Multi-pooled Inception Features for No-reference Video Quality Assessment

Domonkos Varga
Department of Networked Systems and Services, Budapest University of Technology, Hungary

Keywords: No-reference Video Quality Assessment, Convolutional Neural Network.

Abstract: Video quality assessment (VQA) is an important element of a broad spectrum of applications ranging from automatic video streaming to surveillance systems. Furthermore, the measurement of video quality requires an extensive investigation of image and video features. In this paper, we introduce a novel feature extraction method for no-reference video quality assessment (NR-VQA) relying on visual features extracted from multiple Inception modules of pretrained convolutional neural networks (CNN). Hence, we show a solution which incorporates both intermediate- and high-level deep representations from a CNN to predict digital videos' perceptual quality. Second, we demonstrate that processing all frames of a video to be evaluated is unnecessary and examining only the so-called intra-frames saves computational time and improves performance significantly. The proposed architecture was trained and tested on the recently published KoNViD-1k database.

1 INTRODUCTION

In the process of generation, transmission, compression, and storage, digital videos may be corrupted by different distortion types resulting in the degradation of human perceptual quality. Thus, the precise prediction of digital videos' perceived quality is a very hot research topic in the image/video processing community. Video quality assessment (VQA) methods can be classified into two groups: subjective and objective VQA. In subjective VQA, the quality of digital videos is evaluated by human observers. Although subjective methods can achieve very high accuracy, their application is impossible in real-time systems because it is laborious and time-consuming to obtain enough number of ratings. In contrast, objective VQA algorithms make an attempt to create a model that is able to evaluate the perceptual quality of videos relying on our understanding of the human visual system (HVS), different mathematical tools or machine learning techniques. Furthermore objective VQA algorithms can be classified into full-reference (FR), reduced-reference (RR), and no-reference (NR) ones according to the availability of the reference, pristine digital video.

Recently, deep learning techniques have attracted a lot of attention in the fields of image processing (Szegedy et al., 2017), (Lu et al., 2018), (Habibzadeh et al., 2018) and visual quality assessment (Bianco et al., 2018), (Varga, 2019), (Dendi et al., 2019). Furthermore, Zhang *et al.* (Zhang et al., 2018) demonstrated that features extracted from pretrained convolutional neural networks (CNN) are highly effective for predicting perceptual quality. As a consequence, some NR-VQA methods rely on features extracted from pretrained CNNs (Ahn and Lee, 2018b), (Ahn and Lee, 2018a), (Varga, 2019). However, applying pretrained CNNs is not a straight-forward task because they require a fixed input size. Furthermore, previous methods (Ahn and Lee, 2018b), (Ahn and Lee, 2018a), (Varga, 2019) analyze all video frames one by one to predict perceptual video quality. To overcome the fixed input size constraint, previous methods took patches from the input video frames or resized and cropped them. In this paper, we make the following contributions. First, a content-preserving feature extraction method is introduced which relies on the Inception modules of pretrained CNNs, such as GoogLeNet (Szegedy et al., 2015) or Inception-v3 (Szegedy et al., 2016). Specifically, the frame-level visual features are extracted from multiple Inception modules of a pretrained CNN and pooled by a global average pooling (GAP) module together. This way, it is possible to obtain both intermediate- and high-level deep representations from the CNN. Second, we show that the number of frames from a video to be evaluated can be significantly reduced. Namely, previous works (Ahn and Lee, 2018b), (Ahn and Lee,

Varga, D.
Multi-pooled Inception Features for No-reference Video Quality Assessment.
DOI: 10.5220/0008978503380347
In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020) - Volume 4: VISAPP*, pages 338-347
ISBN: 978-989-758-402-2; ISSN: 2184-4321

2018a), (Varga, 2019) examine all frames. In contrast, we demonstrate that processing all frames is unnecessary and examining only the so-called *intra-frames* saves computational time and improves the performance significantly. Quality assessment of videos with authentic distortions is a relatively new topic. Most existing deep learning based NR-VQA methods work only for compression artifacts. From the publicly available VQA databases, we chose KoNViD-1k (Hosu et al., 2017) to train and test our proposed architecture. KoNViD-1k is the largest available collection of videos with authentic distortions and corresponding quality scores.

The rest of this paper is organized as follows. In Section 3 related and previous works are reviewed. Subsequently, we present our proposed method in Section 3. Experimental results and analysis are shown in Section 4. Finally, a conclusion is drawn in Section 5.

2 RELATED WORKS

Early NR-VQA methods mainly dealt with specific distortion types. For example, Zhang *et al.* (Zhang et al., 2009) measured blocking artifacts in low bit rate H.264/AVC videos by applying a specific temporal approach. Similarly, Borer (Borer, 2010) measured jerkiness using the mean squared error of consecutive frames. In contrast, Xue *et al.* (Xue et al., 2014) trained a neural network to predict the quality scores of images with jerkiness. In (Pastrana-Vidal et al., 2004), the authors detected freezing artifacts by monitoring the mean squared error between subsequent frames. In contrast, Yammine *et al.* (Yammine et al., 2012) detected freezing frames by applying motion estimation considerations. Søgaard *et al.* (Søgaard et al., 2015) applied quality-aware features using the NR-IQA method BRISQUE (Mittal et al., 2012a) together with temporal and spatial activity indices and codec specific features to detect MPEG-2 and H.264/AVC artifacts. A comprehensive review of early distortion specific methods can be found in (Shahid et al., 2014).

