IET Image Processing

Research Article

IET Journals
The Institution of Engineering and Technology

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

Feature-based No-Reference Video Quality Assessment using Extra Trees

Hatef Otroshi-Shahreza[1,2], *Arash Amini*[1*], *Hamid Behroozi*[1]

[1] *Electrical Engineering Department, Sharif University of Technology, Tehran, Iran*
[2] *School of Engineering, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland*
* *E-mail: aamini@sharif.edu*

Abstract: With the emerge of social networks and improvements in the Internet speed, the video data has become an ever-increasing portion of the global internet traffic. Besides the content, the quality of a video sequence is an important issue at the user end which is often affected by various factors such as compression. Therefore, monitoring the quality is crucial for the video content and service providers. A simple monitoring approach is to compare the raw video content (uncompressed) with the received data at the receiver. In most practical scenarios, however, the reference video sequence is not available. Consequently, it is desirable to have a general reference-less method for assessing the perceived quality of any given video sequence. In this paper, we propose a no-reference video quality assessment technique based on video features. In particular, we consider a long list of video features (21 sets of features, each consisting of 1 to 216 features) and examine all possible combinations ($2^{21}-1$) for training an Extra Trees regressor. This choice of the regressor is wisely selected and is observed to perform better than other common regressors. Our results reveal that the top 20 performing feature subsets all outperform the existing feature-based assessment methods in terms of the Pearson linear correlation coefficient (PLCC) or the Spearman rank order correlation coefficient (SROCC). Specially, the best performing regressor achieves PLCC $= 0.786$ on the test data over the KonVid-1k dataset. We believe that the results of our comprehensive comparison could be potentially useful for other feature-based video-related problems. The source codes of our implementations are publicly available.

1 Introduction

With the excessive growth of video content and video applications, the quality assessment of video sequences has become of great importance. According to [1], video content will account for 82% of global internet traffic in 2022 from which 17% is dedicated to live streams. Depending on the involved processes, the quality of video sequences might be different. Most notably, optical distortions imposed by the camera at the time of recording, distortions caused by the data compression, and the data loss within the communication channel affect the overall quality. In addition, at the receiving end, the original/reference video file is rarely available. This eliminates the possibility of using a full-reference video quality assessment (FR-VQA) algorithms. In this case, the class of no-reference video quality assessment (NR-VQA) algorithms is the only option. These methods predict the quality of a video file according to certain quality measures (e.g., mean opinion score, or shortly MOS) solely based on the available data. The prediction procedure oftentimes relies on the statistical features of the video files. Some of the known no-reference assessment methods are V-BLINDS [2], VIIDEO [3], V-CORNIA [4], and STFC [5].

For measuring the performance of an assessment method, it is common to check the outcome of the method on a database which is already rated by the users. The LIVE video quality assessment database [6] has been a frequently-used database for evaluating the performance of NR-VQA methods in the past. This database contains 10 video files each accompanied with 15 synthetically distorted versions and the ratings of 38 users. Methods such a [2, 3, 7] have reported very good performances over this dataset. Aside from the small size of this dataset, different distortion types are separately applied to the source files, which is somehow different from the practical scenarios where a video file usually suffers from multiple distortion types with various degrees.

Recently, the KonVid-1k database with 1200 video files is made available [8] which contains real distorted video files. With respect to the size and distortion types of the files, this database is expected to provide a more realistic test-bed for NR-VQA methods. It is counter-intuitive to mention that some of the successful methods on the LIVE Video Quality database perform poorly on the KonVid-1k database [8].

Most NR-VQA techniques rely on learning the quality measure from a training dataset. For the learning process, one can either use the full video content or its reduced version in form of a number of features. The latter approach, besides the advantage of less computational cost, can be applied to arbitrary video frame sizes (if the features are properly chosen). In this paper, we first cover a list of 21 feature sets and consider all $2^{21}-1$ combinations of these feature sets to train an Extra Trees regressor; we should highlight that the Support Vector Regressor (SVR) has been the dominant regressor in the existing techniques and the use of Extra Trees regressor in this work is its first appearance in the VQA-related context. Based on the achieved performance of the trained regressors, we discuss how the features can be efficiently selected to achieve good results while avoiding high computational costs. By dividing the features into two categories of frame-level (within one video frame) and spatiotemporal (considering the frames over time), we find out that the statistics of the luma channel is the most informative frame-level feature (indeed, the most informative among all features). The results also assign significant weight to frame-level HOSA [9] and CORNIA scores as features. Furthermore, the most relevant spatiotemporal feature set is generated by the V-BLINDS method. We also define certain angular-type spatiotemporal features in this paper that enhance the V-BLINDS features.

1.1 Contributions

In this paper, we present a machine-learning based study on the problem of NR-VQA. To this end, we provide a list of video features employed in the existing NR-VQA methods, as well as some new features defined in this work. Next, we evaluate the performance of NR-VQA models formed by combining different features. This study consists of evaluating the effect of both the regressor

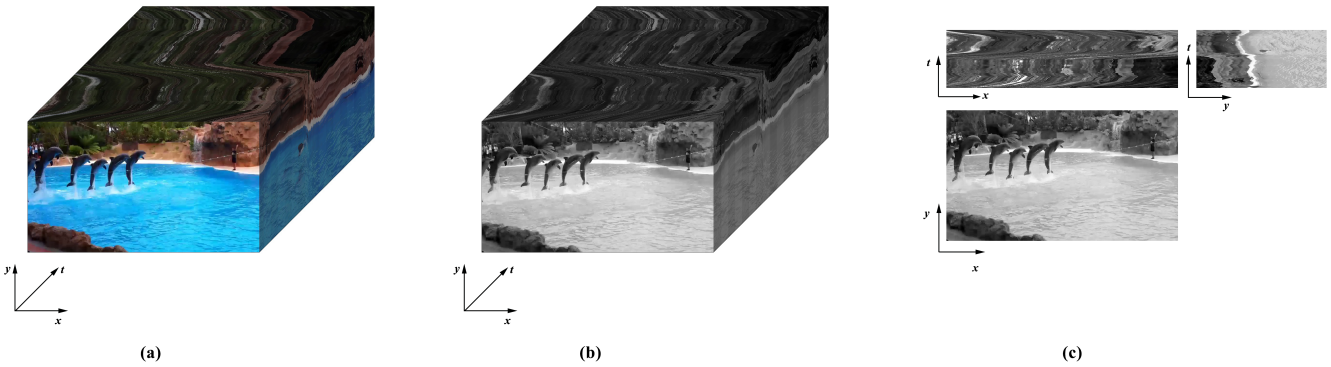


Fig. 1: A video file can be viewed in a 3D spatiotemporal space, where two of the axes indicates the spatial dimensions (x and y) and the third axis represents the time (t). (a) representation of RGB color space in spatiotemporal space. (b) representation of luminance channels in spatiotemporal space. (c) spatial and spatiotemporal slices of the video.

type and the feature combinations. While the choice of the regressor is often ignored in the NR-VQA literature, we observe that it has a significant impact on the performance. To identify important features in predicting the video quality, we perform several experiments and comparisons. Specifically, we find that a success model needs to have a balanced combination of frame-level and spatiotemporal features. We further propose new spatiotemporal features that could be efficiently combined with frame-level features. In brief, the contributions of this paper are

- We explore the effect of the regressor on the outcome of the NR-VQA on a small-size problem. The best performing regressor (Extra Trees regressor) is then used for the remaining large-size problems.
- We introduce new spatiotemporal features by considering the video content as a 3D cube; some of these features later appear in high-performance NR-VQA feature combinations.
- We study the performance of all possible subsets of the features to identify the significance of each feature in the overall performance of a NR-VQA method; in particular, we report the performance of the pairs of the feature sets to introduce the best matching features.
- We investigate the importance of each feature group through a decision tree based regressor. The feature groups achieving the highest scores in this experiment are interpreted as most relevant feature groups in predicting video quality. Interestingly, the results match that of the comprehensive feature subset experiment (feature groups appearing in the top performing subsets).
- We introduce several feature subsets with small size that perform very well; the restriction on the size of the feature subsets helps us to cut the unnecessary computational cost.
- The top 20 performing feature subsets have competitive performance with the state-of-the-art feature-based NR-VQA (even deep-learning approaches)

1.2 Structure of the paper

We first review the related works in Section 2. Next, we describe the proposed method in Section 3 and provide experimental results in Section 4. Finally, the paper is concluded in section 5.

