This is the accepted version of this paper. The version of record is available at https://doi.org/10.1109/MMSP48831.2020.9287080

DEMI: Deep Video Quality Estimation Model using Perceptual Video Quality Dimensions

Saman Zadtootaghaj*, Nabajeet Barman†, Rakesh Rao Ramachandra Rao‡, Steve Göring‡,
Maria G. Martini†, Alexander Raake‡, Sebastian Möller*§
*Quality and Usability Lab, TU Berlin, Germany, saman.zadtootaghaj@qu.tu-berlin.de
†Kingston University, London, UK, {n.barman, m.martini}@kingston.ac.uk
‡Technische Universität Ilmenau, Germany, {rakesh-rao.ramachandra-rao, steve.goering, alexander.raake}@tu-ilmenau.de
§DFKI Projektbüro Berlin, Germany, sebastian.moeller@ {dfki.de, tu-berlin.de}

***Abstract*—Existing works in the field of quality assessment focus separately on gaming and non-gaming content. Along with the traditional modeling approaches, deep learning based approaches have been used to develop quality models, due to their high prediction accuracy. In this paper, we present a deep learning based quality estimation model considering both gaming and non-gaming videos. The model is developed in three phases. First, a convolutional neural network (CNN) is trained based on an objective metric which allows the CNN to learn video artifacts such as blurriness and blockiness. Next, the model is fine-tuned based on a small image quality dataset using blockiness and blurriness ratings. Finally, a Random Forest is used to pool frame-level predictions and temporal information of videos in order to predict the overall video quality. The light-weight, low complexity nature of the model makes it suitable for real-time applications considering both gaming and non-gaming content while achieving similar performance to existing state-of-the-art model NDNetGaming. The model implementation for testing is available on GitHub[1].**

***Index Terms*—Quality of Experience, Video Quality Estimation, Quality Models, Deep Learning, Gaming Video Streaming**

I. Introduction

The video streaming industry is booming with growth of users of media streaming services such as Netflix and YouTube, video conferencing applications (e.g., Zoom and MS Teams), and gaming video streaming (e.g., Twitch and Facebook Gaming) [1]. Gaming video streaming consists of a rapidly growing market with emerging online services such as gaming video streaming, online gaming and cloud gaming (CG) services. In cloud gaming the heavy processes such as rendering is performed on the cloud and hence does not require high-end hardware devices at the user end. Recently introduced services such as Stadia, and existing services such as Nvidia Geforce Now and Magenta Gaming by Deutsche Telekom are some of the examples of such services. Apart from processing power, cloud gaming benefits users by the platform in-dependency 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 gameplay have become popular with hundreds of millions of viewers per year with Twitch.tv currently being the most popular services for passive video game streaming.

[1]https://github.com/stootaghaj/DEMI

In times of exceptional circumstances such as the current Covid-19, it is imperative that such services meet the minimum required quality of experience (QoE) to the end users and video quality forms one of the most important components of QoE. Video quality assessment is a highly subjective task, as several factors (resolution, number of stalling events, etc. [2]) play a role in the final judgement of a user about a given service. Many services rely on the use of objective quality models and metrics which try to predict the quality as perceived by humans. Therefore, over the past many years, there have been significant efforts towards the development and usage of quality models for quality prediction of multimedia services. For example, Netflix developed a video quality metric, Video Multimethod Assessment Fusion (VMAF), to measure the video quality considering encoding and rescaling artifacts, as they are the only compression related artifacts in a HTTP Adaptive Streaming based application [2].

While several studies have been done on proposing quality assessment models and metrics for traditional video streaming services (e.g., Netflix and YouTube), new types of streaming content such as gaming video streaming has only recently started receiving attention of the industry and academia. For a more cleared discussion on the difference between these, we refer the reader to the discussion in Chapter 4 in [3]. Towards this end, in this paper, we present a deep learning model called DEMI to predict the video quality of compressed videos for both gaming and non-gaming content. The remaining part of the paper is organized as follows. In Section II we present a discussion on the related work. The description of datasets that are used for evaluation of the model is presented in Section III. Following this, Section IV presents the proposed model architecture while in Section V we present a discussion on the model development and presents the performance results of the proposed model compared to other existing models. We then conclude the paper in Section VI.

II. Related Work

Over past several years, due to the nature of the service/application in that there is no unimpaired, reference signal available (e.g., user generated content), there is a growing demand for no-reference (NR) metrics. Several recent works

have tried to address such NR models for both gaming as well as non-gaming applications, which we briefly discuss next.