Later, general-purpose NR-VQA algorithms have also appeared. Saad *et al.* (Saad and Bovik, 2012) introduced a feature extraction method in the DCT domain. Furthermore, temporal information was also incorporated into their model by using motion coherency. Similarly in (Saad et al., 2014), the authors also extracted features with DCT using a spatiotemporal model. Finally, a trained SVR mapped the extracted features onto quality scores. In contrast, Video CORNIA (Xu et al., 2014) applied unsupervised feature learning to obtain frame-level feature vectors. Furthermore, an SVR was also used to predict frame-level quality scores based on the frame-level features. Finally, the video's quality was estimated by temporally pooling the frame-level scores. Unlike previous methods, VIIDEO (Mittal et al., 2015) relies on a predefined naturalness model and perceptual quality is predicted based on the deviation from this naturalness model. Men *et al.* (Men et al., 2017) extracted video-level feature vectors containing quality related measures, such as contrast or colorfulness. Subsequently, a trained SVR was used to predict perceptual quality. Later, this model was developed further by incorporating spatiotemporal features (Men et al., 2018). Korhonen (Korhonen, 2019) extracted low complexity features (such as motion intensity, motion consistency, jerkiness, *etc.*) from every second of a video to be assessed in order to determine a representative subset of video frames for computing high complexity features (such as total size of dark regions, number of dark regions, noise density, *etc.*). Finally, low and high complexity features are merged together and mapped onto quality scores using SVR or random forest regression.

Nowadays, deep learning has gained enormous popularity in video processing applications, especially convolutional neural networks (CNN) has been proved effective in video classification (Karpathy et al., 2014), action recognition (Ji et al., 2012), event detection (Xu et al., 2015), *etc.* Furthermore, Zhang *et al.* (Zhang et al., 2018) pointed out that features extracted from pretrained CNNs are highly effective for predicting digital images' perceptual quality. Thus, deep learning based NR-VQA algorithms have appeared also. Giannopoulos *et al.* (Giannopoulos et al., 2018) created a 3D CNN architecture with the common components, such as 3D convolutional layers, max-pooling layers, and fully-connected layers. Furthermore, the trained 3D CNN was used for video feature extraction. Subsequently, the extracted features were fed into a 1D CNN to map them into perceptual quality scores. Similarly, Liu *et al.* (Liu et al., 2018) introduced the so-called Video Multi-task End-to-end Optimized neural Network (V-MEON) for compressed videos where a 3D CNN based feature extractor and a codec classifier were applied to predict quality scores. Zhang *et al.* proposed a deep approach based on weakly supervised learning with a CNN and a resampling strategy. Specifically, an eight-layer CNN was trained first by weakly supervised learning to establish a relationship between the distortions of the 3D discrete cosine transform of video blocks and the corresponding weak labels judged by a FR-VQA algorithm. As a result, effective, quality-

aware features could be extracted from the trained CNN which were mapped onto quality scores by another trained network. You and Korhonen (You and Korhonen, 2019) applied a 3D CNN to extract spatiotemporal features from small clips of a video. Subsequently, a long-short term memory (LSTM) network was used to predict quality subscores for the small clips. Finally, these subscores were pooled together to produce quality scores for the entire video. In contrast, Varga (Varga, 2019) constructed a video-level feature vector by pooling together frame-level features obtained from resized and cropped video frames using a fine-tuned Inception-v3 (Szegedy et al., 2016) CNN model. Similarly, Li *et al.* (Li et al., 2019) extracted features from pretrained convolutional neural networks but these features were integrated into a network with a gated recurrent unit and a temporal pooling layer.

3 PROPOSED METHOD

The overview of the proposed method is depicted in Figure 1. First, the so-called intra-frames are extracted from the input video sequence. In video compression, only the changes are stored which measured between one frame and the next. An intra-frame is by definition a video frame which is completely stored. As a consequence, more intra-frames are inserted into a video sequence, the larger the video file size. Furthermore, the intra-frame interval can considerably influence the perceived quality of a video sequence (Reinhardt, 2010). If the encoder generates too many intra-frames for a given bit-rate, the perceived quality deteriorates. If the number of intra-frames is too low, the transitions in the video will be less smooth or accurate. As a consequence, the perceived quality will also decline. Second, frame-level feature vectors are extracted from the intra-frames with the help of pretrained CNNs, such as GoogLeNet (Szegedy et al., 2015) or Inception-v3 (Szegedy et al., 2016). Unlike previous CNN models, not everything happens sequentially in GoogLeNet, pieces of the network work in parallel. Inspired by a neuroscience model (Serre et al., 2007) where for handling multiple scales a series of Gabor filters were used with a two layer deep model. But contrary to the beforementioned model all layers are learned and not fixed. In GoogLeNet (Szegedy et al., 2015) architecture Inception layers are introduced and repeated many times. Subsequent improvements of GoogLeNet have been called Inception-v*N* where *N* refers to the version number put out by Google.

Subsequently, the frame-level feature vectors are temporally pooled to obtain the video-level feature vector which is mapped with the help of a trained SVR to a perceptual quality score.