2 Related Works

In most of the previous studies of NR-VQA, the quality measure of Mean Opinion Score (MOS) is predicted by applying the SVR method on a set of statistical features extracted from the video data. For instance, [2] uses Natural Video Statistics (NVS), frame-level Natural Scene Statistics (NSS) (i.e., features introduced in [10]), and a number of defined motion-related features.

In [3], the statistics of frame differences directly determine the quality measure (without regression). Although remarkable performance is reported on the LIVE VQA database, the performance of this method on the KonVid-1k database is disappointing [11].

A video file can be considered as the concatenation of multiple images. Based on this fact, the image quality assessment technique in [4] is applied in [12] to get the video quality metric by aggregation of frame-by-frame quality metrics.

A set of five features (i.e., average frame blurriness, average frame contrast, average frame colorfulness, average spatial information in frames, and average temporal information) is introduced in [11] to predict the MOS via SVR. Since most of the feature extractions in [11] are performed frame-by-frame, in order to consider temporal variations, a number of spatiotemporal features are added to the previous five features in [5]. Similarly, two types of regressions (namely, a SVR and a neural network) are trained in [13] by using a set of features from spatiotemporal slices. The NARVAL model in [14], also combines frame-level and video-level features. A modified version of this model called BB-NARVAL takes advantage of natural scene statistics reported in [2, 15]. In [16], the frame-level features are divided into two computational categories of low-complexity and high-complexity; the low-complexity features are extracted from all the frames, while the high-complexity features are extracted from only a subset of the frames. Low-complexity and high-complexity features are concatenated into a single feature vector for training a SVR and a random forest regression (named TLVQM-SVR and TLVQM-RFR, respectively). In [17], several feature-based NR-VQA models are employed to extract statistical video features which are then combined to train a quality prediction SVR.

The deep-learning approaches for video quality assessment are quite recent due to their high computational requirements. In [18, 19], a 2D convolutional neural network (CNN) with recurrent neural networks are proposed. A 3D-CNN with recurrent units is also presented in [20]. Combining the feature-based techniques and deep-learning approaches, [21] uses a pooling of the frame-level feature maps of Inception-V3[22] and Inception-ResNet-V2[23] to train a SVR for each model. Several slices of the spatiotemporal cube are used to train a CNN in [24] which generates the overall score by averaging over the patch scores. In [25], a CNN is employed to extract frame-level features which are then, fed to a gated recurrent unit (GRU). Similarly, in [26, 27], hand-crafted frame-level features are fed to long short-term memory (LSTM) unites. In [28], a deep neural network is proposed which consists of a quality degradation learning sub-network (with convolutional layers) and a motion effect modeling sub-network (with recurrent layers). In [29], several features (12 feature values per frame and 108 pooled feature values per video sequence) are extracted from each video and then, the number of features are reduced after a feature selection using the Extra Trees. Finally, a random forest regression is applied to predict the video quality score. However, no analytical results is reported for the selected features and the importance of the selected features.

Table 1 Summary of video features

	Feature	# features	Exe. Time
1	Frame Rate	1	0
2	Spatial Gradient	24	1.9
3	Spatiotemporal Gradient	48	9.9
4	Spatial Laplacian	12	1.8
5	Spatiotempral Laplacian	24	7.3
6	Spatial Gradient Amplitude	12	2.0
7	Spatiotempral Gradient Amplitude	24	10.1
8	Spatial Angular Information	8	2.0
9	Spatiotempral Angular Information	16	10.1
10	Luma Information	12	2.3
11	Chroma Information	24	3.9
12	Temporal Information	12	2.6
13	Colorfulness	42	2.9
14	LMSCN Statistics	216	39.7
15	BRISQUE Scores	6	39.8
16	V.BLINDS	46	771.7
	(16_NSS)	(37)	(99.1)
	(16_NVS)	(7)	(539.5)
	(16_Motion)	(2)	(133.1)
17	VIIDEO features	72	225.4
18	HOSA Scores	6	62.8
19	CORNIA Scores	6	4.5

As explained above, the main distinguishing part in different feature-based methods (not using deep learning) is the set of used features. Indeed, the SVR has been the dominant regressor. In this paper, we consider a collection of features in the existing works and apply different regressors in a single test. By choosing the best regressor in this test (Extra Trees), we find the optimal feature-set selection by an exhaustive search.

3 Proposed method

3.1 Overview

As mentioned earlier, we consider machine learning approaches for the purpose of no-reference video quality assessment. After describing several video quality metrics in Section 3.2, we shall train several regressors. From machine-learning point of view, each studied method consists of a number of features and a regressor, which translates the feature values into a single quality score. To distinguish between the impacts of the features and the regressors in the final performance , we consider two experiments. In the first experiment, by including a full list of features, we train various regressors. The goal of this experiment is to evaluate the effect of the regressor and to identify the best choice. In the second experiment, we fix the regressor (the best performer in the first experiment) and repeat the training phase by considering all combinations of the features. The result of the latter experiment is likely to identify the features with the highest impact on the quality assessment task. Based on this experiment, we can also find the frequency of features in the top performing models and use that as an importance metric for each feature. In addition, as a special case, we check the performance of single and pairs of features to provide a better intuition about the overlapping or complementary nature of the information encoded by the features. As another metric for the importance of each feature, we train an Extra Trees regressor and find the mean decrease impurity (MDI) score [30] over all trees in the trained regressor.

3.2 Video Features

A video file can be considered as a cube of spatiotemporal data indexed by x, y, and t axes; x and y represent the horizontal and vertical spatial axes, while t represents the time axis. The standard frames of the video are simply the xy slices of this cube. Similarly, one can define xt and yt slices as spatiotemporal frames. As each video pixel consists of luma and chroma components, we can decompose the video cube into one luma cube and two chroma cubes. For better illustration, a sample video cube and the mentioned slices are shown in Figure 1.

In this work, we consider various features extracted from the frames (xy slices), frame differences (temporal features) or spatiotemporal slices (xt and yt). To make the method independent of the video length and frame size, we use the statistics of these features (e.g., mean and variance) instead of their raw values. The summary of considered features and their execution time* are provided in Table1. The explanation of these features is provided below.

3.2.1 Frame rate: The number of frames within each second of video. This value is a single-element feature set which can identify low quality video files, when the frame rate is below the standards.

3.2.2 Spatial Gradient: Each video frame represents an image; the horizontal and vertical gradients (derivatives) of this image reveal the edges. For this feature, we apply two 3×3 filter kernels (horizontal and vertical directions) on the luminance channel of the frame. Next, we compute the mean and standard deviation for each frame (4 values in total). The final features in this set are determined by the minimum, maximum, mean, standard deviation, skewness, and kurtosis of these values along the video (total of 24 features).

3.2.3 Spatiotemporal Gradient: The introduced spatial gradient features are directly extracted from the frames. As explained earlier, we can use spatiotemporal slices (xt and yt) instead of the frames and apply the same strategy as above to obtain 48 features; 24 for xt slices and 24 for yt slices. While the gradient of the frames reveal spatial edges, the gradient of the spatiotemporal slices can reveal scene changes and temporal movements of the objects.