In light of the peculiarities of gaming content, several gaming-specific video quality models have been developed. The focus has mainly been on developing NR models due to the lack of availability of pristine quality reference videos in a typical gaming scenario. Zadtootaghaj *et al.* [4] proposed a NR machine learning-based video quality metric named NR-GVQM for gaming content which is focused on frame-level feature extraction. The authors proposed a model which collects low-level image features from the frame of the video and trained the model using VMAF scores for the frame. The model uses pre-trained model using BRISQUE features, which was trained on non-gaming content and its quality prediction on gaming content was proven to be not satisfactory. Göring *et al.* [5] proposed a NR metric called nofu, which is a pixel-based video quality model designed for gaming content. nofu uses 12 different per frame based values and a center crop approach for the fast computation of frame-level features. It further uses frame-level features pooling at video-level and feeds the features to machine learning based model for the model development. nofu showed promising results on GamingVideoSET [6] using a 10-fold cross-validation approach. Barman *et al.* [7] proposed two NR metrics, NR-GVSQI and NR-GVSQE, to predict the quality of gaming content considering a passive gaming video streaming scenario. NR-GVSQI is designed using Neural Networks and it uses the MOS score as a target value for training. This model uses 15 NR features and three features from the score of three NR metrics for training of the machine learning based model. It uses GamingVideoSET [6] for training and KUGVD (also known as Kingston University Gaming Video data) as the validation dataset. NR-GVQSE was designed as the NR equivalent of VMAF (i.e. using VMAF as groundtruth) and performed well with a Pearson Correlation (PCC) of 0.97 with VMAF. Utke *et al.* [8] proposed a deep learning based gaming video quality metrics which outperforms the existing signal based video quality metrics.

Within the recent years there has been a growing interest in the application of deep neural networks (DNNs) for image and video quality assessment tasks. Since the amount of data, especially datasets with subjective scores is still very less for training a deep learning model, it is difficult to train a "deep" neural network. One approach to train such a model is by the use of transfer learning where the network is learned by transferring information from a related domain. Still, such approaches are limited to images and their application for video quality evaluation is still limited [9], [10], [11].

III. Evaluation Datasets and Methodology

In this work, we used five public video quality datasets, three from gaming namely, GamingVideoSet [6], KUGVD [7] and CGVDS [12] and non-gaming video datasets, namely, Netflix Public Dataset [13] and LIVE-NFLX-II Subjective Video QoE Database (NFLX-SVQD) [14]. The selection of the video quality datasets is done taking into account the similarity of encoding settings and range of parameters used.

GamingVideoSET (henceforth GVSET) presented in [6] consists of 24 reference video sequences from 12 different games with each video of 30 s duration, of 1920×1080 resolution and 30 fps. The reference videos are encoded in multiple resolution-bitrate pairs using H.264 video compression standard resulting in a total of 576 distorted video sequences. The dataset includes subjective ratings for 90 video sequences as well as per-frame scores for several FR and RR metrics for the whole dataset.

KUGVD is another publicly available dataset built in line with the encoding settings used in GamingVideoSET but limited to six reference video sequences presented in [7]. It also consists of 144 distorted video sequences with per-frame scores for the FR and RR metrics, as well as subjective MOS scores for 90 distorted video sequences.

Netflix Public Dataset (NFLX-PD) is a non-gaming video dataset provided by Netflix consisting of nine source video sequences of 1920×1080 resolution with framerates of 24, 25 and 30 fps. The videos are encoded in multiple resolution-bitrate pairs with bitrates ranging from 375 kbps to 5800 kbps and resolution ranging from 288p to 1080p.

LIVE-NFLX-II Subjective Video QoE Database (NFLX-SVQD) [14] consists of 15 source videos and a total of 420 distorted sequences obtained by encoding the source videos at different bitrates at native resolution. The dataset includes both objective and subjective quality ratings, both continuous as well as retrospective prediction scores.

In addition to the above five public gaming and non-gaming video datasets, we used a gaming image and one cloud gaming dataset described next.

GISET: is a gaming image dataset consisting of 164 images extracted from the GamingVideoSET dataset in which from each source image, three encoded images are selected, one with blockiness artifact, one with blurriness and finally one with mixture of these two degradations. GISET is the only image quality dataset annotated with blockiness and blurriness.

CGVDS: Cloud Gaming Video Dataset (CGVDS) [12] consists of a larger number of recording gaming content captured at 60 fps. Similar to the previously discussed gaming datasets, three different resolutions, namely, 480p, 720p and 1080p are considered at three different framerates of 20, 30 and 60 fps. The dataset includes results from five different subjective studies, each with three video games.

