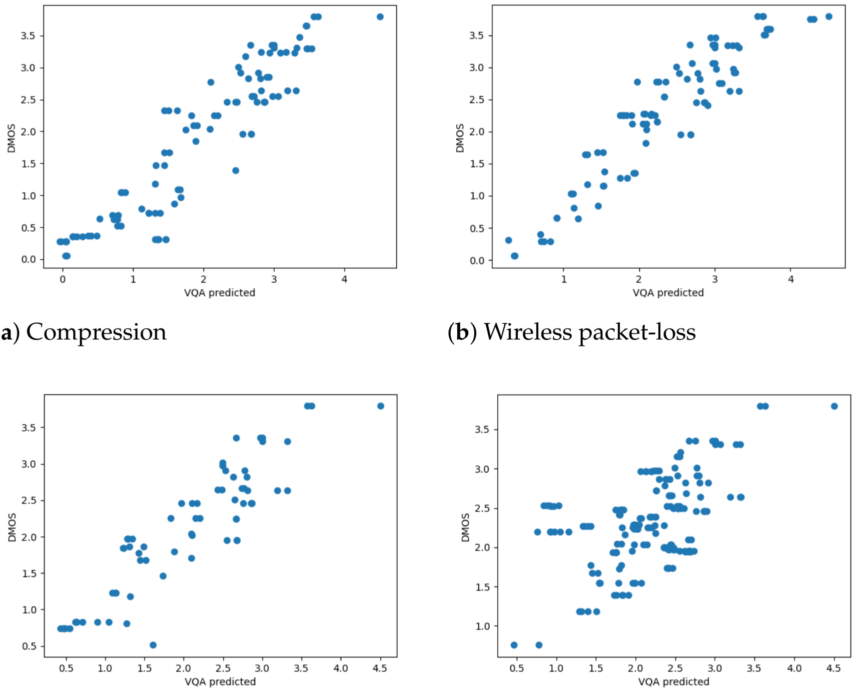
Table 1 shows the values we use for the two parameters γ and C for the LIVE mobile database and the MCL-V database.

Table 1. Values assigned to γ and C for the LIVE mobile database and the MCL-V database.

Method	γ	C
RAVA$_{SSIM}$ (LIVE mobile database—mobile DMOS)	0.025	pow(2,7)
RAVA$_{PSNR}$ (LIVE mobile database—mobile DMOS)	1	pow(2,10)
RAVA$_{SSIM}$ (LIVE mobile database—tablet DMOS)	0.025	pow(2,7)
RAVA$_{PSNR}$ (LIVE mobile database—tablet DMOS)	0.6	pow(2,10)
RAVA$_{SSIM}$ (MCL-V database)	3	pow(10,2)
RAVA$_{PSNR}$ (MCL-V database)	2	pow(2,9)

4.4.1. The LIVE Mobile Database (Mobile DMOS)

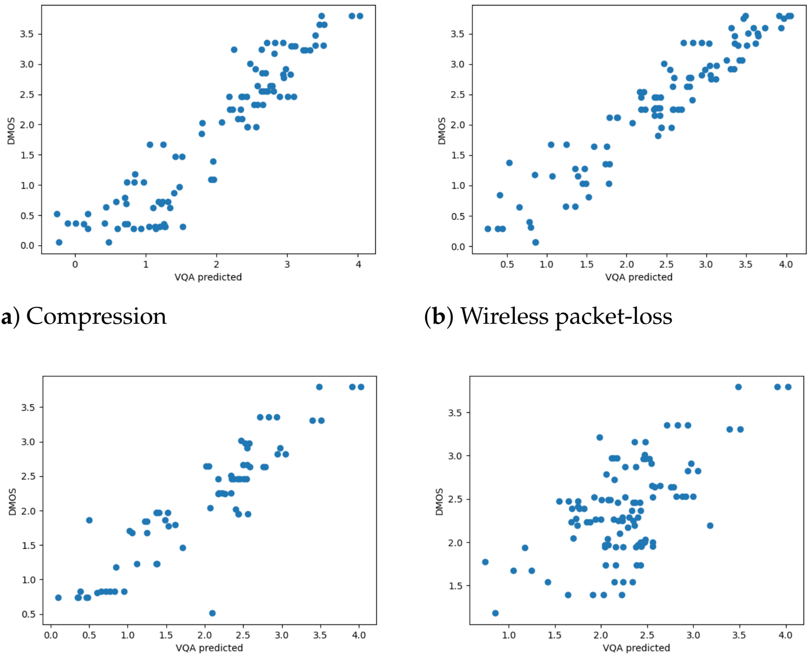
We first analyzed the SCC and PCC on the mobile DMOS for the LIVE Mobile Video Quality Assessment (VQA) Database. To better visualize the results for different distortion types, we drew the scatter plots of the RAVA scores vs. DMOS in Figures 9 and 10. The plots aggregate the test results for 10 runs. As mentioned in [17], how humans rate videos with freeze-frame distortions is still unclear, as we only draw plots for the other four distortion types. The overall performance of the two RAVA methods are drawn in Figure 11a for comparison. The two methods have different ranges. Thus, to visualize using the same scale, we normalized the DMOS and RAVA scores before drawing the plot.