3.1 Frame-level Feature Vectors

The pipeline of the frame-level feature extraction is depicted in Figure 3. To extract frame-level feature vectors, GoogLeNet (Szegedy et al., 2015), Inception-v2 (Szegedy et al., 2016), and Inception-v3 (Szegedy et al., 2016) networks were considered as base models in this study. Specifically, global average pooling (GAP) layers are attached to the output of each Inception module. Similar to max- or min-pooling layers, GAP layers are applied in CNNs to reduce the spatial dimensions of convolutional layers. However, a GAP layer carries out a more extreme type of dimensional reduction than a max- or min-pooling layer. Namely, an $h \times w \times d$ block is reduced to $1 \times 1 \times d$. In other words, a GAP layer reduces a feature map to a single value by taking the average of this feature map (Figure 2 illustrates GAP layer). By adding GAP layers to each Inception module, we are able to extract resolution independent features at different levels of abstraction. Namely, the feature maps produced by neuroscience models inspired Inception modules have been shown representative for object categories and correlate well with human perceptual quality judgments (Szegedy et al., 2015), (Szegedy et al., 2016). As already mentioned, a vector is extracted over each Inception module using a GAP layer. Let $\mathbf{f}_k$ denote the vector extracted from the kth Inception module. The video frame's feature vector is obtained by concatenating the vectors extracted over each Inception module. Formally, we can write $\mathbf{F} = \mathbf{f}_1 \oplus \mathbf{f}_2 \oplus \ldots \oplus \mathbf{f}_N$ where N denotes the number of Inception modules in the base CNN and $\oplus$ stands for the concatenation operator.

3.2 Video-level Feature Vectors

The frame-level feature vectors of a video sequence are temporally pooled to produce the video-level feature vector. Let N_k denote the number of intra-frames and let $\mathbf{F}^{(i)}$ stand for the frame-level feature vector related to the ith intra-frame. In this paper, we adopted average pooling which proved the best choice for visual quality assessment in many works (Varga, 2019), (Bianco et al., 2018). Average pooling can be formally expressed as:

$$\mathbf{V}_i^{avg} = \frac{1}{N_k} \sum_{j=1,\ldots,N_k} \mathbf{F}_i^{(j)}, \tag{1}$$

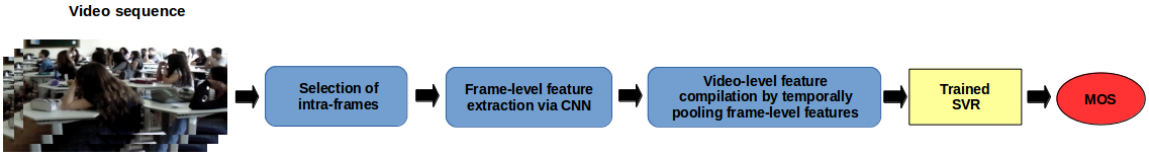


Figure 1: Architecture of the proposed NR-VQA method. First, the so-called intra-frames are extracted from a video sequence using the free and open-source *FFmpeg*. Second, frame-level feature vectors are extracted from the intra-frames via a pretrained CNN. Third, video-level feature vectors are formed by temporally pooling the frame-level features. Finally, the video-level feature vectors are mapped onto perceptual quality scores using a trained SVR.

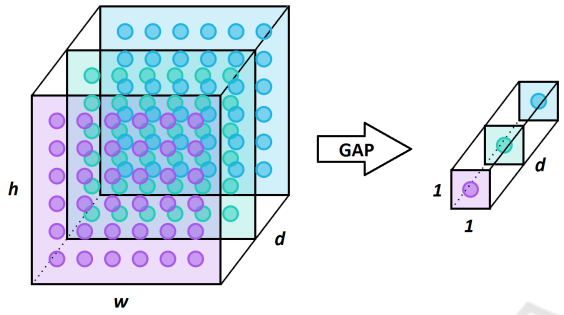


Figure 2: Illustration of global average pooling (GAP) layer. A GAP layer reduces the dimensions $h \times w \times d$ to $1 \times 1 \times d$ by averaging across $h \times w$.

where $\mathbf{F}_i^{(j)}$ denotes the ith entry of the jth frame-level feature vector, and $\mathbf{V}_i$ stands for the ith entry of the video-level feature vector.

Subsequently, an SVR (Drucker et al., 1997) with radial basis function (RBF) kernel is trained to learn the mapping between video-level feature vectors and corresponding perceptual quality scores.

3.3 Database Compilation and CNN Fine-tuning

As already mentioned, KoNViD-1k (Hosu et al., 2017) video quality database was utilized to train and test the proposed architecture. Specifically, KoNViD-1k consists of 1,200 videos with authentic distortions sampled from YFCC100m (Thomee et al., 2015) database. 840 sequences were randomly selected for training, 120 sequences for validation, and 240 sequences for testing. The video sequences' spatial resolution is 960×540 in KoNViD-1k and the frame rate is 25, 27, or 30 fps. Moreover, the length of video sequences fluctuates between 7 and 8 seconds. The MOS distribution is depicted in Figure 4.

The intra-frames were extracted from the training and validation videos. Furthermore, the extracted intra-frames inherited the quality scores of their source videos. The fine-tuning of base CNN models were carried out on the extracted intra-frames of the training and validation videos. Specifically, the output softmax layers were replaced by linear, regression layers with one neuron and mean squared error (MSE) loss function. Moreover, we made experiments to find the optimal input resolution for our model. To this end, we trained the CNN models using the original resolution of KoNViD-1k (960×540) and two down-sampled resolutions (480×270 and 299×299).

4 EXPERIMENTAL RESULTS AND ANALYSIS

In this section, experimental results and analysis are shown. First, the evaluation metrics are presented. Second, the software packages and hardware resources used in the implementation and testing process are given. Subsequently, we present a detailed parameter study related to the proposed architecture. Finally, a performance comparison to the state-of-the-art is shown.

