



# NDNetGaming - development of a no-reference deep CNN for gaming video quality prediction

Markus Utke<sup>1</sup> · Saman Zadtootaghaj<sup>1</sup>  · Steven Schmidt<sup>1</sup> · Sebastian Bosse<sup>2</sup> · Sebastian Möller<sup>1</sup>

Received: 9 January 2020 / Revised: 25 April 2020 / Accepted: 27 May 2020 /

Published online: 24 July 2020

© The Author(s) 2020

## Abstract

Gaming video streaming services are growing rapidly due to new services such as passive video streaming of gaming content, e.g. Twitch.tv, as well as cloud gaming, e.g. Nvidia GeForce NOW and Google Stadia. In contrast to traditional video content, gaming content has special characteristics such as extremely high and special motion patterns, synthetic content and repetitive content, which poses new opportunities for the design of machine learning-based models to outperform the state-of-the-art video and image quality approaches for this special computer generated content. In this paper, we train a Convolutional Neural Network (CNN) based on an objective quality model, VMAF, as ground truth and fine-tuned it based on subjective image quality ratings. In addition, we propose a new temporal pooling method to predict gaming video quality based on frame-level predictions. Finally, the paper also describes how an appropriate CNN architecture can be chosen and how well the model performs on different contents. Our result shows that among four popular network architectures that we investigated, DenseNet performs best for image quality assessment based on the training dataset. By training the last 57 convolutional layers of DenseNet based on VMAF values, we obtained a high performance model to predict VMAF of distorted frames of video games with a Spearman's Rank correlation (SRCC) of 0.945 and Root Mean Score Error (RMSE) of 7.07 on the image level, while achieving a higher performance on the video level leading to a SRCC of 0.967 and RMSE of 5.47 for the KUGVD dataset. Furthermore, we fine-tuned the model based on subjective quality ratings of images from gaming content which resulted in a SRCC of 0.93 and RMSE of 0.46 using one-hold-out cross validation. Finally, on the video level, using the proposed pooling method, the model achieves a very good performance indicated by a SRCC of 0.968 and RMSE of 0.30 for the used gaming video dataset.

**Keywords** Convolutional neural network · Gaming video · Quality assessment

---

✉ Saman Zadtootaghaj  
saman.zadtootaghaj@qu.tu-berlin.de

# 1 Introduction

The gaming industry has been one of the largest digital markets for decades and is rapidly growing due to emerging online services such as gaming video streaming, online gaming and cloud gaming (CG) services. While the game industry is growing, more complex games in terms of processing power are getting developed which requires players to update their end devices every few years in order to play the newest games. One solution for this is to move the heavy processes such as rendering to the cloud and cut the needs for the high-end hardware devices for customers. Cloud gaming is proposed to offer more flexibility to users by allowing them to play any game anywhere and on any type of device. Apart from processing power, cloud gaming benefits users by the platform independency and for game developers offers security to their products and promises a new market to increase their revenue. Besides cloud gaming, passive video streaming of gameplays become popular with hundreds of millions of viewers per year. Twitch.tv and YouTube Gaming are the two most popular services for passive video game streaming.

Quality prediction is a necessary process of any service provider to ensure the satisfaction of its customers. While subjective tests are the basis of any quality assessment of multimedia services, service providers are seeking objective methods<sup>1</sup> for predicting the quality, as subjective experiments are expensive and time-consuming. Depending on the level of access to the reference signal, signal-based image and video quality metrics can be divided into three classes: no-reference (NR), reduced-reference (RR), and full-reference (FR) metrics. For Quality of Experience (QoE) assessment of cloud gaming and passive video streaming services, NR metrics are of interest for service providers, since the reference signal is often not accessible for service providers or network operators.

The previous studies [5, 7] revealed a medium to low level performance of existing NR video quality models on gaming video datasets which motivated us to work on the development of a new NR quality metric for gaming video content. With the recent advancement in the machine learning domain and several use-cases of applied deep learning methods in the vision and multimedia area, we made a first attempt to build a novel machine learning-based model for video gaming quality assessment.

In this paper, we describe the development process of a Convolutional Neural Network (CNN) based NR video quality metric that predicts the quality for video of gaming content impaired by compression artefacts. We refer to the developed No-reference Deep CNN for Gaming video quality prediction as NDNNetGaming. Due to the limited availability of quality datasets for gaming content, obtaining a well-trained CNN on subjective ratings is challenging due to the high effort and costs of conducting subjective tests. Therefore, we overcome this limitation by using transfer learning [38].

The main idea of our work is to train a CNN on a large number of frames which are annotated based on a full-reference quality metric, Video Multimethod Assessment Fusion (VMAF) [27], and then retrain a few last layers of the pre-trained CNN on a smaller subjective image dataset of gaming content that is published together with the source code of proposed model (cf. Section 3.3.4). Finally, we build a pooling method using the temporal information to predict the video quality based on the frame-level quality predictions.

<sup>1</sup>We use the term objective method in this paper, as it is very common in the video quality community. However, it has to be noted that the methods still rely on subjective quality ratings, and thus are not purely objective.

With the aim to investigate how to efficiently train a CNN for quality prediction of gaming video content, we designed six research questions (RQ) to be addressed in this paper. The research questions are based on the lack of practical information in training and fine-tuning process of the CNN models for quality assessment task as well as the limitation of access to a sufficient large scale dataset for training of a CNN.

- RQ1: Which pre-trained CNN architecture performs the best among state-of-the-art models for image quality assessment?
- RQ2: How many CNN layers are required to be trained in order to predict VMAF or Mean Opinion Score (MOS) values with a high precision?
- RQ3: Using our approach, how much data in terms of number of frames and patches is roughly required for the metric development?
- RQ4: How to train a CNN for image quality prediction, while considering a limited number of subjective ratings?
- RQ5: Are machine learning-based quality assessment methods suitable for computer generated imagery?
- RQ6: How to pool the frame quality prediction to predict the video quality?

In order to answer these research questions while considering the high effort of creating very large subjective datasets and while aiming to justify an appropriate selection of a CNN architecture, we retrained the most well-known pre-trained CNN architectures purely based on VMAF values as a ground truth to get some insight on the selection of the architecture and to estimate a suitable number of required frames and layers.

The remainder of the paper is structured as follows. In Section 2, we give a short overview of the related work. Data preparation and the datasets that are used in this work are described in Section 3. Section 4 explains the differences between gaming content and non-gaming content. The CNN framework, training and testing phases as well as results of the model at each level of the training are discussed in Section 5. Section 6 presents the evaluation of the metric on different datasets, gaming and non-gaming datasets. Finally, the paper concludes in Section 7 with a discussion on the findings and future work.

## 2 Related work

Within the last decades, we have been witness to a great number of research works with respect to pseudo-objective image and video quality assessment. In this section, we give a short overview of deep learning based quality models as well as metrics that are developed specifically for computer generated content.

The performances of state of the art video and image quality metrics on gaming videos were investigated in [5, 7]. The authors report a high correlation of VMAF with Mean Opinion Scores (MOS) while most of the NR metrics perform quite poorly. With respect to gaming content, to the best knowledge of the authors, only two NR metrics are developed. Zadtootaghaj et al. proposed a NR machine learning-based video quality metric for gaming content, named NR-GVQM, that is trained based on low level image features with the aim of predicting VMAF without having access to a reference video [39]. Another NR pixel-based video quality metric for gaming QoE, called Nofu, was proposed by Goering et al. [15]. Nofu is also a machine learning metric that extracts low level features of which some are hand-crafted by the authors and some are taken from the state of the art. Nofu as a NR metric has a slightly higher performance compared to VMAF as FR metric on the GamingVideoSET [6]. Two other NR metrics for gaming content are proposed in [3], named