(**a**) Compression (**b**) Wireless packet-loss

(**c**) Rate-adapted (**d**) Temporal dynamics

Figure 9. RAVA$_{SSIM}$ vs. DMOS for different distortion types (mobile).



(**a**) Compression (**b**) Wireless packet-loss

(**c**) Rate-adapted (**d**) Temporal dynamics

Figure 10. RAVA$_{PSNR}$ vs. DMOS for different distortion types. (mobile).

From all the plots, we can see that the performance of the two proposed FR VQA methods RAVA$_{SSIM}$ and RAVA$_{PSNR}$ are very similar. They performed well on videos distorted with compression, wireless packet-loss, and rate adaptation. Furthermore, we compared our PCC and SCC results with eight commonly used video quality assessment methods: PSNR [9]; VQM [32]; MOVIE [46]; MS-SSIM [47]; SS-SSIM [8]; VIF [25]; VSNR [48]; and NQM [49]. The quantitative comparisons are shown in Tables 2 and 3. The results for the two RAVA methods are averaged over 100 runs. A bold value in a column represents the highest value in that column. Note that Co means compression; wl means wireless channel packet loss; Ra means rate adaptation; and Td means temporal dynamics.

As shown in the tables, RAVA$_{SSIM}$ outperformed all the listed VQA methods for both SCC and PCC. Moreover, it achieved the best performance in all the distortion types except for temporal dynamics. Compared to the existing SS-SSIM method, the overall correlation was increased by 0.169 and 0.195 for SCC and PCC, respectively. Furthermore, our RAVA$_{SSIM}$ was improved by 0.076 (SCC) and 0.151 (PCC) compared to MS-SSIM. RAVA$_{PSNR}$ also performed well, though its overall performance ranked second. The correlation was increased by 0.134 (SCC) and 0.152 (PCC) compared to the existing PSNR method.

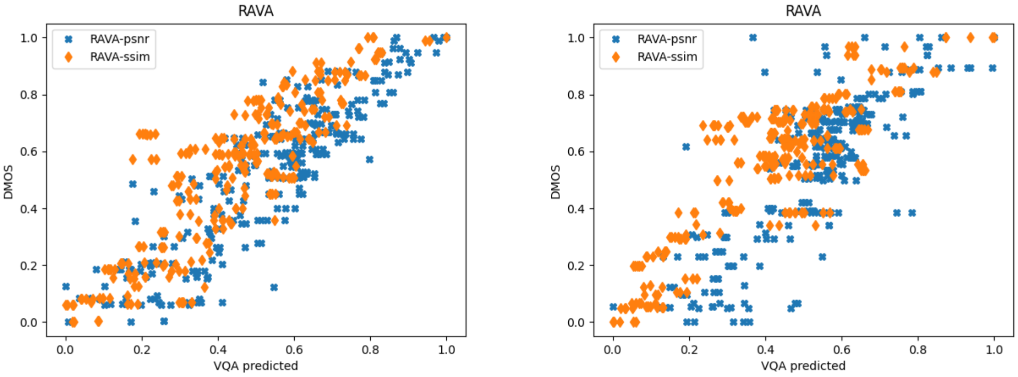
Note that all the existing VQA methods do not perform well for the temporal dynamics distortion. The best performance is 0.386 (VQM) in SCC and 0.427 (VSNR) in PCC. The two proposed RAVA methods obtained higher scores compared to all the existing methods in videos distorted by this type, as all the SCC values were above 0.5 and all the PCC values were above 0.6.