4.1 Evaluation Metrics

The evaluation of objective VQA algorithms is based on measuring the correlation between ground-truth quality scores and predicted quality scores. Two parameters: Pearson's linear correlation coefficient (PLCC) and Spearman's rank order correlation coefficient (SROCC) are widely used to this end in the literature. The PLCC between dataset A and B is defined as

$$PLCC(A,B) = \frac{\sum_{i=1}^{n}(A_i - \overline{A})(B_i - \overline{B})}{\sqrt{\sum_{i=1}^{n}(A_i - \overline{A})^2}\sqrt{\sum_{i=1}^{n}(B_i - \overline{B})^2}}, \quad (2)$$

where $\overline{A}$ and $\overline{B}$ denote the average of set A and B, A_i and B_i stand for the ith element of set A and B, respectively. For two ranked sets A and B, SROCC is calculated as

$$SROCC(A,B) = \frac{\sum_{i=1}^{n}(A_i - \hat{A})(B_i - \hat{B})}{\sqrt{\sum_{i=1}^{n}(A_i - \hat{A})^2}\sqrt{\sum_{i=1}^{n}(B_i - \hat{B})^2}}, \quad (3)$$

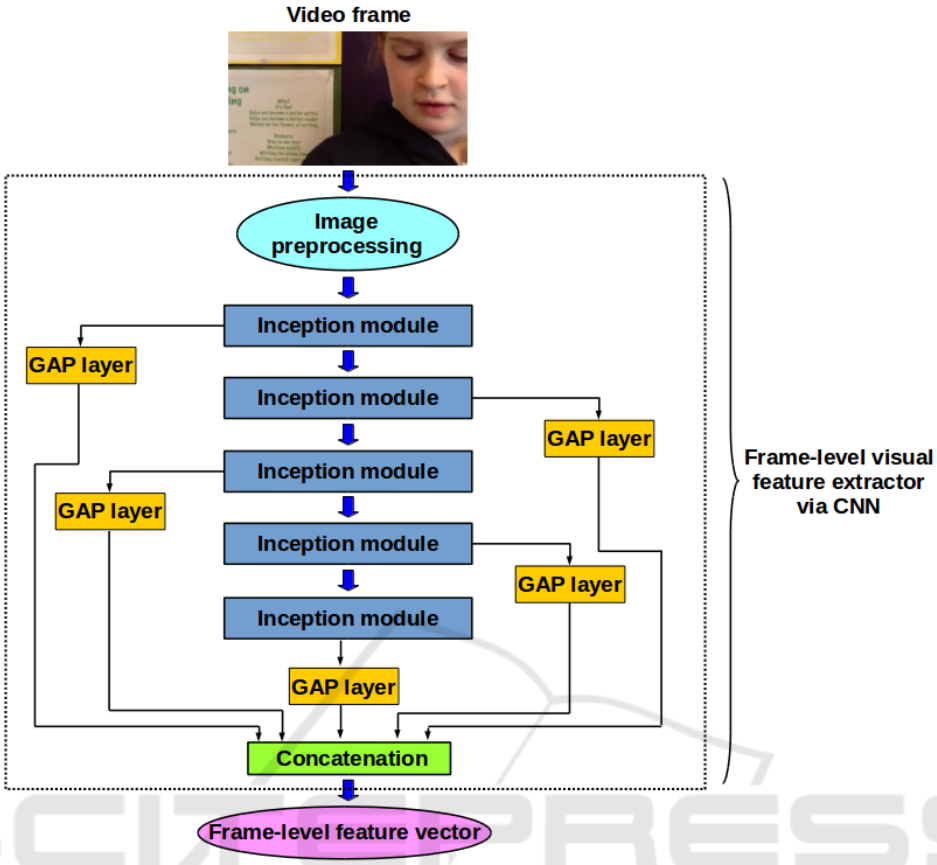


Figure 3: Frame-level feature extraction. A video frame is run through a pretrained CNN body containing Inception modules (GoogLeNet (Szegedy et al., 2015), Inception-v2, and Inception-v3 are considered in this study). Furthermore, global average pooling (GAP) layers are attached to each Inception module to extract resolution independent deep features at different abstraction levels. The features obtained from the Inception modules are concatenated to create the frame-level feature vector.

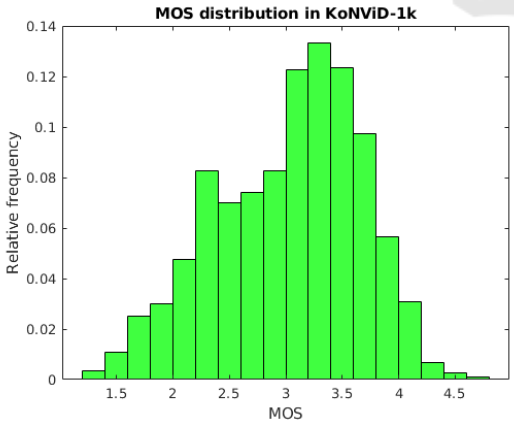


Figure 4: MOS distribution in KoNViD-1k (Hosu et al., 2017). In this database, $MOS = 1$ represents the lowest video quality, while $MOS = 5$ stands for the highest video quality.

where $\hat{A}$ and $\hat{B}$ denote the middle ranks of set A and B, respectively. The range of values for PLCC and SROCC is $[-1;1]$. Furthermore, the closer the value is to 1, the better the correlation between ground-truth and predicted perceptual quality scores. More specifically, PLCC is a number between -1 and $+1$ that indicates the extent to which two variables are linearly correlated, while SROCC is the non-parametric version of the PLCC and measures the strength and direction of association between two ranked variables.