NR-GVSQI and NR-GVSQE. These two proposed models are designed using supervised learning algorithms based on MOS and VMAF values as the target output. NR-GVSQE used a similar approach as NR-GVQM, which gets trained based on the frame level as input features and VMAF values as target, while NR-GVSQI is trained based on MOS values. Recent years have witnessed a growing interest of deep neural networks (DNN) based models for video and image quality predictions. While, in the field of multimedia quality assessment, several works for both contents have been published, more models have been proposed for the image content than videos content for two reasons. First, the video quality datasets are relatively small in terms of number of annotated data compared to image quality datasets due to timely expensive quality assessment of videos compared to image content, which leads to difficulties in training an accurate model due to insufficient data. In addition, many DNN based quality model used transfer learning methods to overcome the shortage of sufficient data for training. Transfer learning is “used to improve a learner from one domain by transferring information from a related domain” [38]. In plain words, transfer learning is referred to a process that uses pre-trained DNN (which could be used for problems in other domains) and retrain a few last layers of DNN based on the new problem. However, transfer learning is more suitable for image content than video content, as more pre-trained models are available for image data, e.g. ImageNET dataset. Bosse et al. [9] presented a neural network-based approach to build FR and NR image quality metrics which are inspired by VGG16 by increasing the depth of the CNN with ten convolutional layers. In order to reduce the complexity of repeated quality estimation, i.e. for optimization, a deep neural network was used to extract an image and distortion type specific distortion sensitivity from the reference image for perceptual meaningful adaptation of a low complex FR quality model [8]. Rakesh et al. [29] proposed a hybrid NR image quality metric which is developed after training and testing a few pre-trained CNNs and extending them by adding signal-based features from state of the art models.

### 3 Datasets

In this section, we explain which datasets we used for the development of NDNetGaming.

#### 3.1 Introduction to datasets

With the aim of building a CNN based quality model for gaming content, the first limitation that we faced was the unavailability of a relatively large quality annotated dataset for gaming video content. Due to this limitation, we decided to train the model on the frame level based on a pseudo-objective quality metric, VMAF, and later fine-tune it based on subjective ratings of images.

By using VMAF as a proxy of perceptual quality, a larger database can be generated compared to psychophysical tests. This allows to train networks with a higher number of parameters, i.e. deeper networks as well as allowing the network learn different types of image distortions such as blockiness and blurriness. This is because we can increase the size of the dataset by simply recording and encoding more gaming video sequences and then calculating VMAF scores for the encoded frames without conducting large scale subjective tests. The selection of VMAF, among many other quality metrics, is based on its high performance shown in previous studies in [5, 7].

To the best of the authors knowledge, there exists no available gaming dataset with frame-level quality labels. Therefore, we decided to build an image quality dataset of gaming content in which we introduced typical video artifacts that can be seen on gaming streaming services. The details on the new gaming image dataset are described in Section 3.3.

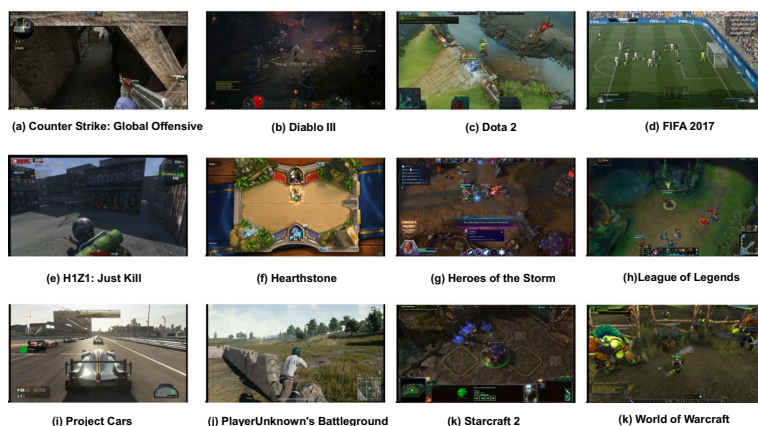
### 3.2 Gaming video datasets

**GamingVideoSET** The GamingVideoSET [6], which we refer to as GVSET in the remainder of the paper, consists of 24 source video sequences from 12 different games (two sequences per game), as shown in Fig. 1. The videos are encoded using H.264/MPEG-AVC under different resolution-bitrate pairs. All videos have a duration of 30 seconds with a framerate of 30 fps. Out of 24 source video sequences, subjective ratings are provided for 6 sequences. For all encoded videos, around 400 sequences, VMAF values are available for each frame.

**KUGVD** The KUGVD dataset similarly consists of six high-quality raw gaming videos of 30 seconds duration, 1080p resolution, and a framerate of 30 fps [3]. Subjective quality ratings exist for 90 video sequences obtained by encoding the raw videos using H.264/MPEG-AVC codec standard in 15 different resolution-bitrate pairs (three resolution, five bitrates each). Additionally, for encoded videos using different resolution-bitrate pairs, also VMAF ratings, among other pseudo-objective metrics, are available.

### 3.3 Gaming image dataset

In order to build a gaming image quality metric as the first step of our framework, due to the lack of any subjective frame-level quality dataset of gaming content, we decided to create a new image quality dataset of gaming scenes. We refer to this dataset in the following as Gaming Image DataSET (GISET). For the creation of the dataset, we conducted a subjective test with 20 participants. For the assessment of the image quality, a discrete 5-point scale, Hidden Reference Absolute Category Rating (HR-ACR) method, was used [18].



**Fig. 1** Screenshots of the gaming videos available in the GamingVideoSET

### 3.3.1 Test setup

For the subjective tests, the light conditions in the test rooms as well as the viewing distance and position of participants were consistent for the experiment. In general, ITU-T Rec. P.910 [20] and ITU-R BT.500-14 [19] were considered. Participants were offered an adjustable chair and table to sit in a proper position (the chair allows participants to have their feet flat on the floor and their knees equal to, or slightly lower than, their hips). The viewing distance ( $D$ ) was equal to three times the picture height ( $H$ ). The test duration did not exceed maximum of 45 minutes. For the image presentation an LCD monitor, 24" display for standard gaming, HD1080 resolution was used. Participants were asked to sit in the center of the image display at the specified viewing distance. The brightness was adjusted according to ITU-R recommendation BT.500-14. The gamma was set to 2.2. and color temperature to 6500 K (default value on most LCDs).

### 3.3.2 Selection of content

With the aim to only keep the relevant distortions of H.264/AVC compression for gaming content, we decided to extract the frames directly from an existing gaming video dataset, GVSET. Thus, a dataset was built consisting of 164 frames chosen from GVSET in which for each source image, we selected three encoded images with different levels of quality. Among the three encoded images, one was extracted from a video with low bitrate level, aiming at the blockiness artifacts, and one with lower resolutions than source image with the aim to trigger the blur in the frame and finally one with a mixture of both artifacts. It has to be noted that in GVSET all encoded videos with resolution lower than 1080p are upscaled to 1080p using bicubic filter which leads to blur artefact. The source frames are selected from multiple video games from different parts of each sequence. We tried to choose an appropriate distribution of distortions by considering their VMAF quality levels. The level of distortion for each frame was selected based on the VMAF values within a specific range from 20 to 80. The upper bound of VMAF of 80, was chosen based on the our previous works which found no significant difference between reference condition and encoded videos with VMAF value above 90 in subjective ratings [6]. In addition, we tried to include all types of frames (I, B, P), since in I-frames, typically blockiness is not dominant compared to B and P frames.

### 3.3.3 Test methodology

In order to get insights on the influence of encoding parameters on the perceived video quality, we assess different types of degradation using the dimension-based subjective quality evaluation that is designed for video telephony [32], describing the identification of relevant perceptual video quality dimensions. While this method was proposed for a video-telephony experiment, we found it a suitable method to identify the types of degradation for image content as well. Therefore, we added two questions in order to assess Fragmentation and Unclearness in addition to overall image quality. Fragmentation is defined by synonyms such as fallen apart, torn and blocky which was only triggered by a low bitrate in our dataset. Unclearness is defined by synonyms such as unclear, blur and smeared image which is triggered by lower resolution due to upscaling artefacts.

Each dimension is explained to the participants in an introduction in a written form using describing adjectives and in form of example videos. The rating scales are designed based on Likert-Scales, using the antonym pairs to describe the range of the scales. However, we

used the 5-point ACR scale for video quality, fragmentation and unclarity, in order to be consistent with typical video quality tests as well as with the ratings of GVSET.