3.2.4 Spatial Laplacian: The Laplacian of a frame is a particular 2nd order derivative of the data and is known to be rotation and scale-invariant. We implement the Laplacian operator via a 5×5 filter kernel applied to the luminance channel of the frame. We find the mean and standard deviation of the result over each frame and use the minimum, maximum, mean, standard deviation, skewness, and kurtosis of these values along the video as the features (total of 12 features).

3.2.5 Spatiotemporal Laplacian: To provide uniformity along all the three axes of the video cube, we also include the features related to the Laplacian of the spatiotemporal slices. Then, similar to the procedure explained for spatial Laplacian features, we obtain 24 features representing the spatiotemporal Laplacian (12 features for each of the xt and yt slices).

3.2.6 Spatial Gradient Amplitude: The gradient of an image is a 2D vector at each pixel (vertical and horizontal derivatives). We define the amplitude of this vector as the spatial information at a pixel:

$$\|\mathbf{g}\| = \sqrt{g_x^2 + g_y^2}, \tag{1}$$

where g_x and g_y are the horizontal and vertical derivatives of the luminance channel of the frame at a given pixel, respectively. We should highlight that the information of the gradient components is already included in the gradient features. However, the relationship between the gradient components and their amplitude is nonlinear.

** On a machine with an Intel(R) Core(TM) i7-8700K CPU over videos of KonVid-1k dataset (each video has a length of around 8 seconds with 24 - 30 frames per second and 960×540 resolution.).*

Therefore, the quality estimates of a regressor based on the gradient amplitude features might be better than the ones based on the gradient components, and vice versa. Hence, we include both sets of features. Similar to the previous features, we calculate the mean and standard deviation of these amplitudes over a frame and report the minimum, maximum, mean, standard deviation, skewness, and kurtosis of them along the video as the features (total of 12 features).

3.2.7 Spatiotemporal Gradient Amplitude: We repeat the above procedure for spatiotemporal xt and yt slices to obtain two sets of 12 features (24 features in total).

3.2.8 Spatial Angular Information: As we included the features related to the components of the gradient and its amplitude separately, it is logical to include a number of features that describe the angle of the gradient. Indeed, the direction of the gradient vector at each pixel reveals the local orientation of the edges, and human visual system is sensitive to angles [31]. We initially define the gradient angle at each pixel as

$$\theta_{\mathbf{g}} = \tan^{-1}\left(\frac{g_y}{g_x}\right).$$

Next, we discard the pixels at which the gradient amplitude value is below a threshold (this threshold is set as 20). This is due to the fact that the orientation of the gradient vector does not carry much information when the gradient amplitude is small. For the remaining pixels, we calculate the mean and standard deviation over the frame and report the mean, standard deviation, skewness, and kurtosis of them along the video as the features (total of 8 features). It is worth mentioning that the angular information is differently represented in [31].

3.2.9 Spatiotemporal Angular Information: Similar to the previous parts, we repeat the procedure for spatial angular features on xt and yt slices to obtain 16 spatiotemporal angular features (two sets of 8 features).

3.2.10 Luma Information: The RGB representation of image pixels is one of the popular representations in which the color and brightness information are simultaneously encoded. The decomposition into brightness and color information is also very common for encoding video frames. The luma, the luminance, or the brightness component includes the morphological and geometrical content of the video. For instance, the sharpness (or blurriness) of the frames could be directly inferred from the luma channel. For this purpose, we compute the mean and standard deviation of the luminance channel of each video frame. Then, we report the minimum, maximum, mean, standard deviation, skewness, and kurtosis of these values along the video as the features (total of 12 features). It is conceptually possible to define the luma channel for xt and yt slices; however, we ignore such spatiotemporal luma features in this paper.

3.2.11 Chroma Information: Complementary to the luma channel, each video frame (xy) consists of two chroma components that encode the color content of the frame. The quality metrics related to the color content are oftentimes derived from these components. To provide uniformity with the luma channel, we derive the mean and standard deviation for each of the chroma components separately, and use the minimum, maximum, mean, standard deviation, skewness, and kurtosis of these values along the video as the chroma-related features (total of 24 features).

3.2.12 Temporal Information: The difference between neighboring pixels within a single frame is considered as spatial information and is properly captured using the derivative-type operators (such as gradient). In contrast, the temporal information of the video is understood as the pixel variations along the t axis. To extract the temporal features, we first subtract each luma frame from its previous luma frame in a pixel-by-pixel fashion (pixels at the same spatial locations). Next, we form the temporal features by evaluating the mean and standard deviation of each difference frame, and then, evaluating the minimum, maximum, mean, standard deviation, skewness, and kurtosis of these values along the video (total of 12 features).

3.2.13 Colorfulness: The introduced chroma features mainly capture the intensity histogram of the present colors in the video. The colorfulness metric $M^{(3)}$ in [32] measures how varied the colors are in a video file. To define this metric, let $rg = R - G$ and $yb = 0.5(R + G) - B$, where R, G and B stand for the red, green, and blue components of each pixel. Now, we have:

$$M^{(3)} = \sigma_{rgyb} + 0.3\mu_{rgyb}, \tag{2}$$

where

$$\sigma_{rgyb} = \sqrt{\mathrm{Var}(rg) + \mathrm{Var}(yb)} \tag{3}$$

$$\mu_{rgyb} = \sqrt{\mathrm{mean}(rg)^2 + \mathrm{mean}(yb)^2}. \tag{4}$$

(Var and mean are acting over the values of each xy frame) To provide a more comprehensive set of colorfulness features, we keep $\mathrm{Var}(rg)$, $\mathrm{mean}(rg)$, $\mathrm{Var}(yb)$, $\mathrm{mean}(yb)$, σ_{rgyb} and μ_{rgyb} in addition to the overall colorfulness measure $M^{(3)}$. For each of these seven frame-level values, we evaluate the minimum, maximum, mean, standard deviation, skewness, and kurtosis along the video (total of 42 features).

3.2.14 LMSCN Statistics: The Luma information explains the statistics of the brightness globally over the frames. For a local (spatially) measure of brightness we use the luminance channel in the form of Mean Subtracted Contrast Normalized (MSCN). For a mathematical description, let $F(i,j)$ represent a $M \times N$ frame (luminance channel) with $i \in \{1, 2, ..., M\}$, $j \in \{1, 2, ..., N\}$. The MSCN frame $\widehat{F}(i,j)$ is defined as

$$\widehat{F}(i,j) = \frac{F(i,j) - \mu(i,j)}{\sigma(i,j) + C}, \tag{5}$$

where $\mu(i,j)$ and $\sigma(i,j)$ are the weighted local mean and standard deviation of the frame

$$\mu(i,j) = \sum_{k=-3}^{3}\sum_{l=-3}^{3} w_{k,l}F(i+k, j+l), \tag{6}$$

$$\sigma(i,j) = \sqrt{\sum_{k=-3}^{3}\sum_{l=-3}^{3} w_{k,l}[F(i+k, j+l) - \mu(i,j)]^2}, \tag{7}$$

and $C = 1$ is a regularizing constant to avoid instability. The 7×7 weight matrix $\mathbf{w} = \left[w_{k,l}\right]_{|k|,|l|\leq 3}$ mimics a symmetric 2D Gaussian filter with

$$w_{k,l} = \frac{\mathrm{e}^{-\frac{18}{49}(k^2+l^2)}}{\sum_{|k|,|l|\leq 3}\mathrm{e}^{-\frac{18}{49}(k^2+l^2)}}.$$

The $\frac{18}{49}$ constant is chosen such that the main lobe of the Gaussian filter ($\pm 3\sigma$) fits within the 7×7 window.