Table 2. Comparison of SCC (Spearman correlation coefficient)—mobile.

Distortion	**Co**	**Wl**	**Ra**	**Td**	**All**
PSNR	0.819	0.793	0.598	0.372	0.678
VQM	0.772	0.776	0.648	0.386	0.695
MOVIE	0.774	0.651	0.720	0.158	0.642
MS-SSIM	0.804	0.813	0.738	0.397	0.743
SS-SSIM	0.709	0.725	0.630	0.343	0.650
VIF	0.861	0.874	0.639	0.124	0.744
VSNR	0.874	0.856	0.674	0.317	0.752
NQM	0.850	0.899	0.678	0.238	0.749
RAVA$_{PSNR}$	0.859	0.873	0.835	**0.554**	0.812
RAVA$_{SSIM}$	**0.902**	**0.912**	**0.844**	0.507	**0.819**

Table 3. Comparison of PCC (Pearson correlation coefficient)—mobile.

Distortion	**Co**	**Wl**	**Ra**	**Td**	**All**
PSNR	0.784	0.762	0.536	0.417	0.691
VQM	0.782	0.791	0.591	0.407	0.702
MOVIE	0.810	0.727	0.681	0.244	0.716
MS-SSIM	0.766	0.771	0.709	0.407	0.708
SS-SSIM	0.748	0.731	0.612	0.392	0.664
VIF	0.883	0.898	0.664	0.105	0.787
VSNR	0.849	0.849	0.658	0.427	0.759
NQM	0.832	0.874	0.677	0.365	0.762
RAVA$_{PSNR}$	0.908	0.905	0.887	**0.659**	0.843
RAVA$_{SSIM}$	**0.929**	**0.941**	**0.894**	0.603	**0.859**



(**a**) Overall performance on mobile DMOS (**b**) Overall performance on tablet DMOS

Figure 11. The overall performance of the RAVA models on the Live Mobile Database (mobile and tablet).

4.4.2. The LIVE Mobile Database (Tablet DMOS)

Similar to the above experiments, we obtained the SCC and PCC for the tablet DMOS considering the average of 100 runs. The results are shown in Tables 4 and 5. There is also a plot comparing the overall performance of the two RAVA methods for 10 runs, as shown in Figure 11b.

Table 4. Comparison of SCC (Spearman correlation coefficients)—tablet.

Distortion	Co	Wl	Ra	Td	All
PSNR	0.791	0.756	0.446	0.098	0.589
VQM	0.632	0.669	0.436	0.051	0.555
MOVIE	0.774	0.845	**0.771**	0.065	0.679
MS-SSIM	0.660	0.645	0.482	0.140	0.568
SS-SSIM	0.495	0.561	0.368	0.077	0.430
VIF	0.892	0.862	0.671	0.070	0.726
VSNR	0.771	0.705	0.443	0.047	0.593
NQM	0.841	0.808	0.464	0.079	0.661
RAVA$_{\mathbf{PSNR}}$	0.777	0.810	0.613	**0.223**	**0.767**
RAVA$_{\mathbf{SSIM}}$	**0.923**	**0.879**	0.672	−0.084	0.763

When predicting the DMOS on tablet devices, most existing methods do not perform well, especially for videos distorted by temporal dynamics. However, the performance of the RAVA methods do not degrade very much. RAVA$_{SSIM}$ performs well on videos distorted by compression and wireless packet-loss. RAVA$_{PSNR}$ outperforms all the other methods in videos distorted by temporal dynamics. If we look at the overall performance, the two RAVA methods have the top two scores in both SCC and PCC.

4.4.3. The MCL-V Database

The experimental results on the MCL-V database over 100 runs are shown in Table 6. We compared our result with the following VQA methods: PSNR [9]; SS-SSIM [8]; MS-SSIM [47]; VIF [25]; VADM [50]; and FSIM [51].