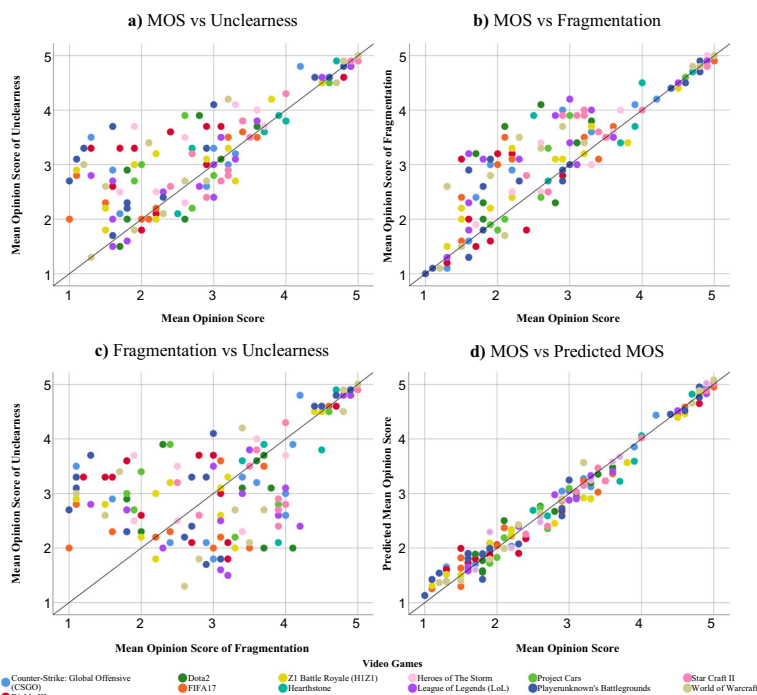
The scatter plots of overall quality, fragmentation and unclarity are shown in Fig. 2, which reveals a proper distribution of ratings from both sides of the scale which plays a vital role for the training phase. If the subjective ratings distribution would be skewed to one side of scale then we would have bias on the training process. One of our interesting observation was high difference between Fragmentation and Unclearness ratings as it can be seen in Fig. 2c. It indicates that participants can clearly differentiate between blockiness and unclarity dimensions and their cause of distortion.

As mentioned earlier, we assessed two other dimensions in addition to overall video quality (VQ), video fragmentation (VF) and video unclarity (VU). We fit a multilinear regression model based on the predictors, video fragmentation and video unclarity, to predict the overall video quality. The multidimensional model can predict the overall video quality with a Pearson Correlation Coefficient (PCC) of 0.98 and RSE of 0.154 as shown in (1). These dimensions can be used as diagnosis approach to spot the reasons behind the low video quality.

$$VQ_{Estimated} = -1.073 + 0.657 * VF + 0.573 * VU \quad (1)$$

### 3.3.4 Public dataset

All frames used in the dataset are provided in the shared folder that can be accessed through the following link:



**Fig. 2** The scatter plot of ratings distribution for different quality dimensions as well as Video quality prediction

<https://github.com/stootaghaj/GISET>

Four folders are created in the shared link as follows:

<b>MOS:</b>	Mean Opinion Scores for each frame is provided in this folder.
<b>Plots and Metrics:</b>	The results of objective metrics are provided together with some scatter plots of each metric and subjective results
<b>Materials:</b>	The reference images together with encoded videos in png format at 1080p resolution are provided in this folder . The raw reference images are extracted from raw recorded videos format. Based on the name of the image, readers can identify the video from GVSET and the position of the frame in the video that is extracted.

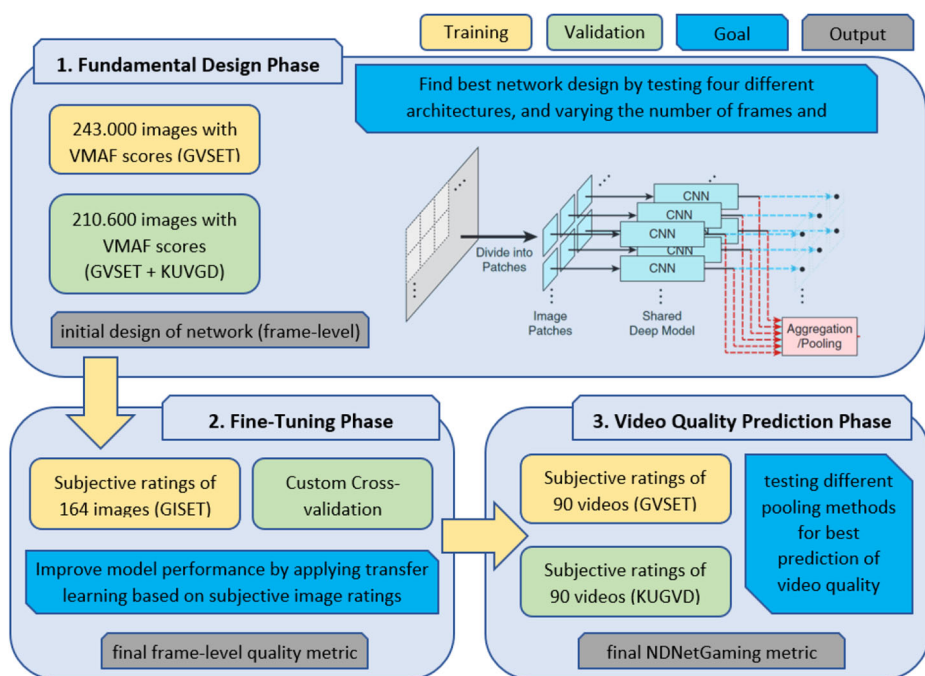
## 4 Gaming content characteristics

In contrast to traditional video content, gaming content has special characteristics such as extremely high motion for some games, special motion patterns, synthetic content, and repetitive content, which makes the state-of-the-art video and image quality metrics perform a bit weaker for this special computer generated content [4, 40]. Video games are usually created based on a pool of limited objects that appear in different scenes of a game. Therefore, there is a high similarity of different scenes of a particular game in respect to the spatial domain. In addition, for many video games, due to the same design style across the games, a game shares similar visual features such as background scene, color diversity, and pattern of motion [40]. Finally, for many games, the game world is small and similar scenes are repeated several times. These features make the gaming content a perfect target for training machine learning-based quality models as there is a high similarity in both temporal and spatial domain within the same game. Therefore, we can train a model for some specific types of games with higher accuracy compared to state-of-the-art metrics.

## 5 Framework and results

With the aim to develop a CNN based quality model, as discussed in Section 3, a large annotated quality dataset was required. However, due to the very limited number of available gaming video/image quality datasets, we were not able to rely on already collected subjective data and had to find an alternative way to overcome this issue. In addition, taking a video instead of a single image as an input to a CNN comes with limitations such as higher requirements of data and costly processing power. Therefore, we designed the model in three phases. In the first phase, we allow the model to learn typical video compression artifacts based on an objective quality metric on the frame-level for the sake of simplicity. The second phase was designed to fine-tune the model based on subjective ratings of image quality since any objective quality metric comes with inaccuracy. In the third phase, we pool the frame-level predictions to a video quality score.

The three phases of the proposed procedure are visualized in Fig. 3. In the first phase, the fundamental design phase, we addressed RQ1, RQ2, and RQ3 by investigating the suitability of four different, but well-known, CNN architectures aiming at providing an objective quality metric on the frame-level. While for the training a large part of the GVSET was used, the remaining parts and images from the KUGVD were used for the validation phase.



**Fig. 3** Procedure of NDNetGaming metric development

Here, also the number of required patches and frames to derive a well-performing metric was evaluated. In the second phase, the fine-tuning phase, the best-performing CNN from the first phase was improved while considering subjective ratings of 164 images of the GISET. Therefore, we froze early layers and retrained a few last layers of the CNN again based on the MOS ratings (transfer learning). In the last phase, we aimed for developing the final NDNetGaming metric predicting gaming video quality. Thus, we tried three different pooling methods to fit the frame-level prediction to the video level MOS values. Therefore, subjective video quality ratings of the GVSET (training) and several datasets, among them also 90 rated video stimuli from the KUGVD (validation) were used. The procedure is explained in the following sections in more detail. The source code of this project can be found in the following link: <https://github.com/stootaghaj/NDNetGaming>.