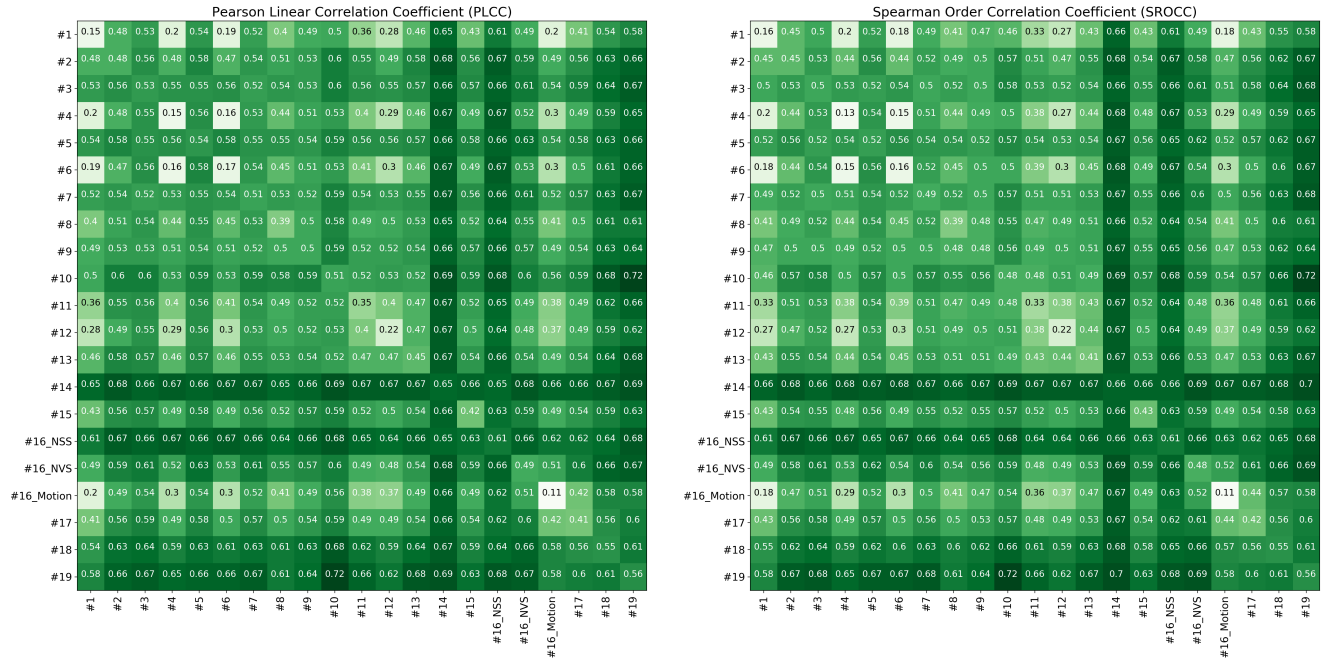
The histogram of $\widehat{F}(i,j)$ is known to be fairly described by the Generalized Gaussian Distribution (GGD) of the form[10, 15]:

$$f(x;\alpha,\beta) = \frac{\alpha}{2\beta\Gamma(\frac{1}{\alpha})}\exp\left(-\frac{|x|^\alpha}{\beta^\alpha}\right),$$

where $\Gamma(.)$ is the gamma function. Besides, $\widehat{F}(i,j) \times \widehat{F}(\tilde{i},\tilde{j})$ for $|i-\tilde{i}| + |j-\tilde{j}| = 1$ also fairly follows an Asymmetric Generalized Gaussian Distribution (AGGD):

$$f(x;\gamma,\beta_l,\beta_r) = \begin{cases} \frac{\gamma}{(\beta_l+\beta_r)\Gamma(\frac{1}{\gamma})}\exp\left(-\frac{|x|^\gamma}{\beta_l^\gamma}\right), \forall x \leq 0, \\ \frac{\gamma}{(\beta_l+\beta_r)\Gamma(\frac{1}{\gamma})}\exp\left(-\frac{x^\gamma}{\beta_r^\gamma}\right), \forall x \geq 0. \end{cases}$$

The mean of this distribution is given by $\eta = (\beta_r - \beta_l)\frac{\Gamma(\frac{2}{\gamma})}{\Gamma(\frac{1}{\gamma})}$. The parameters $(\gamma, \beta_l, \beta_r, \eta)$ for four orientations of pairing in addition



(a) (b)

Fig. 2: (a) PLCC and (b) SROCC of the ground-truth MOS and the predicted MOS by considering up to two subsets of features. The results are achieved using 5-folded cross-validation. The main diagonals represents single group feature subsets. The brighter (darker) cells have lower (higher) scores.

to (α, β) form a total of 18 statistical parameters. We further consider the image in two scales, the original one and a down-sampled version by factor two. We estimated the involved parameters (except for η) using a moment-matching approach [33]. Finally, we form the features based on these 36 parameters by finding the minimum, maximum, mean, standard deviation, skewness, and kurtosis along the video (total of 216 features).

3.2.15 BRISQUE Scores: BRISQUE [15] is one of the well-known methods in estimating the quality of an image. Here, we evaluate the BRISQUE score for each frame and generate 6 corresponding features by computing the minimum, maximum, mean, standard deviation, skewness, and kurtosis of the scores along the video.

3.2.16 V.BLINDS features: As described earlier, the V.BLINDS method [2] uses three groups of video features: frame-level Natural Scene Statistics (NSS) [10]), Natural Video Statistics (NVS), and some motion-related features. Here, we consider these feature groups separately (referred to as #16_NSS, #16_NVS, and #16_Motion).

3.2.17 VIIDEO features: The method of [3] called VIIDEO estimates the quality based on the MSCN of frame differences. The method generates 12 parameters for any 3 consecutive frames. Here, we produce 72 features by evaluating the minimum, maximum, mean, standard deviation, skewness, and kurtosis of these parameters along the video.

3.2.18 HOSA Scores: HOSA is another NR-IQA method introduced in [9]. Similar to BRISQUE features, we extract the HOSA score of each frame and generate 6 related features by evaluating the minimum, maximum, mean, standard deviation, skewness, and kurtosis of the scores along the video.

3.2.19 CORNIA Scores: We repeat the previous step by replacing HOSA with CORNIA; the latter NR-IQA method is introduced in [4] and provides a quality score for each frame. We generate 6 CORNIA-related features by evaluating the minimum, maximum, mean, standard deviation, skewness, and kurtosis of this scores along the video.

4 Experiments and Discussion

In this section, we explain our experiments and discuss our results. First, we describe our experimental setup in Section 4.1. Next, we evaluate and analyze different regresssor models in Section 4.2. Finally, we discuss the results in Section 4.3.

4.1 Experimental setup

As stated previously, we use the KonVid-1k video dataset[8] to train and test the models in our experiments. KonVid-1k dataset includes 1200 in-the-wild and public domain video sequences with the diversity in terms of content and quality. Each video in this dataset has a length of around 8 seconds with 24 - 30 frames per second and 960×540 resolution. The videos are annotated in 5-point absolute category rating (ACR) scale (1 for low quality, up to 5 for high quality) using a crowd-working platform with a total number of 642 workers from 64 countries; each video has received a minimum of 50 judgments. We first extract all the features listed in Section 3.2 (total of 611 features) for each of the video sequence in this dataset. We use the extracted features to train our regressors. The implementations of regression models are based on the scikit-learn package [34] in Python. The source codes of our experiments are publicly available*.

4.2 Evaluating and Analyzing regressor models

In our first experiment, we combine all these features into a single quality score using each of the following 9 regressors: Support Vector Regressor (SVR), Decision Tree, Random Forest, Extra Trees,

*https://github.com/otroshi/FeatureBased-NRVQA

Table 2 The performance of different regressors in terms of Pearson linear correlation coefficient (PLCC), Spearman rank order correlation coefficient (SROCC), and root mean square error (RMSE) using 10-folded cross-validation.