4.2 Implementation Details

The proposed models were implemented with the help of Keras[1] (Chollet et al., 2015) with a TensorFlow backend (Abadi et al., 2016). The intra-frames were extracted using the free and open-source *FFmpeg*[2] using *ffmpeg -i "$FILE_NAME" -vf "select='eq(pict_type,PICT_TYPE_I)'"* command. Further, the models were trained on a NVidia Geforce GTX 1080 GPU.

[1] https://keras.io/
[2] https://www.ffmpeg.org/

As mentioned above, KoNViD-1k (Hosu et al., 2017) database was used for training and testing. Specifically, 840 videos were selected randomly for the training set and 120 videos were selected randomly for the validation set. The remaining 240 videos were reserved to test the proposed method. Moreover, average PLCC and SROCC were measured over 20 random train-validation-test splits.

In all experiments, Adam optimizer (Kingma and Ba, 2014) was utilized for fine-tuning base CNN architectures with default parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Unlike stochastic gradient descent, Adam does not maintain a single learning rate for all weight updates. Instead, Adam adapts the learning rate by calculating an exponential moving average of the gradient and the squared gradient, while β_1 and β_2 control the decay rates of these moving averages. Specifically, the learning rate was set to 10^{-4} and divided by 10 when the validation error stopped improving. Furthermore, early stopping was applied to avoid overfitting. This means that training was stopped when the validation error showed no improvement despite the decimation of the learning rate. The proposed architecture was trained and tested on a PC with 8-core i7-7700K CPU and two NVidia TitanX GPUs.

4.3 Parameter Study

First, we conducted experiments to determine the optimal resolution of video frames for fine-tuning base CNN architectures. Because we trained the proposed architecture with different resolution images, we were not able to train all models with exactly the same batch size because of the memory limitations of the GPU. Hence, the largest available 2^n batch size that fit on the GPU memory was used for each model.

First, the models were trained on the original resolution — 960×540 of KoNViD-1k[3] (Hosu et al., 2017). Second, the models were trained on images down-sampled by a factor of two using bilinear interpolation (480×270). Finally, we trained the models on those down-sampled images that correspond to the minimal input size of Inception-v3 — 299×299. Furthermore, the we measured the performance if all frames are used in the training and if only the so-called keyframes are used.

The results of the parameter study are summarized in Figures 5, 6, 7, and 8. Surprisingly, we found that models fine-tuned on half-sized video frames (480×270) achieved slightly better results than those fine-tuned on the original resolution video frames (960×540) or on the video frames with 299×299 resolution. One reason for this result could be that the applied pretrained CNNs are optimized for images with resolutions smaller than 960×540. Furthermore, we were only able to apply a batch size of two when fine-tuning on 960×540-sized images because of GPU memory limitations. Moreover, the deeper Inception-v3 base CNN yielded better results than GoogLeNet or Inception-v2. Our analysis also demonstrates that CNN fine-tuning improves both PLCC and SROCC significantly. Figure 5 summarizes the results without fine-tuning. Figure 6, 7, and 8 depict those results when the base CNNs were fine-tuned on 299×299, 480×270, and 960×540 sized video frames, respectively. As already mentioned, we report on average PLCC and SROCC values obtained by 5-fold cross-validation with 20 repetitions.

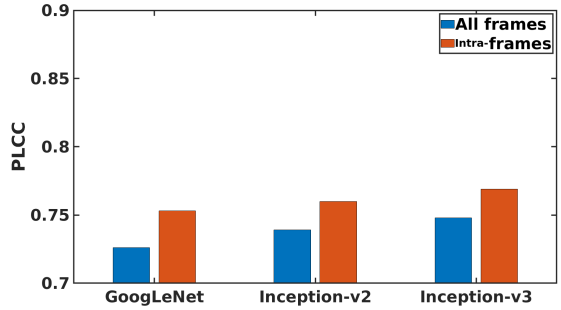
Our analysis demonstrated that models trained only on the so-called *intra-frames* outperform models trained on all frames. More specifically, considering only the intra-frames in the training process improves the performance by $0.05 - 0.1$ both in terms of PLCC and SROCC. Furthermore, the computational time of a video-level feature vector lasts for approximately 185 secs if we consider all frames in a video sequence for KoNViD-1k. In contrast, the computational time fluctuates between 1.5 - 75 secs (videos may contain different number of intra-frames) given the experimental setup described in Section 4.2. As a consequence, we have shown that processing all frames is not necessary and a selection strategy is able to improve performance and decrease computational time.

On the whole, the best-performing architecture relied on Inception-v3 fine-tuned on half-sized (480×270) video frames. Further, intra-frames are selected in the training and testing procedure. In the next subsection, we refer to this model as *MultiInception*, and it is compared with other state-of-the-art algorithms.