Table 5. Comparison of PCC (Pearson correlation coefficient)—tablet.

Distortion	Co	Wl	Ra	Td	All
PSNR	0.771	0.732	0.437	0.252	0.635
VQM	0.643	0.735	0.490	0.273	0.735
MOVIE	0.828	0.877	**0.802**	0.071	0.783
MS-SSIM	0.702	0.706	0.564	0.213	0.621
SS-SSIM	0.586	0.590	0.422	0.081	0.489
VIF	0.851	0.854	0.594	0.048	0.764
VSNR	0.775	0.731	0.508	0.220	0.644
NQM	0.812	0.830	0.412	0.120	0.718
RAVA$_{\mathbf{PSNR}}$	0.843	0.869	0.642	**0.279**	**0.848**
RAVA$_{\mathbf{SSIM}}$	**0.957**	**0.921**	0.700	−0.207	0.847

As suggested in [18] and [52], we applied the following non-linear regression on the VQA scores before calculating the PCC and SCC scores for all the VQA metrics when evaluating on this dataset:

$$y = \beta_1 \cdot \left(0.5 - \frac{1}{1 + e^{\beta_2(x-\beta_3)}}\right) + \beta_4 \cdot x + \beta_5 \quad (13)$$

β_1–β_5 are the five fitting parameters and x is the objective VQA score.

Table 6. Comparison of PCC and SCC on the MCL-V database.

	PCC			SCC		
Distortion	**Co**	**Scaling**	**All**	**Co**	**Scaling**	**All**
PSNR	0.471	0.463	0.472	0.422	0.493	0.464
SS-SSIM	0.650	0.635	0.650	0.633	0.649	0.648
MS-SSIM	0.617	0.609	0.621	0.609	0.630	0.623
VIF	0.667	0.636	0.660	0.609	0.661	0.655
VADM	0.747	0.728	0.742	0.735	0.741	0.755
FSIM	0.770	0.722	0.750	**0.775**	0.702	0.752
RAVA$_{\mathbf{SSIM}}$	**0.783**	0.709	0.749	0.763	0.696	0.744
RAVA$_{\mathbf{PSNR}}$	0.767	**0.750**	**0.765**	0.750	**0.742**	**0.759**

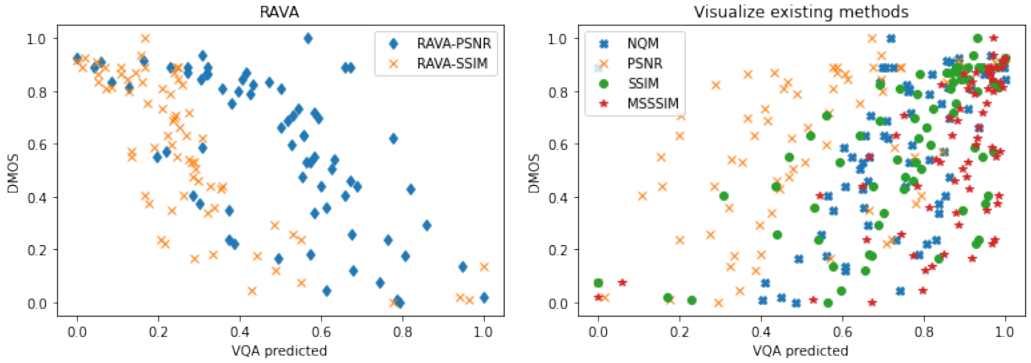
Note: Co represents distortion type compression.

In Table 6, RAVA$_{PSNR}$ has the best performance. It has the best overall performance for videos with distortion scaling in both PCC and SCC. RAVA$_{SSIM}$ also performs well. It is not as good as RAVA$_{PSNR}$ in terms of dealing with videos distorted by scaling, but exceeds it in predicting videos with compression. Both of our methods have better performance than existing PSNR, SS-SSIM, and MS-SSIM methods. RAVA$_{PSNR}$ outperforms PSNR by 0.293 in PCC and 0.331 in SCC. RAVA$_{SSIM}$ improves SS-SSIM by 0.099 in PCC and 0.096 in SCC. In addition, it is better than MS-SSIM by 0.128 in PCC and 0.121 in SCC.