Regressor	Train Data			Test Data		
	↑PLCC	↑SROCC	↓RMSE	↑PLCC	↑SROCC	↓RMSE
SVR	0.601	0.604	0.103	0.535	0.536	0.109
Decision Tree	0.981	0.972	1.7×10^{-5}	0.576	0.588	0.116
Random Forest	0.977	0.975	0.032	0.760	0.757	0.082
Extra Trees	**0.999**	**0.999**	$\mathbf{6.6 \times 10^{-6}}$	**0.771**	**0.767**	**0.081**
Gradient Boosting	0.999	0.999	0.007	0.753	0.750	0.083
AdaBoost	0.913	0.901	0.056	0.756	0.754	0.083
XGBoost	0.752	0.753	0.110	0.676	0.671	0.114
MLP	0.716	0.701	0.088	0.532	0.529	0.109
Gaussian Process	0.999	0.999	0.006	0.752	0.753	0.083

Gradient Boosting, AdaBoost[35], XGBoost[36], multi-layer perceptron (MLP)†, and Gaussian Process algorithms. For each case, we employed a 10-folded cross-validation and evaluated the Pearson linear correlation coefficient (PLCC), the Spearman rank order correlation coefficient (SROCC), and the root mean square error (RMSE) of the estimated score (using the test data). The results are reported in Table 2. This table indicates that the *Extra Trees* regressor has the best performance according to both metrics. In fact, the Extra Trees regressor is a decision-based algorithm with an ensemble learning approach. Other ensemble learning-based models such as Random Forest, AdaBoost, and Gradient Boosting achieve comparable (but slightly worse) performances.

Next, we fix the regressor as Extra Trees and focus on the subset of features. As V.BLINDS features (number 16 in our list in Section 3.2) consist of 3 groups of features, we can think of having 21 sets of features (instead of 19) and examine all possible $2^{21} - 1 \approx 2\text{e}6$ non-empty subsets. To increase the reliability of the regression outcome, we apply a 5-folded cross-validation; i.e., we randomly divide the dataset into 5 equal parts, use any 4 of them for the training and leave the fifth for the test. The average performance (over 5 possibilities of selecting the training files) is then reported. This implies that there are $5 \times (2^{21} - 1)$ regression procedures involved in our experiment. This is a computationally heavy task, but allows us to identify influential sets of features and feature combinations. The regression tasks combined (after the feature extraction procedure) took 550 hours (about 23 days) with a multiprocessing implementation on an Intel(R) Xeon(R) CPU E5-2695 v3 with 32 cores and 2.30 GHz frequency. In Tables 3 and 4, we show the top 20 feature subsets in terms of the achieved PLCC and SROCC metrics, respectively. The check-marks in these tables stand for the inclusion of a feature group in the considered subset. Tables 3 and 4 also include the complexity of each model in term of the average execution time for computing the corresponding features for videos in the KonVid-1k dataset*.

Besides the top 20 scorers shown in Tables 3 and 4, we have also shown the performance of pairs of feature groups in Figure 2. The main diagonal of the tables in this figure represent the performance of single feature groups. It is interesting to mention that PLCC and SROCC value of 0.72 is attainable with only two feature groups ($\#10$ and $\#19$), while the best achievable values among all subsets is less than 0.79 (for both metrics).

In another experiment, we train an Extra Trees regressor with all the extracted features and find the importance of each feature according to the number of splits in the trees. For this end, we calculate the mean decrease impurity (MDI) score [30] over all trees in the trained regressor. We consider the maximum MDI score for each feature in a group as the feature group importance score which is reported in Figure 3. As this figure shows, the feature groups #19, #18, and #10 have higher importance scores in the regressor structure.

4.3 Discussion

While SVR has been the dominant choice for the purpose of video quality assessment, our experiments show that the corresponding PLCC and SROCC values could be easily improved by replacing the regressor (see Table 2). In particular, our experiments shows that the Extra Trees regressor achieves the best performance among a number of well-known regressor well ahead of the SVR. In other words, by using a better regressor, it is possible to achieve a considerable performance gain. To the best of our knowledge, this fact is widely ignored in the existing methods. We should highlight that the Extra Trees method is a tree-based regressor which utilizes an ensemble learning approach. The results in Table 2 indicate that other ensemble-learning-based regressors such as Random Forest, Gradient Boosting, and AdaBoost are also possible alternatives to Extra Trees.

By adopting the Extra Trees regressor in our second experiment, we are indeed, selecting the best performing regressor according to both PLCC and SROCC metrics. The results in Tables 3 and 4, besides identifying the best performing feature subsets, show that with as few as 8 feature groups (rows 4 and 1 in Tables 3 and 4, respectively) out of the total of 21, it is possible to surpass the performance of a full-feature setup. It is worth mentioning that $232,071$ and $63,859$ feature subsets achieved a performance superior to the full-feature setup according to SROCC and PLCC metrics, respectively. It seems that the PLCC metric is a more restricting performance criterion as there are $63,498$ feature-subsets that surpass the full-feature setup with both metrics. In an attempt to determine the feature groups with higher impact (in the VQA task), we have evaluated the frequency of appearance of each feature group in the $63,498$ feature-subsets that result in a better performance than the full-feature setup (according to both metrics). The result is shown in Figure 4. As expected from the list of top 20 feature subsets, the feature groups of #10, #19, and #16_NVS have the highest frequency. It is interesting to mention that feature groups #10 and #19 have appeared in all the $63,498$ feature-subsets. Additionally, these two feature groups (#19 and #10) are among the top MDI scorers according to Figure 3. Besides, these two groups include spatial features while the #16_NVS group includes the spatiotemporal features.

As explained earlier, the list of considered features could be generally divided into two categories of spatial (#1, #2, #4, #6, #8, #10, #11, #13, #14, #15, #16_NSS, #18, #19) and spatiotemporal (#3, #5, #7, #9, #12, #16_NVS, #16_Motion, #17). Indeed, the spatiotemporal features play the main role in distinguishing the video quality assessment from image quality assessment. The results in Table 3 and Table 4 indicate that all the top scoring feature subsets are composed of both categories. For better clarification, the best performing feature subsets composed solely of spatial features achieve PLCC and SROCC values of 0.759 and 0.758, respectively, and are ranked $1,672,515^{th}$ and $1,642,710^{th}$ overall. We further observe that features in the same category do not have the same importance level; for instance, the spatial HOSA features (#19) are dominantly used in all the top 20 feature subsets, while the spatial LMSCN features (#14) rarely appear in top scoring feature subsets. The statistical features of the luma channel are also among the important spatial features (#10 and #16_NSS); note that the luma LMSCN features (#14) are rarely used mainly because other luma features (#10 and #16_NSS) are included as popular substitutions. A similar observation is also valid for spatiotemporal features: #16_NVS, #16_Motion and #9 (which is introduced in this paper) are among popular spatiotemporal features within top performing subsets, while VIIDEO features (#17) are not equally popular. It should be highlighted that besides the choice of the features, their combinations also play a significant role in determining the overall performance.

Next, we consider the performance of single features and pairs of features. The main diagonal of tables in Figure 2 show the performance of single features in predicting the MOS values. Interestingly, in both cases of PLCC and SROCC the LMSCN features (#14, and consisting of 216 features) have the best performance achieving the

†*The MLP used in this experiment has* 3 *hidden layers each with* 500 *neurons and* tanh *activation function. The network was optimized by Adam optimizer[37] with an adaptive learning rate. Please note that this structure was achieved by experiment.*

**Each video in this database has a length of around* 8 *seconds with* 24 - 30 *frames per second and* 960×540 *resolution.*

Table 3 The top 20 feature subsets in terms of PLCC using a 5-folded cross-validation. The regressor is Extra Trees and the check-marks indicate the inclusion of a feature group in a feature subset. The execution time (Exe. Time) refers to the average time in seconds for computing the corresponding features for the videos in KonVid-1k dataset.