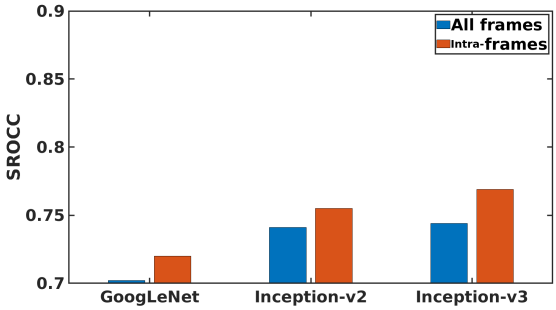
4.4 Comparison to the State-of-the-Art

We compared the proposed architecture to seven state-of-the-art NR-VQA methods (V-CORNIA (Ye et al., 2012), V-NIQE (Mittal et al., 2012b), V-BLIINDS (Saad et al., 2014), VIIDEO (Mittal et al., 2015), Inception-v3 + avg. pooling (Varga, 2019), TLVQM (Korhonen, 2019), VSFA (Li et al., 2019)) whose source code available online. Furthermore, we reimplemented FC Model (Men et al., 2017). All of them were trained using those setup that we applied to our proposed method, that is 960 videos (80%) were used for training and validation purposes, while the remaining 240 videos (20%) were utilized in the testing process. Moreover, the average PLCC and SROCC values were measured over 20 random train-

[3] http://database.mmsp-kn.de/konvid-1k-database.html

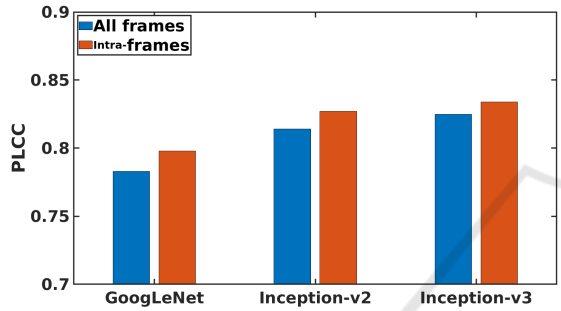


(a) Avg. PLCC obtained by 5-fold cross-validation with 20 repetitions.

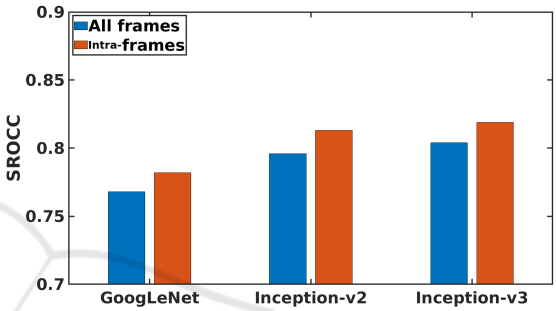


(b) Avg. SROCC obtained by 5-fold cross-validation with 20 repetitions.

Figure 5: CNN base architecture comparison without fine-tuning.

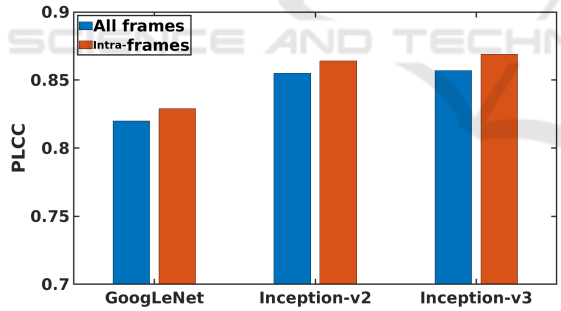


(a) Avg. PLCC obtained by 5-fold cross-validation with 20 repetitions.

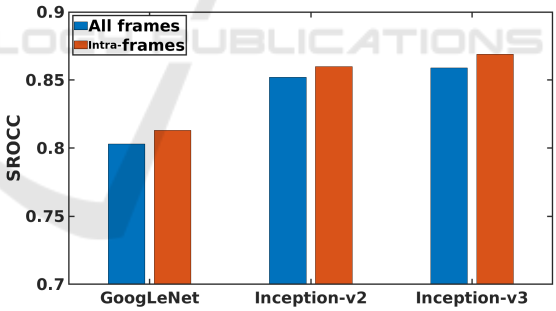


(b) Avg. SROCC obtained by 5-fold cross-validation with 20 repetitions.

Figure 6: CNN base architecture comparison with fine-tuning on resolution 299×299.



(a) Avg. PLCC obtained by 5-fold cross-validation with 20 repetitions.

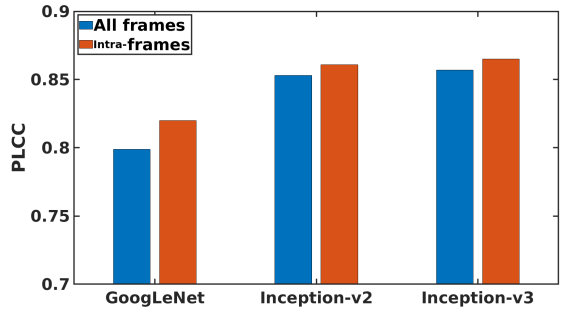


(b) Avg. SROCC obtained by 5-fold cross-validation with 20 repetitions.

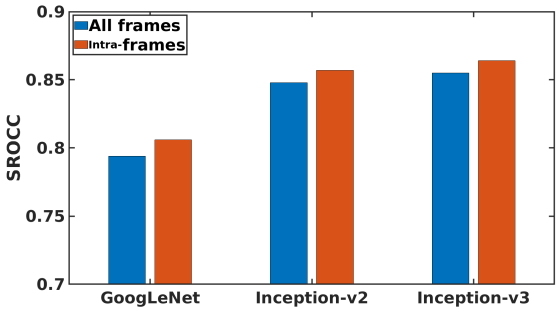
Figure 7: CNN base architecture comparison with fine-tuning on resolution 480×270.

validation-test splits. The results of the comparison are summarized in Table 1.