## 5.1 Fundamental design phase

To train our networks, we used VMAF values as a ground truth for the image quality of all frames involved in this phase. With the aim of training and validating the frame-level model based on the VMAF values, the GVSET and KUGVD were used.

All frames and corresponding VMAF values for those 18 source video sequences (and their encoded version using 15 different resolution-bitrate pairs) from the GVSET were selected and used in the training process of the model, for which no subjective video quality ratings are available. Thus, a total of 243,000 frames were used for the training.

For the validation of each investigated CNN, all frames and VMAF values of the encoded videos of the remaining six source video sequences were selected. In addition, we used all frames of the 144 encoded videos and their VMAF values of the KUGVD dataset [3] for

evaluating the model in each step to avoid any bias from the training. This leads to a total of 210.600 frames for the validation.

### 5.1.1 Comparison of CNN architectures

Using Keras [12], a high level neural network library, different model architectures are available along with their pre-trained weights on the ImageNet database [30]. We chose four popular architectures, DenseNet-121 [17], ResNet50 [16], Xception [13], and MobileNetV2 [31], and compared their performances for our dataset. The problem with a fair comparison of CNN architectures is that each architecture has a different number of parameters and layers as well as different design styles. We decided to compare them by the percentage of the networks parameters that were involved in the training, c.f. Table 1, assuming that a decent architecture could get trained by a smaller percentage of network.

Since the content of our database is artificial, the similarity to the ImageNet content can vary a lot for different games. Therefore, it is difficult to decide how many layers of each network have to be retrained. For a fair comparison, we used the total number of trainable parameters of each architecture and comparing it with the results when training only on the last 25%, 50% and 75% of parameters. The results are summarized in Table 1. Since we used the pre-trained networks that are initially trained for a classification task, for each of these architectures the fully connected layer with multiple output neurons at the end of network was removed in order to allow the model to get trained for regression task. In addition, we added one dense layer consisting of only one output neuron with linear activation. The output of the network was directly compared to the actual VMAF values of the validation set that consists of different videos that were not used in training session, and are encoded under various bit-resolution pairs settings. Because of the large size of the frames ( $1920 \times 1080$ ) in the dataset, we could not efficiently train the network on the images directly. Hence, we cropped random patches of size  $299 \times 299$  from the frames we wanted to train on. This was done in parallel to the training, such that in each epoch a new random patch of each image was chosen.

Based on the result presented in Table 1, it can be observed that ResNet50 and DenseNet-121 deliver the best results among the four architectures. Since DenseNet-121 has an advantage in terms of numbers of parameters over ResNet50, we decided to use DenseNet-121 for all following investigations. However, it should be noted that for an extensive evaluation, multiple hyperparameter settings should have been compared for every architecture. Also, for this comparison every training was only done once per configuration, so the actual average numbers could vary slightly.

### 5.1.2 Required number of layers for training

The pre-trained DenseNet-121 is trained based on a huge number of annotated images (over 14 million images) which is almost 100 times larger than our whole training dataset. In

**Table 1** RMSE on the GamingVideoSET for four different network architectures retraining only 25%, 50% or 75% of total trainable weights

	MobileNetV2	DenseNet-121	Xception	ResNet50
25%	9.59	7.58	7.33	7.60
50%	7.98	6.84	7.25	7.34
75%	7.34	6.74	7.29	6.71

addition, the early layers of a CNN, mostly basic features are learned by the models such as edges, shapes, corners and intensity which can be shared across different types of tasks. However, in the later layers of a CNN mostly features linked to task/application would be learned. Therefore, depends on the size of training data and diversity of content, there is an optimal point that the model is required to be retrained up to that point. In other words, training the whole DenseNet-121 model based on the our dataset could lead the model to learn weaker in early layers due to lack of sufficient data or diversity of content.

To further investigate how many layers should be retrained for DenseNet-121, we trained differently sized parts of the network on our dataset and compared the performance. The DenseNet-121 architecture consists of four blocks, each containing between 12 and 48 convolutional layers. Table 2 shows a comparison of the results after training of the model multiple time by increased step size of half dense blocks in each training iteration. We observe that the results in terms of Root Mean Score Error (RMSE) get better when more layers are used in the training process. However, this effect plateaus when we reach 57 layers and even inverts when more than 107 layers are trained. The Spearman’s Rank Correlation Coefficient (SRCC) does not vary much, except for the case where only the dense layer is trained.

5.1.3 Patch selection for training

One of the typical problems when using CNNs in image quality assessment is how to select the patches in the training and testing process of the model, as most of the common architectures have a fixed input patch size. Therefore, we first investigated if only cropping the center of images would be a better choice for training the model compared to randomized cropping during the training of the CNN, given the fact that users tend to look at the center of images [23]. Our results revealed that center crops cannot improve the performance of training compared to single random crops, while taking multiple random patches from each frame in the training phase achieves much higher performance.

Furthermore, we decided to train the model based on a smaller number of frames. Since the training dataset consists of over 300,000 frames and consecutive frames can be very similar, only every  $n$ -th frame from every video is used for training. In order to find a suitable step size  $n$ , a very high number was chosen ( $n = 400$ ) and then lowered step by step to find the point, where the performance of the model stops improving. The analysis showed that  $n = 13$  is the ideal threshold for our dataset in which lowering down the step size would not improve the result. The results are summarized in Table 3.

**Table 2** RMSE and SRCC for different choices of the number of convolutional layers

Dense blocks	Number of layers	Number of weights	RMSE	SRCC
4	120	7039 k	8.11	0.925
3 1/2	113	6878 k	7.02	0.942
3	107	6657 k	6.74	0.945
2 1/2	94	6268 k	6.77	0.946
2	82	5594 k	6.84	0.942
1 1/2	57	4461 k	6.82	0.946
1	33	2191 k	7.22	0.939
1/2	16	1233 k	7.39	0.936
0	0	1 k	10.60	0.870

**Table 3** RMSE and R-squared for different interval size between two selected consecutive frames in the dataset

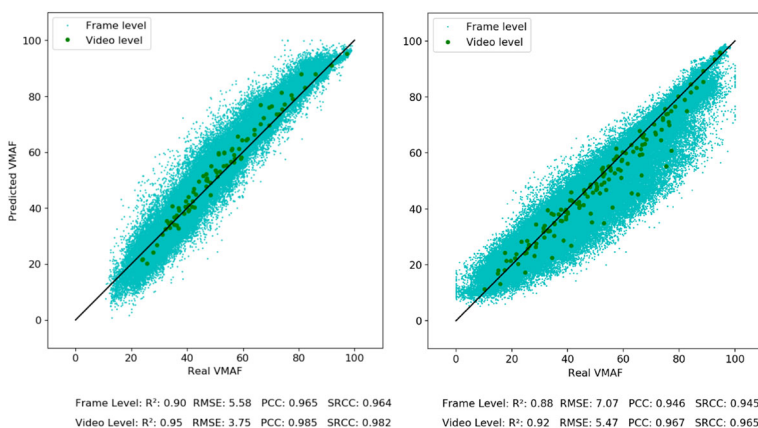
$n$ -th frame interval	RMSE	R-squared
3	6.54	0.87
7	6.65	0.88
13	6.39	0.90
27	6.67	0.88
53	6.95	0.88
103	6.90	0.88
203	6.94	0.87
403	7.02	0.88

### 5.1.4 The final model for VMAF prediction

To build the model that can predict the VMAF values on the image level, we retrained DenseNet-121 after replacing the fully connected layer with a dense layer, taking 57 layers for training while using every 13th frame. The model scores an RMSE of 6.19 with a PCC of 0.954 on frame level and an RMSE of 2.62 and PCC of 0.96 on the video level, using average pooling. Figure 4 shows the scatter plot of predicted and calculated VMAF values pooling from local to global quality on the validation dataset.