The two RAVA methods perform better in the LIVE Mobile database than in the MCL-V database. This is due to the limitations of the training data for MCL-V's content distinction network. We trained the model with the LIVE Mobile database, but that dataset only contains real-life video clips. Thus, the trained model is not good at predicting the cartoons and animations which are in the MCL-V database.

4.4.4. Cross-Library Experiment on the Netflix Public Dataset

We also conducted cross-library validation on the Netflix Public Dataset [19] using the content distinction network and SVR model directly pre-trained on the LIVE Mobile Video Quality Assessment (VQA) database [17]. Figure 12 shows the comparison of the performance of the two pre-trained RAVA models and some existing methods, namely PSNR [9], SS-SSIM [8], MS-SSIM [47], and NQM [49]. Clearly, the dots for the existing methods are more discrete while the dots for the two RAVA methods are more concentrated. Note, as discussed in Section 4.1, that the criteria for collecting the DMOS on the Netflix Public Dataset [19] are different from the criteria for collecting DMOS on the LIVE Mobile Video Quality Assessment (VQA) database [17]. Thus, in our pre-trained model, a lower score is better while for ground truth, higher is better. In Figure 12a, the predicted RAVA scores and the ground truth DMOS are negatively correlated, but we can still see a strong correlation. This can be reaffirmed in Table 7, since the RAVA$_{PSNR}$ obtains a higher PCC and SCC than PSNR, while RAVA$_{SSIM}$ performs much better than SS-SSIM and MS-SSIM.



(**a**) Pre-trained RAVA models (**b**) Existing methods

Figure 12. The performance of the pre-trained RAVA models and the existing methods on the Netflix Public Dataset.

Table 7. Comparison of PCC and SCC on the Netflix Public Dataset.

Methods	**PCC**	**SCC**
PSNR	0.536	0.551
SS-SSIM	0.635	0.621
MS-SSIM	0.585	0.631
NQM	0.491	0.579
RAVA$_{PSNR}$	−0.635	−0.625
RAVA$_{SSIM}$	−**0.771**	−**0.781**

5. Conclusions

We introduced a new video quality evaluation approach that integrated various image quality assessment methods, namely region-based detection, temporal weights from optical flow, and content distinction features. Our RAVA technique was applied to extend two full-reference IQA metrics. We first separated foreground and background regions for all the video frames. Then, we integrated the motion features into the weights while designing the VQA metrics. The region weights were defined as the percentage of the average magnitudes of the optical flows for those regions out of all the regions. The foreground feature was the weighted average of the foreground IQA scores, and the background feature was the simple average of background IQA scores. Furthermore, a content distinction network was added to generalize the RAVA scores for videos with various types of content. All the features were passed to an SVR model to predict the final VQA score. We tested on two different datasets to validate the RAVA technique. The LIVE Mobile VQA database and the MCL-V database are widely used VQA datasets, so we used them to compare the performance of the RAVA methods with existing methods. By analyzing the correlation of the RAVA scores and the DMOS (or MOS) provided by the datasets, we noticed that RAVA$_{PSNR}$ and RAVA$_{SSIM}$ performed very well. Furthermore, the results produced by RAVA$_{PSNR}$ were better than those of the PSNR of existing video quality assessment methods. RAVA$_{SSIM}$ also performed better than SS-SSIM and MS-SSIM.

In summary, we believe that the RAVA approach has practical significance. It can extend IQA methods to VQA methods, and we expect it to be widely applicable for video quality assessment in the future.

Author Contributions: Formal analysis, X.W.; funding acquisition, Z.Z.; methodology, X.W.; supervision, I.C. and A.B.; validation, X.W.; writing—original draft, X.W.; writing—review and editing, I.C. and A.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by UAH Joint Innovation Center, and NSERC, Canada.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analysed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