From these results, it can be concluded that the proposed model is able to outperform state-of-the-art algorithms. Specifically, the proposed method improves both PLCC and SROCC by approximately 0.05 compared to TLVQM (Korhonen, 2019), which is currently one of the best and recent methods proposed in the literature. Moreover, our method was the only one that produced results over 0.8 of PLCC and SROCC. We attribute this improvement to the fact that our method extracts resolution independent features from video frames at different levels of abstraction using powerful CNN architectures. This statement is supported by the observation that the proposed architecture achieves the state-of-the-art without fine-tuning as well. Namely, the application of fine-tuning (transfer learning) improves both PLCC and SROCC by approximately 0.06. The overall 20-run-results of our proposed method with fine-tuning are depicted in the form of box plots in Figure 9.



(a) Avg. PLCC obtained by 5-fold cross-validation with 20 repetitions.



(b) Avg. SROCC obtained by 5-fold cross-validation with 20 repetitions.

Figure 8: CNN base architecture comparison with fine-tuning on resolution 960×540.

Table 1: Comparison of state-of-the-art NR-VQA algorithms measured on KoNViD-1k (Hosu et al., 2017). Average PLCC and SROCC values were measured over 20 random train-validation-test splits. Furthermore, the standard deviation values are given in parentheses. The best average PLCC and SROCC results are typed by **bold**.

Method	PLCC	SROCC
V-CORNIA (Ye et al., 2012)	0.586 (0.037)	0.591 (0.040)
V-NIQE (Mittal et al., 2012b)	0.543 (0.041)	0.545 (0.043)
V-BLIINDS (Saad et al., 2014)	0.567 (0.044)	0.578 (0.046)
VIIDEO (Mittal et al., 2015)	0.300 (0.051)	0.284 (0.049)
FC Model (Men et al., 2017)	0.496 (0.015)	0.473 (0.015)
Inception-v3 + avg. pooling (Varga, 2019)	0.783 (0.023)	0.790 (0.022)
TLVQM (Korhonen, 2019)	0.776 (0.019)	0.783 (0.020)
VSFA (Li et al., 2019)	0.760 (0.030)	0.768 (0.030)
MultiInception (without finetuning)	0.769 (0.025)	0.769 (0.025)
MultiInception	**0.828** (0.026)	**0.829** (0.025)

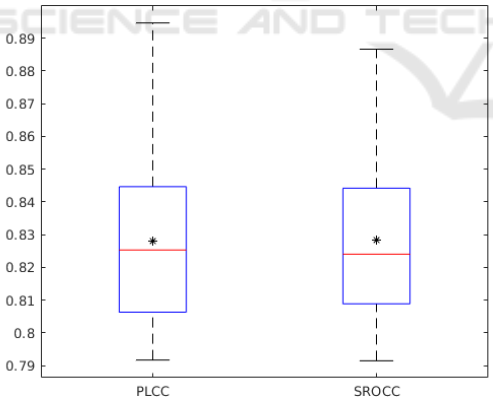


Figure 9: The overall 20-run-results of our proposed method with fine-tuning in the form of box plots. On each box, the red line indicates the median, the star indicates the mean, and the bottom and top edges of the box stand for the $25th$ abd $75th$ percentiles, respectively. The whiskers extend to the most extreme data points.

5 CONCLUSIONS

In this paper, we introduced a framework for NR-VQA relying on visual features extracted from pre-trained CNNs. Specifically, we presented a content-preserving feature extraction method which relies on the Inception modules of pretrained CNNs, such as GoogLeNet or Inception-v3. Unlike previous methods, patches are not taken from the input video frames but treated them as a whole. More specifically, the frame-level visual features were extracted from multiple Inception modules and pooled by a global average pooling layer together. This way, we incorporated both intermediate- and high-level deep representations to the frame-level feature vectors. Another contribution of this study was that we showed it is unnecessary to process all frames of a video to be evaluated. Unlike previous algorithms, we do not examine all frames. In contrast, only the so-called intra-frames are considered. We demonstrated that considering only intra-frames saves computational time and improves the performance significantly both in terms of PLCC and SROCC. Finally, we compared our method to six state-of-the-art NR-VQA methods. The proposed architecture surpassed the best state-of-the-art method by approximately in terms of PLCC and SROCC.

ACKNOWLEDGEMENTS

The author would like to show his gratitude to Professor Dietmar Saupe, Hanhe Lin, Vlad Hosu, Franz Hahn, Hui Men, and Mohsen Jenadeleh for sharing their knowledge in visual quality assessment and helping to use KoNViD-1k. The author would like to thank the anonymous reviewers for their helpful and constructive comments that greatly contributed to improving the final version of the paper.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.

Ahn, S. and Lee, S. (2018a). Deep blind video quality assessment based on temporal human perception. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 619–623. IEEE.

Ahn, S. and Lee, S. (2018b). No-reference video quality assessment based on convolutional neural network and human temporal behavior. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1513–1517. IEEE.

Bianco, S., Celona, L., Napoletano, P., and Schettini, R. (2018). On the use of deep learning for blind image quality assessment. *Signal, Image and Video Processing*, 12(2):355–362.

Borer, S. (2010). A model of jerkiness for temporal impairments in video transmission. In *2010 second international workshop on quality of multimedia experience (QoMEX)*, pages 218–223. IEEE.

Chollet, F. et al. (2015). Keras. `https://keras.io`.