	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	#16_NSS	#16_NVS	#16_Motion	#17	#18	#19	PLCC	Exe. Time
1	✓	✓							✓	✓						✓	✓	✓		✓	✓	**0.786**	853.3
2	✓			✓					✓	✓						✓	✓	✓		✓	✓	**0.785**	853.2
3	✓	✓					✓	✓		✓						✓	✓	✓		✓	✓	**0.784**	855.3
4	✓					✓			✓	✓						✓	✓			✓	✓	**0.784**	720.3
5	✓					✓			✓	✓						✓	✓	✓			✓	**0.784**	790.6
6		✓							✓	✓						✓	✓	✓		✓	✓	**0.784**	853.3
7	✓	✓				✓		✓	✓	✓	✓					✓	✓			✓	✓	**0.784**	728.1
8	✓	✓							✓	✓	✓				✓	✓	✓	✓		✓	✓	**0.784**	897.0
9	✓	✓		✓		✓		✓		✓	✓					✓	✓	✓		✓	✓	**0.784**	852.9
10	✓	✓		✓					✓	✓					✓	✓	✓			✓	✓	**0.783**	761.8
11	✓					✓		✓		✓	✓					✓	✓	✓		✓	✓	**0.783**	849.2
12	✓	✓		✓				✓		✓						✓	✓	✓		✓	✓	**0.783**	847.0
13		✓							✓	✓	✓				✓	✓	✓	✓		✓	✓	**0.783**	897.0
14	✓			✓		✓			✓	✓	✓				✓	✓	✓	✓		✓	✓	**0.783**	898.9
15	✓	✓				✓		✓		✓	✓					✓	✓	✓		✓	✓	**0.783**	851.1
16	✓			✓		✓	✓	✓		✓	✓					✓				✓	✓	**0.783**	188.5
17	✓	✓		✓		✓			✓	✓	✓					✓	✓	✓		✓	✓	**0.783**	861.0
18	✓	✓						✓	✓	✓	✓					✓	✓	✓		✓	✓	**0.783**	859.2
19	✓	✓							✓	✓			✓		✓	✓	✓			✓	✓	**0.782**	762.9
20		✓		✓					✓	✓	✓					✓	✓			✓	✓	**0.782**	725.9

Table 4 The top 20 feature subsets in terms of SROCC using a 5-folded cross-validation. The regressor is Extra Trees and the check-marks indicate the inclusion of a feature group in a feature subset. The execution time (Exe. Time) refers to the average time in seconds for computing the corresponding features for the videos in KonVid-1k dataset.

	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	#16_NSS	#16_NVS	#16_Motion	#17	#18	#19	SROCC	Exe. Time
1	✓					✓			✓	✓						✓	✓	✓			✓	**0.787**	790.6
2	✓			✓					✓	✓	✓					✓	✓	✓			✓	**0.786**	794.3
3	✓	✓				✓	✓		✓	✓					✓	✓	✓	✓	✓		✓	**0.786**	1067.8
4	✓					✓		✓	✓	✓	✓					✓	✓	✓			✓	**0.785**	796.5
5	✓					✓			✓	✓		✓			✓	✓	✓				✓	**0.785**	699.9
6	✓	✓					✓		✓	✓	✓		✓	✓		✓	✓	✓			✓	**0.785**	847.1
7	✓			✓		✓			✓	✓						✓	✓	✓	✓		✓	**0.785**	1017.8
8	✓								✓	✓	✓	✓			✓	✓	✓				✓	**0.785**	701.8
9	✓			✓					✓	✓						✓	✓	✓		✓	✓	**0.785**	853.2
10				✓		✓			✓	✓						✓	✓				✓	**0.785**	659.3
11	✓	✓				✓	✓		✓	✓		✓	✓	✓		✓	✓	✓			✓	**0.785**	847.8
12	✓	✓				✓			✓	✓	✓	✓				✓	✓	✓			✓	**0.784**	799.0
13						✓			✓	✓		✓				✓	✓	✓			✓	**0.784**	793.2
14	✓					✓			✓	✓						✓	✓				✓	**0.784**	657.5
15	✓			✓	✓	✓			✓	✓	✓	✓				✓	✓	✓	✓		✓	**0.784**	1031.6
16	✓	✓							✓	✓						✓	✓	✓		✓	✓	**0.784**	853.3
17		✓			✓	✓			✓	✓	✓			✓			✓				✓	**0.784**	611.2
18	✓	✓		✓					✓	✓						✓	✓				✓	**0.784**	659.2
19				✓					✓	✓						✓	✓				✓	**0.784**	657.3
20	✓	✓		✓		✓			✓	✓	✓				✓	✓	✓				✓	**0.784**	704.9

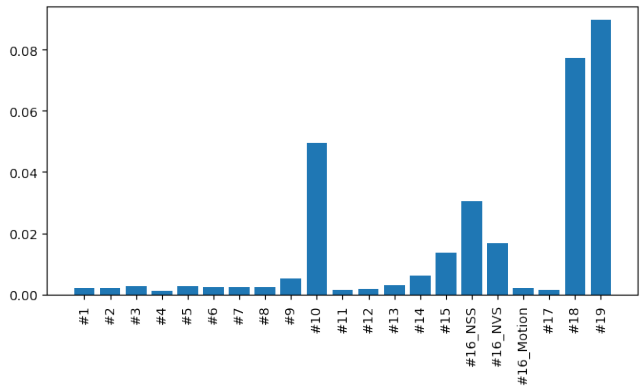


Fig. 3: Feature group importance using the feature MDI scores.

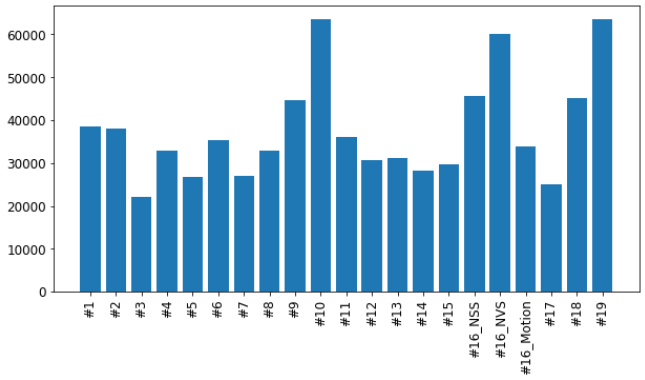


Fig. 4: Frequency of appearance of each feature group in models which have better performance than full-feature setup ($63,498$ models).

value 0.65 (either PLCC or SROCC). We should recall that these features are spatial and appeared rarely in the top 20 feature subsets. This shows that by considering other collection of luma features, one can outperform the LMSCN features; however, with restriction on the number of feature families, LMSCN family is among the bests. Another observation regarding the diagonal of the tables is that the spatial features are generally superior than the spatiotemporal ones. In other words, the performance of top 20 feature subsets are primarily based on spatial features, and spatiotemporal features have incremental role; i.e., we do not expect a huge degradation of the performance if we model the video data as a collection of unrelated images (frames). With a similar pattern, we observe that the highest performance among the pairs of feature families (non-diagonal elements in Figure 2) belong to $(\#10, \#19)$ (luma information and CORNIA scores) with PLCC and SROCC values of 0.72. Again, both features are spatial and the pair has a small performance gap compared to the top feature subsets (0.78).