RAVA	region-based average video quality assessment
NR-IQA	no-reference image quality assessment
FR	full reference
NIMA	neural image assessment
NR	no-reference
PEVQ	perceptual evaluation of video quality
RR	reduced reference
VQuad-HD	objective perceptual multimedia video quality measurement of HDTV
VQA	video quality assessment
PEXQ	perceptual quality by OPTICOM, a software name
IQA	image quality assessment
ANN	artificial neural network
SSIM	structural similarity
PSD	power spectral density
VSSIM	video structural similarity
PVQA	perceptual video quality assessment
HVS	human visual system
VQA	video quality metric
PSNR	peak signal-to-noise ratio
ANSI	American National Standards Institute
PSNR-HVS	PSNR based on HVS
ITU	International Telecommunication Union
PSNR-HVS-M	PSNR based on between-coefficient contrast masking
VSFA	quality assessment of in-the-wild videos
fps	frames per second
CRF	constant rate factor
PCC	Pearson's (linear) correlation coefficient
SCC	Spearman's rank correlation coefficient

References

1. Cordeiro, P.J.; Assunção, P. Distributed Coding/Decoding Complexity in Video Sensor Networks. *Sensors* **2012**, *12*, 2693–2709. [CrossRef]
2. Kawai, T. Video Slice: Image Compression and Transmission for Agricultural Systems. *Sensors* **2021**, *21*, 3698. [CrossRef] [PubMed]