## 5.2 Fine-tuning phase

As the last step to obtain an image quality model for gaming content we applied transfer learning on the model that was trained on VMAF values to further fine-tune the model using the subjective image quality ratings. Since the number of MOS-labeled images are limited, we took 13 patches from each image, with a special pattern shown in Figure 5 top right side, in the training phase. It has to be noted that we did not use the MOS for training but the Differential MOS (DMOS) according to ITU-T recommendation P.913 [21].



**Fig. 4** Scatter plot of actual VMAF and predicted VMAF values on frame and video level of GVSET (*left*) and KUGVD (*right*) datasets

Due to the before mentioned limited number of frames for this fine-tuning step, we tried to improve the learning process by taking multiple patches from a frame for the quality estimation, instead of just one, and averaged their predictions. We designed specific patterns for different numbers of patches to be taken from a frame to get a better representation of the image than by just taking patches randomly. Figure 5 shows some example patterns and compares them to a random patch choice. Since there are often special elements like texts or maps in the corners of a gaming video, we believe that these parts are important for most of the games, which is why we included the four corners in all patterns. Additionally the center patch will be used in all patterns, since participants in image tests tend to look at the center of an image [23].

Table 4 compares the results for different numbers of patches using the patterns we designed. The results for multiple patches are much better than with only one patch per frame. However it is difficult to conclude on the optimum number of patches because there is no clear trend. This is probably due to the small size of the image data set, GISET.

Due to the small set of images in the training set, we employed a leave-one-out cross-validation where for every iteration of training the network, we kept one game completely out of training process (reference video together with all encoded videos of that game) and evaluate the model based on the holdout game. This process repeats twelve times for every game in the GISET. Based on the custom cross-validation we obtained an aggregated (after testing all games) RMSE of 0.354 and PCC of 0.959.

During the training, one quarter of the frames is used for validation. However before the final model will be tested on the video level, it will be trained on all the available subjective data.

### 5.3 Local quality predictions

In this section we visualize the local predictions of the model to get insights into the function of inter-mediate feature layers and the operation of the model. To calculate the local predictions, we built a new network with the full frame as an input (size  $1080 \times 1920$ ) but with the same weights for the convolutional layers as our original model. Additionally, we skipped the global average layer between the convolutional and dense layer of our CNN architecture. Usually, the dense layer expects 1024 numbers as its input and calculates one prediction as the output. When skipping the global averaging layer it instead gets  $1024 \times 33 \times 60$  matrices from the convolutional part of the network. We applied the dense part of the model element-wise to these matrices in order to get the local prediction matrix. From

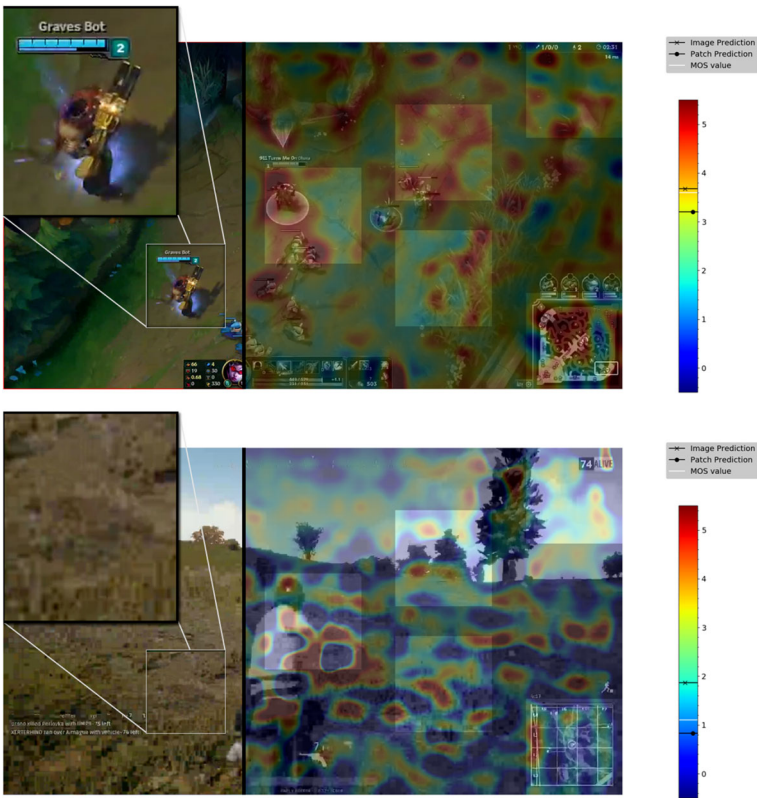


**Fig. 5** Five, nine and thirteen patches chosen with a pattern (*above*) and randomly (*below*)

**Table 4** RMSE and SRCC for different numbers of patches used for testing the model

Number of patches	RMSE	SRCC
1	0.481	0.894
3	0.413	0.944
5	0.390	0.953
7	0.374	0.957
9	0.380	0.954
11	0.381	0.958
13	0.377	0.953

the local quality predictions of many distorted images, we observed that the model has trouble to distinguish between the edges from blockiness and real edges of an image. Therefore, when it deals with images with high blockiness (fragmentation), it results in high variation of low and high local quality prediction as it can be seen in Fig. 6 for the game PlayerUnknown’s Battlegrounds. However, on average the image quality prediction is not deviated much, RMSE of 0.43 for images with blockiness degradation (1080p resolution and bitrate lower than 2 Mbps).



**Fig. 6** Local quality predictions for one frame of League of Legends (*top side*) and PlayerUnknown’s Battlegrounds (*bottom side*)

While there was a high variation of local quality predictions for blockiness artifacts, the model performs more consistent when dealing with blur as it can be seen in Figure 6 for the game League of Legends. We observed that our model tends to predict the quality higher for regions with high texture complexity which could be also in-line with human perception as image distortion can be masked in high texture region of an image.

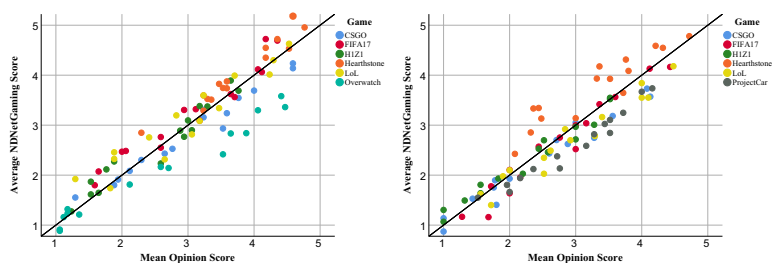
## 5.4 Video quality prediction phase

Predicting video quality based on an image quality metric is typically done by average pooling of the frame level predictions. While several pooling methods have been studied for gaming content, [7], and non-gaming content, [33], no significant improvement compared to average pooling method has been observed. However, it has been shown that participants tend to rate the image quality worse than video quality due to cognitive load as well as temporal masking effect [7, 35]. Temporal masking is one of the important aspect of human visual system (HVS), which has proven to an impact on perception of video artifacts. Choi et al., [11], analyzed the influence of motion on the performance of image level quality metrics after dividing LIVE VQA database into two subsets of low-level motions and high-level motion contents. Their results revealed that many frame-based quality metrics such as PSNR, perform poor in case of high-level motion content. We had similar observation, since the average pooled video quality prediction of our metric for high-level motion content, Overwatch, was lower than low-level motion content as shown in Fig. 7.

Due to the temporal masking effect, the widely used average pooling method would not lead us achieving the highest video quality prediction, specially for content with high temporal complexity. Therefore, we aim at pooling the image level prediction based on the temporal activities of the video. Thus, we assign higher weights for the contents with lower temporal complexity considering the temporal masking effect.

In order to measure the temporal complexity, the motion or optical flow pattern might be a good choice. However, the motion estimation could get affected significantly by compression distortion such as blur and blockiness. Therefore, we calculate the temporal complexity (TC) based on the difference between frames, similar to TI [18] but with a minor modification.