To compare the results presented in this paper with the existing methods for video quality assessment, we have reported the PLCC and SROCC value of existing methods applied to the KonVid-1k database as well as the top performing feature subsets in Table 5. We should mention that except for the result of the proposed methods, the reported values of other methods on the KonVid-1k database are directly reported from the literature*[5, 13, 14, 16–18, 20, 24–28]. In this table, the method names shown in *italic* are based on deep learning approaches. Within the subgroup of feature-based methods (not using deep learning), Table 5 confirms that the proposed extra trees (F-ET) regression technique has the best performance. Furthermore, the proposed technique includes multiple choices (feature subsets) that achieve better performances compared to the existing methods; thus, one has partial flexibility in choosing the features. We also observe that the proposed feature-based method is superior to 7 of the total of 11 models based on deep learning. Moreover, the proposed model is 0.08 and 0.06 behind the best performing models in terms of PLCC and SROCC, respectively. While we acknowledge the performance gain of the best performing deep learning models, we should highlight that this gain is achieved at the expense of considerably higher computational cost both in the training and testing stages. Roughly speaking, a neural network with competitive results involves at least $50,000,000$ degrees of freedom (coefficients that should be learned), which complicates the training phase (both processing and storage requirements). In addition, the computational cost of the such neural networks for evaluating the overall output score for a given input (evaluating the score for a test video) is well above the simple regression techniques. This fact is commonly ignored as neural networks are widely implemented using parallel processing techniques which is not the case for regression techniques. All in all, the deep learning approaches are able to provide better results but with considerably more computational cost; in fast settings such as real-time video quality assessment, this computational cost might not be feasible. Another issue which requires attention is the small size of the training set in most available video datasets; the KonVid-1k which is by far the largest database, contains only 1200 video files [39–43]. Thus, a neural network with large number of parameters that is trained on this dataset is likely to be a tailored solution just for this database which cannot be easily adapted to other settings.

Table 5 Comparison of No-Reference Video Quality Assessment methods on KonVid-1k Database. The method names shown in *italic* are based on deep learning approaches. (The values of PLCC and SROCC for the existing methods are directly reported from the references in the last column.)

Method	PLCC	SROCC	Reference
VIIDEO[3]	-0.015	0.031	[5]
V.BLINDS[2]	0.565	0.526	[5]
FC Model [11]	0.492	0.472	[5]
STFC Model [5]	0.639	0.606	[5]
FRIQUEE[44]	0.740	0.740	[16]
HIGRADE[45]	0.720	0.730	[16]
BRISQUE[15]	0.626	0.654	[18]
NIQE[10]	0.546	0.544	[18]
STS-MLP [13]	0.407	0.420	[13]
STS-SVR [13]	0.680	0.673	[13]
GM-LOG[46]	0.663	0.658	[17]
V.CORNIA[12]	0.747	0.765	[13]
NARVAL [14]	0.689	N/A	[14]
BB-NARVAL [14]	0.761	N/A	[14]
TLVQM-SVR [16]	0.770	0.780	[16]
TLVQM-RFR [16]	0.740	0.740	[16]
VIDEVAL [17]	0.780	0.783	[17]
F-ET-PLCC-1st [proposed]	**0.786**	0.784	[Ours]
F-ET-SROCC-1st [proposed]	0.784	**0.787**	[Ours]
VSFA [18]	*0.744*	*0.755*	[18]
3D-CNN [20]	*0.781*	*0.771*	[20]
3D-CNN+LSTM [20]	***0.808***	***0.800***	[20]
RIRNet [28]	*0.781*	*0.775*	[28]
MDTVSFA [25]	*0.786*	*0.781*	[25]
F-RNN[26]	*0.683*	*0.710*	[26]
DNet[27]	*0.596*	*0.591*	[27]
STS-CNN100 [24]	*N/A*	*0.733*	[24]
STS-CNN200 [24]	*N/A*	*0.735*	[24]
PaQ-2-PiQ [47]	*0.601*	*0.613*	[17]
KonCept-512 [48]	*0.749*	*0.735*	[17]

Recently, [38] discovered that the actual results by implementing the methods in [21] are significantly worse than the ones reported in the original paper. Therefore, the methods of [21] are excluded from our comparison.

5 Conclusion

We studied the problem of no-reference video quality assessment with a machine-learning-based approach. First, we extracted several statistical features from all the video sequences in the KonVid-1k dataset. Next, we trained several models using different types of regressors with all the extracted features. The results showed that the change in the algorithm of regression may considerably affect the performance. We found Extra Trees regression to achieve the best performance. However, Support Vector Regressor (SVR) is mostly used in the prior works. To the best of our knowledge this is the first work to investigate several types of regressions for the purpose of video quality assessment. Next, we considered all possible selections of features, and evaluated the models using 5-folded cross-validation for each selection of feature subsets. This experiment helped us to identify the best selections of extracted features which achieved a better performance compared to the previous methods reported on KonVid-1k database. Moreover, the results show the importance of each feature subset in assessment of video quality. It also shows the common or complementary information of different feature subsets. The combinations of spatial and spatiotemporal feature subsets are among the best feature selections. We further evaluated the MDI scores for each feature, and accordingly found an importance score for each feature group. The results are consistent in identifying most important feature groups for the NR-VQA task.

6 References

1 Cisco White Paper: 'Cisco visual networking index: Forecast and methodology, 2017–2022', , 2019,

2 Saad, M.A., Bovik, A.C., Charrier, C.: 'Blind prediction of natural video quality', *IEEE Transactions on Image Processing*, 2014, **23**, (3), pp. 1352–1365

3 Mittal, A., Saad, M.A., Bovik, A.C.: 'A completely blind video integrity oracle', *IEEE Transactions on Image Processing*, 2016, **25**, (1), pp. 289–300

4 Ye, P., Kumar, J., Kang, L., Doermann, D. 'Unsupervised feature learning framework for no-reference image quality assessment'. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (Rhode Island, New England , USA: IEEE, 2012. pp. 1098–1105

5 Men, H., Lin, H., Saupe, D. 'Spatiotemporal feature combination model for no-reference video quality assessment'. In: Proceedings of the Tenth International Conference on Quality of Multimedia Experience (QoMEX). (Sardinia, Italy: IEEE, 2018. pp. 1–3

6 Seshadrinathan, K., Soundararajan, R., Bovik, A.C., Cormack, L.K.: 'Study of subjective and objective quality assessment of video', *IEEE Transactions on Image Processing*, 2010, **19**, (6), pp. 1427–1441

7 Li, X., Guo, Q., Lu, X.: 'Spatiotemporal statistics for video quality assessment', *IEEE Transactions on Image Processing*, 2016, **25**, (7), pp. 3329–3342

8 Hosu, V., Hahn, F., Jenadeleh, M., Lin, H., Men, H., Szirányi, T., et al. 'The konstanz natural video database (konvid-1k)'. In: Proceeding of the Ninth International Conference on Quality of Multimedia Experience (QoMEX). (IEEE, 2017. pp. 1–6

9 Xu, J., Ye, P., Li, Q., Du, H., Liu, Y., Doermann, D.: 'Blind image quality assessment based on high order statistics aggregation', *IEEE Transactions on Image Processing*, 2016, **25**, (9), pp. 4444–4457

10 Mittal, A., Soundararajan, R., Bovik, A.C.: 'Making a "completely blind" image quality analyzer.', *IEEE Signal Processing Letters*, 2013, **20**, (3), pp. 209–212

11 Men, H., Lin, H., Saupe, D. 'Empirical evaluation of no-reference vqa methods on a natural video quality database'. In: Proceedings of the Ninth International Conference on Quality of Multimedia Experience (QoMEX). (Sardinia, Italy: IEEE, 2017. pp. 1–3

12 Xu, J., Ye, P., Liu, Y., Doermann, D. 'No-reference video quality assessment via feature learning'. In: Proceedings of the IEEE International Conference on Image Processing (ICIP). (Paris, France: IEEE, 2014. pp. 491–495