3. Lee, S.Y.; Rhee, C.E. Motion Estimation-Assisted Denoising for an Efficient Combination with an HEVC Encoder. *Sensors* **2019**, *19*, 895. [CrossRef]
4. Cheng, I.; Basu, A. Perceptually Optimized 3-D Transmission Over Wireless Networks. *IEEE Trans. Multimed.* **2007**, *9*, 386–396. [CrossRef]
5. Yixin, P.; Irene, C.; Basu, A. Quality metric for approximating subjective evaluation of 3-D objects. *IEEE Trans. Multimed.* **2005**, *7*, 269–279. [CrossRef]
6. Guo, J.; Vidal, V.; Cheng, I.; Basu, A.; Baskurt, A.; Lavoue, G. Subjective and Objective Visual Quality Assessment of Textured 3D Meshes. *ACM Trans. Appl. Percept.* **2016**, *14*, 1–20. [CrossRef]
7. Lavoué, G.; Cheng, I.; Basu, A. Perceptual Quality Metrics for 3D Meshes: Towards an Optimal Multi-attribute Computational Model. In Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, Manchester, UK, 13–16 October 2013; pp. 3271–3276.
8. Zhou, W.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612.
9. Avcibaş, I.; Sankur, B.; Sayood, K. Statistical evaluation of image quality measures. *J. Electron. Imaging* **2002**, *11*, 206–223. [CrossRef]
10. Yang, J.; Zhao, Y.; Liu, J.; Jiang, B.; Meng, Q.; Lu, W.; Gao, X. No Reference Quality Assessment for Screen Content Images Using Stacked Autoencoders in Pictorial and Textual Regions. *IEEE Trans. Cybern.* **2020**. [CrossRef]
11. Kottayil, N.K.; Valenzise, G.; Dufaux, F.; Cheng, I. Blind Quality Estimation by Disentangling Perceptual and Noisy Features in High Dynamic Range Images. *IEEE Trans. Image Process.* **2018**, *27*, 1512–1525. [CrossRef]
12. Talebi, H.; Milanfar, P. NIMA: Neural Image Assessment. *IEEE Trans. Image Process.* **2018**, *27*, 3998–4011. [CrossRef] [PubMed]
13. Gupta, P.; Srivastava, P.; Bhardwaj, S.; Bhateja, V. A modified PSNR metric based on HVS for quality assessment of color images. In Proceedings of the 2011 International Conference on Communication and Industrial Application, Beijing, China, 14–16 October 2011. [CrossRef]
14. Ponomarenko, N.; Silvestri, F.; Egiazarian, K.; Carli, M.; Astola, J.; Lukin, V. On between-coefficient contrast masking of DCT basis functions. In Proceedings of the 3rd Int Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, AZ, USA, 25–26 January 2007.
15. Chatterjee, M.; Cao, J.; Kothapalli, K.; Rajsbaum, S. Distributed Computing and Networking. In Proceedings of the 15th International Conference, ICDCN 2014, Coimbatore, India, 4–7 January 2014; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germay, 2014; p. 427.
16. Drucker, H.; C, C.; Kaufman, L.; Smola, A.; Vapnik, V. Support Vector Regression Machines. *Adv. Neural Inf. Process. Syst.* **2003**, *9*, 155–161.
17. Moorthy, A.K.; Choi, L.K.; Bovik, A.C.; de Veciana, G. Video Quality Assessment on Mobile Devices: Subjective, Behavioral and Objective Studies. *IEEE J. Sel. Top. Signal Process.* **2012**, *6*, 652–671. [CrossRef]
18. Lin, J.Y.; Song, R.; Wu, C.H.; Liu, T.; Wang, H.; Kuo, C.C.J. MCL-V: A streaming video quality assessment database. *J. Vis. Commun. Image Represent.* **2015**, *30*, 1–9. [CrossRef]
19. Li, Z.; Aaron, A.; Katsavounidis, I.; Moorthy, A.; Manohara, M. *Toward A Practical Perceptual Video Quality Metric*; Netflix TechBlog, 2016.
20. García, B.; Gortázar, F.; Gallego, M.; Hines, A. Assessment of QoE for Video and Audio in WebRTC Applications Using Full-Reference Models. *Electronics* **2020**, *9*, 462. [CrossRef]
21. Sheikh, H.; Sabir, M.; Bovik, A. A Statistical Evaluation of Recent Full Reference Image Quality Assessment Algorithms. *IEEE Trans. Image Process.* **2006**, *15*, 3440–3451. [CrossRef] [PubMed]
22. Wang, J. Pearson Correlation Coefficient. In *Encyclopedia of Systems Biology*; Dubitzky, W., Wolkenhauer, O., Cho, K.H., Yokota, H., Eds.; Springer: New York, NY, USA, 2013; p. 1671. [CrossRef]
23. Spearman Rank Correlation Coefficient. In *The Concise Encyclopedia of Statistics*; Springer: New York, NY, USA, 2008; pp. 502–505. [CrossRef]
24. Lin, M.; Chenwei, D.; Ngan, K.N.; Weisi, L. Recent advances and challenges of visual signal quality assessment. *China Commun.* **2013**, *10*, 62–78. [CrossRef]
25. Sheikh, H.R.; Bovik, A.C. Image information and visual quality. *IEEE Trans. Image Process.* **2006**, *15*, 430–444. [CrossRef]
26. Li, S.; Zhang, F.; Ma, L.; Ngan, K.N. Image Quality Assessment by Separately Evaluating Detail Losses and Additive Impairments. *IEEE Trans. Multimed.* **2011**, *13*, 935–949. [CrossRef]
27. Li, Z.; Bampis, C.; Novak, J.; Aaron, A.; Swanson, K.; Moorthy, A. *VMAF: The Journey Continues*; Netflix TechBlog, 2018.
28. Liu, L.; Wang, T.; Huang, H.; Bovik, A.C. Video quality assessment using space–time slice mappings. *Signal Process. Image Commun.* **2020**, *82*, 115749. [CrossRef]
29. Ngo, C.W.; Pong, T.C.; Zhang, H.J. Motion analysis and segmentation through spatio-temporal slices processing. *IEEE Trans. Image Process.* **2003**, *12*, 341–355. [CrossRef]
30. Aabed, M.A.; Kwon, G.; AlRegib, G. Power of tempospatially unified spectral density for perceptual video quality assessment. In Proceedings of the 2017 IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, China, 10–14 July 2017; pp. 1476–1481.