We calculate the per frame TI values and ignore the extreme values by using exponentially weighted moving average (EWMA). Then we calculate the Inverse Probability weighting (IP weighting) of EWMA results, that gives higher probability to small values compared to high values. The Inverse Probability weighting will be used to give higher weights to the low motion part of the game compared to high motion frames.



**Fig. 7** Scatter plot of averaged pooled video prediction of NDNetGaming and MOS for KUGVD (*left side*) and GVSET (*right side*)

$$ewma_{TI} = smooth_{ewma}(std_{space}[M_n(i, j)]) \quad (2)$$

$$weights_{frame} = ewma_{TI} / sum_{time}[ewma_{TI}] \quad (3)$$

$$inverse_{weights} = \frac{(1 - P(F = 1))}{1 - P(F = 1|W = w)} \quad (4)$$

where  $M_n(i, j)$  is the difference of pixel values between two adjacent frames considering only the luminance plane, and  $smooth_{ewma}$  is the exponentially weighted moving average function. The weighting average is useful to pool the quality with more weight on low temporal complex frames. However, these weights are only considering the local temporal complexity of frames in a video and not between video sequences. In order to take into account the difference between the temporal activity of video games, we used the TC values for each video to weight the quality ratings of complex videos in terms of temporal activities over the low complex videos. Therefore, we fit a polynomial model to predict MOS values based on the average TC value of videos and NDNetGaming Score. The selection of polynomial model was based on our observation of relation between TI values and residual of NDNetGaming prediction and MOS values. Due to limited available data, we fit the model based on the GVSET and test it on the KUGVD and vice versa for which the results are reported in Section 6.2. In addition to the two fitted models, we provide a third model which is fitted on both gaming datasets for future works. Equation (6) presents the structure of temporal pooling model, and Table 5 presents the coefficient of the model trained based on different datasets.

$$TC = mean_{time}[std_{space}[M_n(i, j)]] \quad (5)$$

$$NDNG_{Temporal} = c_1 + c_2 \times NDNG + c_3 \times TC^3 + c_4 \times TC^2 + c_5 \times TC \quad (6)$$

## 6 Evaluation

In this section, we evaluate our model for different datasets, image and video content, and compare the results with the state-of-the-art video quality metrics. Thus, we first test our model on image quality datasets for which no gaming dataset is available. Then we evaluate our model for gaming and non-gaming video datasets and evaluate its performance.

We evaluated a total of twelve image/video quality assessment (VQA) metrics on the dataset as follows:

**Peak Signal to Noise Ratio (PSNR)** is the most widely used VQA metric and relies on the computation of the logarithmic difference between corresponding pixels in the original and impaired frame.

**Structural Similarity Index Metric (SSIM)** measures the structural similarity between two images and usually provides better video quality predictions compared to PSNR.

**Table 5** Coefficients of temporal pooling methods,  $eq_{GVSET}$ ,  $eq_{KUGVD}$  and  $eq_{GVSET-KUGVD}$  are trained based on GVSET, KUGVD and datasets combined respectively

	$c_1$	$c_2$	$c_3$	$c_4$	$c_5$
$eq_{GVSET}$	-1.99	1.097	0.00069	-0.031	0.43
$eq_{KUGVD}$	-0.532	1.116	0.00011	-0.0043	0.084
$eq_{GVSET-KUGVD}$	-1.71	1.107	0.00053	-0.024	0.353

**Video Multi-Method Assessment Fusion (VMAF)** developed by Netflix, fuses three different metrics together to obtain a single score as the estimation of the video quality [27].

**Spatio-temporal-reduced reference entropic differences time Optimized (ST-RREDOpt)** is a RR metric which measures the amount of spatial and temporal information differences in terms of wavelet coefficients of the frames and frame differences between the distorted and received videos [10]

**Spatial Efficient Entropic Differencing for Quality Assessment (SpEED-QA)** is a RR metric which relies on local spatial operations on image frames and frame differences to compute perceptually relevant image/video quality features in an efficient way [1].

**Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE)** uses locally normalized luminance coefficient tries to quantify the possible loss of “naturalness” [25].

**Natural Image Quality Evaluator (NIQE)** is based on a space domain NSS model, is a learning-based quality estimation metric [26].

**Perception based Image Quality Evaluator (PIQE)** is an NR metric that uses cues from human visual system to build a NR quality metric [37].

**Multi-task End-to-end Optimized deep Neural network (MEON)** is blind image quality metric by decomposing the quality assessment task into two subtasks with dependent loss functions [24]

**No Reference Gaming Video Quality Metric (NR-GVQM)** is machine learning-based video quality metric for gaming content which is trained based on low level image features with the aim of predicting VMAF without having access to a reference video [39].

**Lightweight NR pixel based model for gaming content (Nofu)** is a pixel-based video quality model for Gaming Content [15]. Nofu uses 12 different per frame based values and a venter crop approach for the fast computation of frame-level features.

**NR-GVSQI** is a NR metric designed for gaming content using Neural Networks [3]. NR-GVSQI uses 15 low level signal features including the score of NR metrics such as BRISQUE and NIQE for training the model.

## 6.1 Image quality assessment

In this section, we investigate the performance of our quality model on the non-gaming image datasets in order to check whether it performs well with other types of content. Thus, we looked for image quality datasets that are consisted of two types of degradation that we trained our model on, blockiness and blur. We selected LIVE Public-Domain Subjective Image Quality Database (the first release), and LIVE Multiply Distorted Image Quality Database [22, 34], which we refer to them as LPDSET and LMDSET for the rest of the paper. LMDSET consists of two parts, one part made based on the images that are first blurred and then compressed by a JPEG encoder (named Part-1 in this paper). In the second part, images are first blurred due to narrow depth of field or other defocus and then corrupted

by white Gaussian noise to simulate sensor noise (named Part-2 in this paper). We used the first part in our analysis as noise is not known to our model. It has to be noted that we did not perform VMAF on the image dataset as VMAF is video metric. The results in Table 6 show that the NDNetGaming metric performs very well on LPDSET compared to the state-of-the-art metrics. On the LMDSET dataset, our proposed model does not perform as good as NIQE and PIQE but better than PSNR, SSIM and Brisque. We also performed our model on the second part of LMDSET which has unknown degradation to the model. As expected, our model did not perform well with PCC of 0.45. However, it has to be noted that the network can be retrained for new types of distortion with the same approach that we introduced in this paper which is not in the scope of this paper.

### 6.2 Video quality assessment

In this section, we investigate the performance of our model based on the different video databases, both gaming and non-gaming datasets. For gaming dataset, the GVSET and the KUGVD dataset are used for evaluation. Table 7 presents the performance of our metric on gaming datasets in terms of PCC and SRCC based on the average pooling method. It has to be noted that the performance of nofu for GVSET is reported based on results in the paper [15] in which authors used leave-one-out cross-validation to gain the result. In addition, the result of nofu for KUGVD is not reported due to unavailability of the source code. In addition, the scatter plot of temporal pooled NDNetgaming scores and actual MOS values for both datasets are plotted in Fig. 7. The RMSE of the model, using average pooling, based on the gaming datasets are 0.347 and 0.464 for GVSET and KUGVD respectively.

The results revealed high performance of the model on the two available gaming datasets. For fair comparison between our proposed model and others, we used the average pooling method for all metrics in Table 7. We also tested the temporal pooling method we proposed in Section 5.4. The result shows PCC of 0.961 (RMSE = 0.27) for GVSET and 0.968 (RMSE = 0.30) for KUGVD. The scatter plots of predicated MOS and assessed MOS are shown in Fig. 8. It can be seen that the outliers in Fig. 7 does not exist anymore.