Dendi, S. V. R., Krishnappa, G., and Channappayya, S. S. (2019). Full-reference video quality assessment using deep 3d convolutional neural networks. In *2019 National Conference on Communications (NCC)*, pages 1–5. IEEE.

Drucker, H., Burges, C. J., Kaufman, L., Smola, A. J., and Vapnik, V. (1997). Support vector regression machines. In *Advances in neural information processing systems*, pages 155–161.

Giannopoulos, M., Tsagkatakis, G., Blasi, S., Toutounchi, F., Mouchtaris, A., Tsakalides, P., Mrak, M., and Izquierdo, E. (2018). Convolutional neural networks for video quality assessment. *arXiv preprint arXiv:1809.10117*.

Habibzadeh, M., Jannesari, M., Rezaei, Z., Baharvand, H., and Totonchi, M. (2018). Automatic white blood cell classification using pre-trained deep learning models: Resnet and inception. In *Tenth International Conference on Machine Vision (ICMV 2017)*, volume 10696, page 1069612. International Society for Optics and Photonics.

Hosu, V., Hahn, F., Jenadeleh, M., Lin, H., Men, H., Szirányi, T., Li, S., and Saupe, D. (2017). The konstanz natural video database (konvid-1k). In *2017 Ninth international conference on quality of multimedia experience (QoMEX)*, pages 1–6. IEEE.

Ji, S., Xu, W., Yang, M., and Yu, K. (2012). 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., and Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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

Li, D., Jiang, T., and Jiang, M. (2019). Quality assessment of in-the-wild videos. In *In Proceedings of the 27th ACM International Conference on Multimedia*. ACM.

Liu, W., Duanmu, Z., and Wang, Z. (2018). End-to-end blind quality assessment of compressed videos using deep neural networks. In *ACM Multimedia*, pages 546–554.

Lu, Z., Jiang, X., and Kot, A. (2018). Deep coupled resnet for low-resolution face recognition. *IEEE Signal Processing Letters*, 25(4):526–530.

Men, H., Lin, H., and Saupe, D. (2017). Empirical evaluation of no-reference vqa methods on a natural video quality database. In *2017 Ninth international conference on quality of multimedia experience (QoMEX)*, pages 1–3. IEEE.

Men, H., Lin, H., and Saupe, D. (2018). Spatiotemporal feature combination model for no-reference video quality assessment. In *2018 Tenth international conference on quality of multimedia experience (QoMEX)*, pages 1–3. IEEE.

Mittal, A., Moorthy, A. K., and Bovik, A. C. (2012a). No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708.

Mittal, A., Saad, M. A., and Bovik, A. C. (2015). A completely blind video integrity oracle. *IEEE Transactions on Image Processing*, 25(1):289–300.

Mittal, A., Soundararajan, R., and Bovik, A. C. (2012b). Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212.

Pastrana-Vidal, R. R., Gicquel, J. C., Colomes, C., and Cherifi, H. (2004). Sporadic frame dropping impact on quality perception. In *Human Vision and Electronic Imaging IX*, volume 5292, pages 182–193. International Society for Optics and Photonics.

Reinhardt, R. (2010). *Video with Adobe Flash CS4 Professional Studio Techniques*. Adobe Press.

Saad, M. A. and Bovik, A. C. (2012). Blind quality assessment of videos using a model of natural scene statistics and motion coherency. In *2012 Conference Record of the Forty Sixth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, pages 332–336. IEEE.

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

Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., and Poggio, T. (2007). Robust object recognition with cortex-like mechanisms. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (3):411–426.

Shahid, M., Rossholm, A., Lövström, B., and Zepernick, H.-J. (2014). No-reference image and video quality assessment: a classification and review of recent approaches. *EURASIP Journal on image and Video Processing*, 2014(1):40.

Søgaard, J., Forchhammer, S., and Korhonen, J. (2015). No-reference video quality assessment using codec analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(10):1637–1650.

Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Thomee, B., Shamma, D. A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., and Li, L.-J. (2015). Yfcc100m: The new data in multimedia research. *arXiv preprint arXiv:1503.01817*.

Varga, D. (2019). No-reference video quality assessment based on the temporal pooling of deep features. *Neural Processing Letters*, pages 1–14.

Xu, J., Ye, P., Liu, Y., and Doermann, D. (2014). No-reference video quality assessment via feature learning. In *2014 IEEE international conference on image processing (ICIP)*, pages 491–495. IEEE.

Xu, Z., Yang, Y., and Hauptmann, A. G. (2015). A discriminative cnn video representation for event detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1798–1807.

Xue, Y., Erkin, B., and Wang, Y. (2014). A novel no-reference video quality metric for evaluating temporal jerkiness due to frame freezing. *IEEE Transactions on Multimedia*, 17(1):134–139.

Yammine, G., Wige, E., Simmet, F., Niederkorn, D., and Kaup, A. (2012). Blind frame freeze detection in coded videos. In *2012 Picture Coding Symposium*, pages 341–344. IEEE.

Ye, P., Kumar, J., Kang, L., and Doermann, D. (2012). Unsupervised feature learning framework for no-reference image quality assessment. In *2012 IEEE conference on computer vision and pattern recognition*, pages 1098–1105. IEEE.

You, J. and Korhonen, J. (2019). Deep neural networks for no-reference video quality assessment. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2349–2353. IEEE.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595.

Zhang, Z., Shi, H., and Wan, S. (2009). A novel blind measurement of blocking artifacts for h. 264/avc video. In *2009 Fifth International Conference on Image and Graphics*, pages 262–265. IEEE.