31. Soong, H.; Lau, P. Video quality assessment: A review of full-referenced, reduced-referenced and no-referenced methods. In Proceedings of the 2017 IEEE 13th International Colloquium on Signal Processing its Applications (CSPA), Penang, Malaysia, 10–12 March 2017; pp. 232–237.
32. Pinson, M.H.; Wolf, S. A new standardized method for objectively measuring video quality. *IEEE Trans. Broadcast.* **2004**, *50*, 312–322. [CrossRef]
33. Li, D.; Jiang, T.; Jiang, M. Quality Assessment of In the-Wild Videos. In Proceedings of the 27th ACM International Conference on Multimedia (MM '19), Nice, France, 21–25 October 2019.
34. Zadtootaghaj, S.; Barman, N.; Ramachandra Rao, R.R.; Rao, R.; Göring, S.; Martini, M.; Raake, A.; Möller, S. DEMI: Deep Video Quality Estimation Model using Perceptual Video Quality Dimensions. In Proceedings of the 2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP), Tampere, Finland, 21–24 September 2020. [CrossRef]
35. Horn, B.K.; Schunck, B.G. Determining Optical Flow. In *Techniques and Applications of Image Understanding*; Pearson, J.J., Ed.; International Society for Optics and Photonics; SPIE: Washington, WA, USA, 1981; Volume 0281, pp. 319–331. [CrossRef]
36. Burton, A.; Radford, J. *Thinking in Perspective: Critical Essays in the Study of Thought Processes*; Methuen: North Yorkshire, UK, 1978.
37. Warren, D.; Strelow, E.R. *Electronic Spatial Sensing for the Blind: Contributions from Perception, Rehabilitation, and Computer Vision*; Dordrecht: Boston, MA, USA, 1985.
38. Sim, K.; Yang, J.; Lu, W.; Gao, X. MaD-DLS: Mean and Deviation of Deep and Local Similarity for Image Quality Assessment. *IEEE Trans. Multimed.* **2020**. [CrossRef]
39. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
40. Korhonen, J. Two-Level Approach for No-Reference Consumer Video Quality Assessment. *IEEE Trans. Image Process.* **2019**, *28*, 5923–5938. [CrossRef]
41. Kim, J.; Lee, J.; Kim, T. AdaMM: Adaptive Object Movement and Motion Tracking in Hierarchical Edge Computing System. *Sensors* **2021**, *21*, 4089. [CrossRef]
42. Bae, D.H.; Kim, J.W.; Heo, J.P. Content-Aware Focal Plane Selection and Proposals for Object Tracking on Plenoptic Image Sequences. *Sensors* **2019**, *19*, 48. [CrossRef] [PubMed]
43. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
44. Bradski, G. The OpenCV Library. *Dr. Dobb's J. Softw. Tools* **2000**. Available online: https://opencv.org/ (accessed on 5 July 2020).
45. Ilg, E.; Mayer, N.; Saikia, T.; Keuper, M.; Dosovitskiy, A.; Brox, T. FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1647–1655.
46. Seshadrinathan, K.; Bovik, A.C. Motion Tuned Spatio-Temporal Quality Assessment of Natural Videos. *IEEE Trans. Image Process.* **2010**, *19*, 335–350. [CrossRef] [PubMed]
47. Wang, Z.; Simoncelli, E.P.; Bovik, A.C. Multiscale structural similarity for image quality assessment. In Proceedings of the Thrity-Seventh Asilomar Conference on Signals, Systems Computers, Pacific Grove, CA, USA, 9–12 November 2003; Volume 2, pp. 1398–1402. [CrossRef]
48. Chandler, D.M.; Hemami, S.S. VSNR: A Wavelet-Based Visual Signal-to-Noise Ratio for Natural Images. *IEEE Trans. Image Process.* **2007**, *16*, 2284–2298. [CrossRef] [PubMed]
49. Damera-Venkata, N.; Kite, T.D.; Geisler, W.S.; Evans, B.L.; Bovik, A.C. Image quality assessment based on a degradation model. *IEEE Trans. Image Process.* **2000**, *9*, 636–650. [CrossRef] [PubMed]
50. Li, S.; Ma, L.; Ngan, K.N. Full-Reference Video Quality Assessment by Decoupling Detail Losses and Additive Impairments. *IEEE Trans. Circuits Syst. Video Technol.* **2012**, *22*, 1100–1112. [CrossRef]
51. Zhang, L.; Zhang, L.; Mou, X.; Zhang, D. FSIM: A Feature Similarity Index for Image Quality Assessment. *IEEE Trans. Image Process.* **2011**, *20*, 2378–2386. [CrossRef] [PubMed]
52. Chikkerur, S.; Sundaram, V.; Reisslein, M.; Karam, L.J. Objective Video Quality Assessment Methods: A Classification, Review, and Performance Comparison. *IEEE Trans. Broadcast.* **2011**, *57*, 165–182. [CrossRef]