13 Yan, P., Mou, X. 'No-reference video quality assessment based on perceptual features extracted from multi-directional video spatiotemporal slices images'. In: Optoelectronic Imaging and Multimedia Technology V. vol. 10817. (International Society for Optics and Photonics, 2018. p. 108171D

14 Lemesle, A., Marion, A., Roux, L., Gouaillard, A.: 'Narval: A no-reference video quality tool for real-time communications', *Electronic Imaging*, 2019, **2019**, (12), pp. 213–1

15 Mittal, A., Moorthy, A.K., Bovik, A.C.: 'No-reference image quality assessment in the spatial domain', *IEEE Transactions on Image Processing*, 2012, **21**, (12), pp. 4695–4708

16 Korhonen, J.: 'Two-level approach for no-reference consumer video quality assessment', *IEEE Transactions on Image Processing*, 2019, **28**, (12), pp. 5923–5938

17 Tu, Z., Wang, Y., Birkbeck, N., Adsumilli, B., Bovik, A.C.: 'Ugc-vqa: Benchmarking blind video quality assessment for user generated content', *IEEE Transactions on Image Processing*, 2021, **30**, pp. 4449–4464

18 Li, D., Jiang, T., Jiang, M. 'Quality assessment of in-the-wild videos'. In: Proceedings of the 27th ACM International Conference on Multimedia. (Nice, France: ACM, 2019. pp. 2351–2359

19 Varga, D., Szirányi, T.: 'No-reference video quality assessment via pretrained cnn and lstm networks', *Signal, Image and Video Processing*, 2019, pp. 1–8

20 You, J., Korhonen, J. 'Deep neural networks for no-reference video quality assessment'. In: Proceedings of the 26th IEEE International Conference on Image Processing (ICIP). (Taipei, Taiwan: IEEE, 2019. pp. 2349–2353

21 Varga, D.: 'No-reference video quality assessment based on the temporal pooling of deep features', *Neural Processing Letters*, 2019, pp. 1–14

22 Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z. 'Rethinking the inception architecture for computer vision'. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (Las Vegas, Nevada, USA, 2016. pp. 2818–2826

23 Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A. 'Inception-v4, inception-resnet and the impact of residual connections on learning'. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. (San Francisco, California, USA, 2017.

24 Yan, P., Mou, X. 'No-reference video quality assessment based on spatiotemporal slice images and deep convolutional neural networks'. In: Optoelectronic Imaging and Multimedia Technology VI. vol. 11187. (International Society for Optics and Photonics, 2019. p. 111870A

25 Li, D., Jiang, T., Jiang, M.: 'Unified quality assessment of in-the-wild videos with mixed datasets training', *arXiv preprint arXiv:201104263*, 2020,

26 Shahreza, H.O., Amini, A., Behroozi, H. 'No-reference video quality assessment using recurrent neural networks'. In: 2019 5th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS). (IEEE, 2019. pp. 1–5

27 Otroshi.Shahreza, H., Amini, A., Behroozi, H.: 'Predicting the empirical distribution of video quality scores using recurrent neural networks', *International Journal of Engineering*, 2020, **33**, (5), pp. 984–991

28 Chen, P., Li, L., Ma, L., Wu, J., Shi, G. 'Rirnet: Recurrent-in-recurrent network for video quality assessment'. In: Proceedings of the 28th ACM International Conference on Multimedia. (, 2020. pp. 834–842

29 Göring, S., Rao, R.R.R., Raake, A. 'nofu—a lightweight no-reference pixel based video quality model for gaming content'. In: 2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX). (IEEE, 2019. pp. 1–6

30 Breiman, L., Friedman, J., Stone, C.J., Olshen, R.A.: 'Classification and regression trees'. (CRC press, 1984)

31 Pinson, M.H., Wolf, S.: 'A new standardized method for objectively measuring video quality', *IEEE Transactions on Broadcasting*, 2004, **50**, (3), pp. 312–322

32 Hasler, D., Suesstrunk, S.E. 'Measuring colorfulness in natural images'. In: Human vision and electronic imaging VIII. vol. 5007. (International Society for Optics and Photonics, 2003. pp. 87–96

33 Lasmar, N.E., Stitou, Y., Berthoumieu, Y. 'Multiscale skewed heavy tailed model for texture analysis'. In: Proceedings of the 16th IEEE International Conference on Image Processing (ICIP). (Cairo, Egypt: IEEE, 2009. pp. 2281–2284

34 Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al.: 'Scikit-learn: Machine learning in Python', *Journal of Machine Learning Research*, 2011, **12**, pp. 2825–2830

35 Freund, Y., Schapire, R.E.: 'A decision-theoretic generalization of on-line learning and an application to boosting', *Journal of computer and system sciences*, 1997, **55**, (1), pp. 119–139

36 Chen, T., Guestrin, C. 'Xgboost: A scalable tree boosting system'. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. (San Francisco, California, USA, . pp. 785–794

37 Kingma, D.P., Ba, J. 'Adam: A method for stochastic optimization'. In: Proceedings of International Conference on Learning Representations (ICLR). (San Diego, California., USA, 2015.

38 Götz.Hahn, F., Hosu, V., Saupe, D.: 'Comment on" no-reference video quality assessment based on the temporal pooling of deep features"', *arXiv preprint arXiv:200504400*, 2020,

39 Li, D., Jiang, T., Jiang, M.: 'Recent advances and challenges in video quality assessment', *ZTE Communications*, 2019, **17**, (1)

40 Moorthy, A.K., Choi, L.K., Bovik, A.C., De.Veciana, G.: 'Video quality assessment on mobile devices: Subjective, behavioral and objective studies', *IEEE Journal of Selected Topics in Signal Processing*, 2012, **6**, (6), pp. 652–671

41 Nuutinen, M., Virtanen, T., Vaahteranoksa, M., Vuori, T., Oittinen, P., Häkkinen, J.: 'CVD2014 - a database for evaluating no-reference video quality assessment algorithms', *IEEE Transactions on Image Processing*, 2016, **25**, (7), pp. 3073–3086

42 Ghadiyaram, D., Pan, J., Bovik, A.C., Moorthy, A.K., Panda, P., Yang, K.: 'In-capture mobile video distortions: A study of subjective behavior and objective algorithms', *IEEE Transactions on Circuits and Systems for Video Technology*, 2018, **28**, (9), pp. 2061–2077

43 Sinno, Z., Bovik, A.C.: 'Large-scale study of perceptual video quality', *IEEE Transactions on Image Processing*, 2018, **28**, (2), pp. 612–627

44 Ghadiyaram, D., Bovik, A.C. 'Feature maps driven no-reference image quality prediction of authentically distorted images'. In: Human Vision and Electronic Imaging XX. vol. 9394. (International Society for Optics and Photonics, 2015. p. 93940J

45 Kundu, D., Ghadiyaram, D., Bovik, A.C., Evans, B.L.: 'No-reference quality assessment of tone-mapped hdr pictures', *IEEE Transactions on Image Processing*, 2017, **26**, (6), pp. 2957–2971

46 Xue, W., Mou, X., Zhang, L., Bovik, A.C., Feng, X.: 'Blind image quality assessment using joint statistics of gradient magnitude and laplacian features', *IEEE Transactions on Image Processing*, 2014, **23**, (11), pp. 4850–4862

47 Ying, Z., Niu, H., Gupta, P., Mahajan, D., Ghadiyaram, D., Bovik, A. 'From patches to pictures (paq-2-piq): Mapping the perceptual space of picture quality'. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (, 2020. pp. 3575–3585

48 Hosu, V., Lin, H., Sziranyi, T., Saupe, D.: 'Koniq-10k: An ecologically valid database for deep learning of blind image quality assessment', *IEEE Transactions*

on Image Processing, 2020, **29**, pp. 4041–4056