A. Perceptual Video Quality Dimensions

One of the reasons for the increasing popularity of adaptive streaming is the fact that in adaptive bitrate streaming using TCP there exist no visual quality impairment due to packet losses and bit-errors. The major impairments that arise during the lossy encoding process are compression artifacts and scaling artifacts which in turn affect the end user's QoE. Therefore, we decided to train our model based on the two of the three video quality dimensions (Fragmentation (Blockiness)

and Unclearness (Blurriness)) that are introduced in the ITU-T Rec. P.918 for the design of our video quality metric, which can also serve as a diagnostic tool. Table I summarizes the three identified dimensions which later are used to build the quality model using a Direct Scaling method. The video discontinuity dimension was not used in the training process.

TABLE I: Perceptual video quality dimensions introduced in ITU-T Rec P.918 [15]

VQD	Name	Description	Example Impairment
I	Fragmentation (FRA)	Fallen apart, torn and blockiness	Low Coding Bitrate
II	Unclearness (UCL)	Unclear and blurry image	Upscaling effect using bicubic function
III	Discontinuity (DIC)	Interruptions in the flow of the video	Low frame rate

IV. DEMI Model Architecture

In this section, we describe the architecture of the proposed model, DEMI, and the special model design. DEMI is a CNN based metric which takes into account different types of artifacts such as blockiness, blurriness and jerkiness, to predict the overall gaming video quality. The structure of the model is shown in Figure 1. DEMI has three components. The first component is a CNN which is used to predict the frame level blurriness and blockiness. Second component is a temporal complexity index which is based on Block Motion estimation (BM) and Temporal Index (TI). Finally the predicted blockiness, blurriness, TI and BM for multiple patches of a video is pooled using a random forest model to predict the video quality which is the third component of the proposed model.

A. Phase 1 – VMAF training

In order to train a CNN for the quality estimation task, a major limitation is the availability of a large scale image quality dataset with images and their subjective ratings. Mixing multiple datasets could be one option but it suffers strongly from many shortcomings such as subjective bias, difference in viewing conditions, display used, etc. and hence, requires an anchor dataset to deal with this bias which is missing in such cases. For training, we use the annotated frames using an objective, full-reference video quality metric called VMAF, as was done in [8]. The selection of VMAF is due to high performance of the metric for different types of content (including gaming content [16] when compression artifacts are present.

As the underlying CNN architecture, we chose the light-weight, DenseNET-121 architecture [17], which has been shown to perform well for image quality estimation tasks [8]. Selection of the DenseNET-121 is also considering the fact that the model is of very low complexity, with almost 8 million parameters (e.g. compared to ResNet50 with 25 million parameters), while reaching high accuracy for quality prediction problems [8]. In order to let DenseNET-121 learn a regression task (instead of the originally trained classification task), the fully connected layer at the end of the CNN was removed. Instead, we added a dense layer consisting of only one output neuron with linear activation. Training the model using the VMAF annotated frames allows the network to learn different types of image compression degradation such as blurriness and blockiness.

For DenseNET-121, we used the implementation available in [18]. For training the model, we crop nine non-overlapping patches, each of size 299×299 instead of default DenseNet patch size of 224×224, as recommended in [8], from each frame for training the model based on the VMAF. In its entirety, we used over 200k frames and their respective VMAF scores as the target. The frames are extracted from multiple videos from several datasets (see Section III for more information). Since nine patches are extracted per frame, the total number of inputs during the training phase is over a million.

Since we have VMAF scores only at frame level, we used Partial PSNR to determine the quality and the weight of each patch that contribute to the overall VMAF score. Thus, for the patch i of frame j, the weight of patch is calculated as follows:

$$W_{(i,j)} = PPSNR_{(i,j)}/PSNR_j \tag{1}$$

The quality of each patch then was determined based on the VMAF of each frame which is calculated as:

$$VMAF_{(i,j)} = VMAF_j * W_{(i,j)} \tag{2}$$

The selection of PSNR is due to the simplicity and nature of the metric as it only measures the signal to noise ratio and avoids any content bias or scaling adjustment as also used earlier by authors in [19]. Due to the high similarity between neighbouring frames, in the training process, we only used every 20th frame. The number is selected based on our experience from previous work in [8], which showed that a too long interval might negatively affect the result due to a smaller training set .

B. Phase 2- Fine-tuning

Once the model is trained based on VMAF, the model is then fine-tuned two times based on a small image quality dataset using Fragmentation (blockiness) and Unclearness (blurriness) subjective ratings. We retrain the 33 layers of Densenet-121 (one DenseNet block including 2191k parameters) using transfer learning. The 25% of CNN was retrained once based on the blur ratings and once based on the blockiness ratings. Since only 25 percent of the CNN was retrained two times, the overhead of double training (for blur and blockiness) does only result in computational overhead for testing the model for one additional DenseNet block. It needs to be noted that this additional step would slightly increase the prediction computation due to forward propagation of the prediction process.