In order to confirm the performance of the model, we evaluated the model on a non-gaming video quality dataset. Thus, we used the public part of Live-NFLX-I dataset [2] and the Netflix Public dataset [28]. It has to be noted that, distorted videos of Live-NFLX-I were generated from 3 video contents of spatial resolution 1080p by imposing a set of 8 different playout patterns on them ranging from dynamically changing H.264 compression rates and re-buffering events to a mixture of compression and re-buffering. Therefore, all metrics are calculated after removing rebuffered frames for Live-NFLX-I. The public Netflix dataset is

**Table 6** Performance of our model in terms of Pearson and Spearman correlation on LPDSET and LMDSET

Metrics		LMDSET		LPDSET	
		PCC	SRCC	PCC	SRCC
FR Metrics	PSNR	−0.69	−0.64	0.80	0.93
	SSIM	−0.58	−0.61	0.92	0.94
NR Metrics	BRISQUE	0.57	0.43	−0.93	−0.92
	NIQE	0.87	−0.62	−0.92	−0.89
	PIQE	0.82	0.77	−0.90	−0.87
	NDNetGaming	−0.77	−0.68	0.95	0.92

**Table 7** VQA metrics performance on the two gaming video datasets

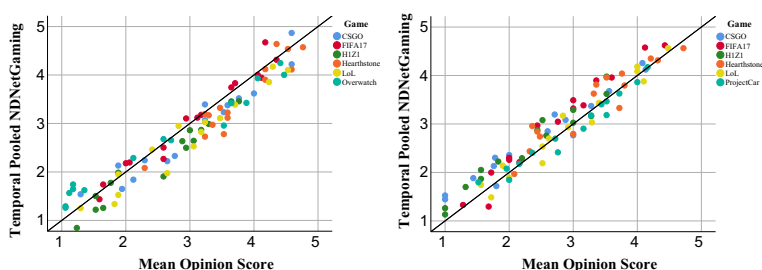
Metrics		GVSET		KUGVD	
		PCC	SRCC	PCC	SRCC
FR Metrics	PSNR	0.75	0.74	0.80	0.78
	SSIM	0.80	0.80	0.89	0.88
	VMAF	0.87	0.87	0.92	0.92
RR Metrics	ST-RREDOpt	−0.75	−0.77	−0.73	−0.72
	SpEEDQA	−0.75	−0.77	−0.70	−0.70
NR Metrics	BRISQUE	−0.44	−0.46	−0.62	−0.60
	BIQI	−0.42	−0.45	−0.60	−0.59
	NIQE	−0.72	−0.71	−0.85	−0.84
	MEON	−0.35	−0.30	−0.43	−0.39
	NR-GVQM	0.89	0.87	0.91	0.91
	NR-GVSQI	0.87	0.86	0.89	0.88
	nofu	0.91	0.91	–	–
	NDNetGaming	0.934	0.933	0.934	0.929

created from 9 video source contents with a spatial resolution of 1080p at 24, 25 and 30 fps. The source videos are compressed using different bitrate-resolution pairs with bitrate ranging from 375 kbps to 5800 kbps and resolution ranging from 288p to 1080p.

The result for Netflix Public Dataset shows higher performance of NDNetGaming compared to other NR metrics, while VMAF performs the highest among all metrics (Table 8). For LIVE-NFLX-I, the result revealed higher performance of proposed model compared to VMAF, but PIQE achieved the highest performance. It has to be noted that LIVE-NFLX-I consists of only low quality ratings that are dense in the lower side of the scale. This results in very low SRCC values for some metrics, e.g. VMAF, as the distribution is not spread enough to deal with errors.

## 7 Discussion and limitation

This paper is presented to demonstrate the effectiveness of the usage of CNNs for quality assessment of multimedia services. While such CNN based quality metrics come with high computation cost which might not be suitable for real-time services monitoring, the high



**Fig. 8** Scatter plot of temporal pooled video prediction of NDNetGaming and MOS for KUGVD (*left side*) and GVSET (*right side*)

**Table 8** VQA metrics performance on the two public non-gaming video datasets

Metrics		Netflix public dataset		LIVE-NFLX-I	
		PCC	SRCC	PCC	SRCC
FR Metrics	PSNR	0.64	0.66	0.49	0.27
	SSIM	0.69	0.76	0.24	-0.10
	VMAF	0.93	0.91	0.78	0.24
NR Metrics	BRISQUE	-0.77	-0.76	-0.65	-0.68
	NIQE	-0.83	-0.81	-0.67	-0.28
	PIQE	-0.78	-0.80	-0.85	-0.83
	NDNetGaming	0.89	0.85	0.82	0.71

performance of these methods motivates us to consider them as a future of QoE assessment for some special use cases such as post assessment of cloud gaming service quality and measurement of quality for uploaded video content for transcoding purposes, e.g. on Twitch.tv.

Our testing showed that the proposed model can predict the frame-level quality of 1800 frames of a 1080p resolution video sequence in approximately five minutes using a PC with a NVIDIA's GeForce GTX 1080 graphic card, and 32 Mb Ram. However, it can be reduced significantly by using center cropping method which is proposed in [14] or taking sub-sample frames per sequence as it is used for fast VMAF calculation [27].

Our results show that the proposed model can predict the quality with high accuracy for gaming content. In addition to KUGVD and GVSET datasets that are discussed in the paper, the model was performed in a large-scale gaming video dataset, named CGVDS, which outperformed the existing video quality metrics [41]. CGVDS has a wider range of video games as well as encoding parameters. This result confirms that the high performance of the model is not due to similar encoding setting of KUGVD and GVSET.

While, the NDNetGaming is trained for gaming content, it performs well with non-gaming content. The results of the paper give insight on the weakness and strength of these CNN models. While CNN based quality models can lead us to gain high accuracy in prediction for a specific type of content or distortion, they fail to accurately predict the quality of unknown degradation to the model or new types of content which deviate much from the training set. In addition, we tried to train the model based on multiple distortions from non-gaming image quality dataset, which revealed that increasing the diversity of distortion in training set would decrease the prediction in overall. Moreover, the accuracy of prediction for different types distortions differs depending on the distortion type. For example, in our dataset, the model predicts blur artifacts better than blockiness.

While it would be interesting to compare our model with deep learning-based models, we did not make such a comparison for a few reasons. First, the source codes of those models are not always available. Second, most of deep learning models are trained on datasets with different type of artifacts which result in low correlation with our gaming dataset and it is not fair to make such a comparison. For example, Neural Image Assessment (NIMA) [36] was tested on our gaming dataset which resulted in a low PLCC of 0.54 with subjective ratings, which apparently is due to the training process. Retraining the model would not be an option due to the limited number of data for gaming content. The only way to fairly compare our work with other deep learning-based models would be retraining other models in similar framework we used in this paper. We believe by retraining the model in a similar way we proposed, would rather result in a comparison between

the two CNN architectures and not the modelling process which is the main focus of this work.

For cloud gaming and live gaming streaming, typically we do not have distortion such as block loss or noise. Therefore, these artifacts are excluded in our training dataset. Consequently, our model would not perform well with these types of distortions. However, the model can be retrained in the similar structure to cover a wider range of distortions.

Also, it must be noted that the model can be improved by changing the CNN architecture, a better objective metric instead of VMAF, a higher content diversity, and finally a larger dataset.

## 8 Conclusion and future works

In this paper, we present a framework to build a deep learn-based quality metric to assess gaming video quality. While there are several deep learning based image and video quality models are available in research community, this work focuses on a special user generate content which dealt with limitations such as limited available dataset and diversity of content. In addition to the proposed framework and modeling, we made a public image quality dataset of gaming scenes and proposed a temporal pooling method to take into account the temporal masking effect.

Due to the inherent nature of the available datasets, the proposed model is limited to only compression and scaling artefacts, however, it can be extended to other type of distortion if the training dataset is available. Also, currently both datasets, KUGVD and GVSET, are limited in scope considering the number of different games and the resolution-bitrate pairs considered. Since the datasets consist of videos compressed with the H.264 encoder, the performance of the proposed model on videos encoded with other encoders such as H.265, VP9, or AV1 remains for future works.

It has to be noted that the idea of this paper is not to propose a new CNN architecture for image and video quality prediction, but a framework that can help the research community to efficiently train a CNN for the content that limited number of datasets are available.

We plan to extend our work by using more divers content from gaming and non-gaming content with more types of video distortion. In addition, 3-dimensional convolutions can used to train the CNN directly from video sequences instead of frames of the videos. While, this method comes with higher computation cost, we believe it can improve the accuracy of the model which remains for future investigation.

**Acknowledgments** This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 871793.

**Funding Information** Open Access funding provided by Projekt DEAL.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Bampis CG, Gupta P, Soundararajan R, Bovik AC (2017) SpEED-QA: spatial efficient entropic differencing for image and video quality. *IEEE Signal Proc Lett* 24:1333–1337
- Bampis CG, Li Z, Moorthy AK, Katsavounidis I, Aaron A, Bovik AC (2017) Study of temporal effects on subjective video quality of experience. *IEEE Trans Image Process* 26(11):5217–5231
- Barman N, Jammeh E, Ghorashi SA, Martini MG (2019) No-reference video quality estimation based on machine learning for passive gaming video streaming applications. *IEEE Access* 7:74511–74527
- Barman N, Martini MG, Zadtootaghaj S, Möller S, Lee S (2018) A comparative quality assessment study for gaming and non-gaming videos. In: 2018 tenth international conference on quality of multimedia experience (QoMEX). IEEE, pp 1–6
- Barman N, Schmidt S, Zadtootaghaj S, Martini MG, Möller S (2018) An evaluation of video quality assessment metrics for passive gaming video streaming. In: Proceedings of the 23rd packet video workshop. ACM, pp 7–12
- Barman N, Zadtootaghaj S, Schmidt S, Martini MG, Möller S (2018) GamingvideoSET: a dataset for gaming video streaming applications. In: 2018 16th annual workshop on network and systems support for games (NetGames). IEEE, pp 1–6
- Barman N, Zadtootaghaj S, Schmidt S, Martini MG, Möller S (2018) An objective and subjective quality assessment study of passive gaming video streaming. *Int J Netw Manag* 30(3):e2054
- Bosse S, Becker S, Müller K-R, Samek W, Wiegand T (2019) Estimation of distortion sensitivity for visual quality prediction using a convolutional neural network. *Digital Signal Processing* 91:54–65
- Bosse S, Maniry D, Müller K-R, Wiegand T, Samek W (2018) Deep neural networks for no-reference and full-reference image quality assessment. *IEEE Trans Image Process* 27(1):206–219
- Bovik AC, Soundararajan R, Bampis C (2017) On the robust performance of the ST-RRED video quality predictor. <http://live.ece.utexas.edu/research/quality/ST-RRED/>
- Choi LK, Bovik AC (2018) Video quality assessment accounting for temporal visual masking of local flicker. *Signal Process Image Commun* 67:182–198
- Chollet F (2015) Keras. <https://keras.io>
- Chollet F (2017) Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1251–1258
- Göring S, Krämmer C, Raake A (2019) Cencro-speedup of video quality calculation using center cropping. In: 2019 IEEE international symposium on multimedia (ISM). IEEE, pp 1–17
- Göring S, Rao RRR, Raake A (2019) Nofu- a lightweight no-reference pixel based video quality model for gaming QoE. In: Accepted at eleventh international workshop on quality of multimedia experience (QoMEX), pp 1–6
- He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
- Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708
- ITU-t recommendation p.910: Subjective video quality assessment methods for multimedia applications. International telecommunication union (1999)
- ITU-r recommendation bt-500-19: Methodology for the subjective assessment of the quality of television pictures. International telecommunication union (2002)
- ITU-t recommendation p.910: Subjective video quality assessment methods for multimedia applications. International telecommunication union (2008)
- ITU-T rec. p.913: Methods for the subjective assessment of video quality, audio quality and audiovisual quality of internet video and distribution quality television in any environment, March 2016
- Jayaraman D, Mittal A, Moorthy AK, Bovik AC (2012) Objective quality assessment of multiply distorted images. In: 2012 conference record of the forty sixth asilomar conference on signals, systems and computers (ASILOMAR). IEEE, pp 1693–1697
- Judd T, Ehinger K, Durand F, Torralba A (2009) Learning to predict where humans look. In: 2009 IEEE 12th international conference on computer vision. IEEE, pp 2106–2113
- Ma K, Liu W, Zhang K, Duanmu Z, Wang Z, Zuo W (2017) End-to-end blind image quality assessment using deep neural networks. *IEEE Trans Image Process* 27(3):1202–1213
- Mittal A, Moorthy AK, Bovik AC (2012) No-reference image quality assessment in the spatial domain. *IEEE Trans Image Process* 21(12):4695–4708
- Mittal A, Soundararajan R, Bovik AC (March 2013) Making a “completely blind” image quality analyzer. *IEEE Signal Process Lett* 20(3):209–212

27. Netflix. VMAF - video multi-method assessment fusion. <https://github.com/netflix/vmaf>. [online: Accessed 2 Oct 2018]
28. Netflix Public Dataset. <https://github.com/netflix/vmaf/blob/master/resource/doc/datasets.md>. [online: Accessed 06 September 2019]
29. Rao RRR, Göring S, Vogel P, Pachatz N, Villarreal JJV, Robitzka W, List P, Feiten B, Raake A (2019) Adaptive video streaming with current codecs and formats: ex-tensions to parametric video quality model itu-t p. 1203. Electronic Imaging, Image Quality and System Performance XVI
30. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, et al. (2015) Imagenet large scale visual recognition challenge. *Int J Comput Vis* 115(3):211–252
31. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C (2018) Mobilenetv2: inverted residuals and linear bottlenecks. In: The IEEE conference on computer vision and pattern recognition (CVPR)
32. Schiffner F, Möller S (2018) Direct scaling & quality prediction for perceptual video quality dimensions. In: 2018 tenth international conference on quality of multimedia experience (QoMEX). IEEE, pp 1–3
33. Seufert M, Slanina M, Egger S, Kottkamp M (2013) To pool or not to pool: a comparison of temporal pooling methods for http adaptive video streaming. In: 2013 fifth international workshop on quality of multimedia experience (QoMEX). IEEE, pp 52–57
34. Sheikh HR, Sabir MF, Bovik AC (2006) A statistical evaluation of recent full reference image quality assessment algorithms. *IEEE Trans Image Process* 15(11):3440–3451
35. Sperling G (1965) Temporal and spatial visual masking. I. Masking by impulse flashes. *JOSA* 55(5):541–559
36. Talebi H, Milanfar P (2018) Nima: neural image assessment. *IEEE Trans Image Process* 27(8):3998–4011
37. Venkatanath N, Praneeth D, Bh MC, Channappayya SS, Medasani SS (2015) Blind image quality evaluation using perception based features. In: 2015 twenty first national conference on communications (NCC). IEEE, pp 1–6
38. Weiss K, Khoshgoftaar TM, Wang D (2016) A survey of transfer learning. *Journal of Big Data* 3(1):9
39. Zadtootaghaj S, Barman N, Schmidt S, Martini MG, Möller S (2018) NR-GVQM: a no reference gaming video quality metric. In: 2018 IEEE international symposium on multimedia (ISM). IEEE, pp 131–134
40. Zadtootaghaj S, Schmidt S, Barman N, Möller S, Martini MG (2018) A classification of video games based on game characteristics linked to video coding complexity. In: 2018 16th annual workshop on network and systems support for games (NetGames). IEEE, pp 1–6
41. Zadtootaghaj S, Schmidt S, Shafiee Sabet S, Moeller S, Griwodz C (2020) Quality estimation models for gaming video streaming services using perceptual video quality dimensions. In: Proceedings of the 11th international conference on multimedia systems. ACM

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

## Affiliations

Markus Utke<sup>1</sup> · Saman Zadtootaghaj<sup>1</sup>  · Steven Schmidt<sup>1</sup> · Sebastian Bosse<sup>2</sup> · Sebastian Möller<sup>1</sup>

Markus Utke  
markus.utke@campus.tu-berlin.de

Steven Schmidt  
steven.schmidt@tu-berlin.de

Sebastian Bosse  
sebastian.bosse@hhi.fraunhofer.de

Sebastian Möller  
sebastian.moeller@tu-berlin.de

<sup>1</sup> Technische Universität Berlin, Berlin, Germany

<sup>2</sup> Fraunhofer Heinrich Hertz Institute, Berlin, Germany