C. Phase 3: Video Level

Once the model is fine-tuned based on the blockiness and bluriness, we collect the frame prediction level of the model to be used in the training process at the video level. In

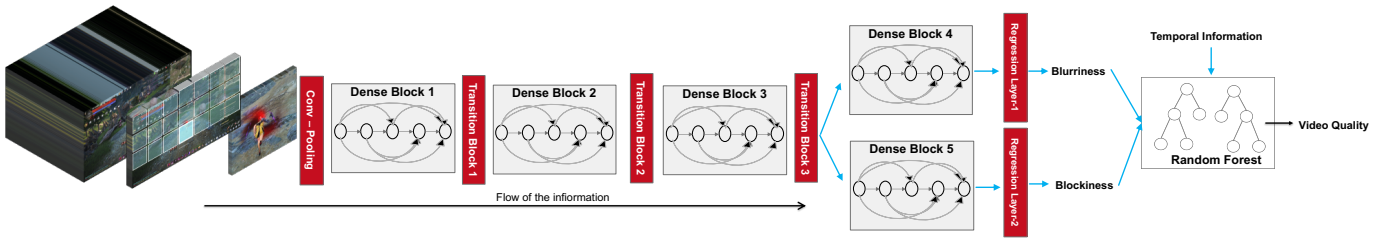


Fig. 1: Architecture of the proposed model (adapted based on [17]). Each transition block consists of 1x1 Conv and 2x2 Pool with stride 2. The regression layer has an average pool and a dense layer consisting of only one output with linear activation.

addition to the frame level prediction, we extracted temporal features (temporal index and block motion estimation) for better prediction on the video level. We then use Random Forest (RF) as the training algorithm to fuse the features for prediction of video quality. Since we have subjective scores from multiple datasets, in order to compensate for subjective bias, we used a linear mapping as recommended in [ITU-T. P.1401] per dataset to the objective quality scores before computing the performance of the evaluation metrics. This was done at video level only, as in the subjective scores are available only for video sequence.

We define temporal information (TI) at a frame level similar to ITU-T Rec. P.910 [20] as:

$$TI = std[M_p^n] \tag{3}$$

where M_p^n is the pixel intensity difference between F_p^n, current frame n, calculated as

$$M_p^n = F_p^n - F_p^{n-1} \tag{4}$$

where F_p^{n-1}, previous frame $n-1$. Block Motion (BM) estimation with a block size of 8x8 is calculated based on Sci-kit video library [21]. The block motion is then averaged over a frame (between two frames) and one value per frame (second frame in each prediction) is stored for training. With consideration of the low computation complexity during the test (considering real-time prediction requirement in real world applications), the frame-level information was extracted for every 20th frames. This number is based on previous research [8] and our investigation.

V. Model Training and Performance Evaluation

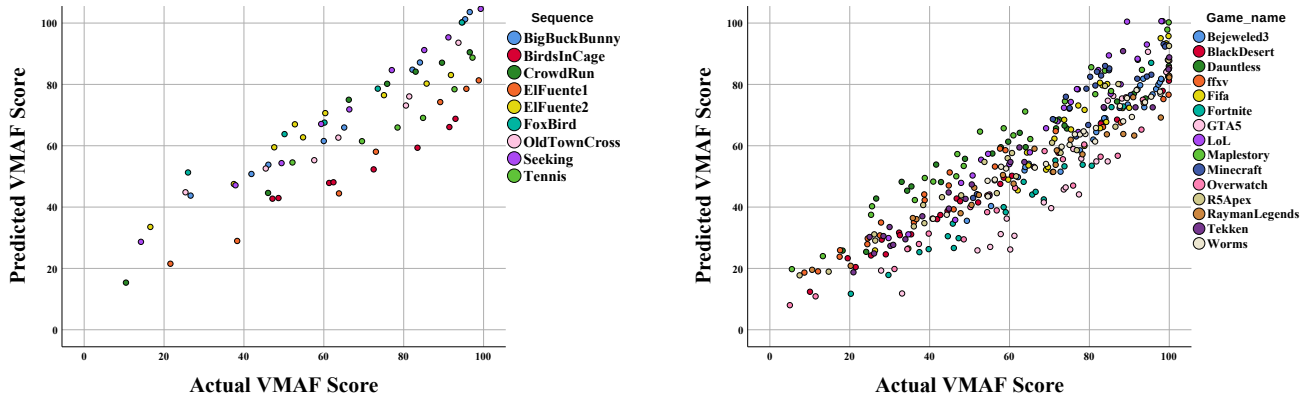
The model development was carried out in three phases of model training which we discuss next. In this section, we report the performance in terms of Pearson Linear Correlation Coefficient (PCC), Spearman's Rank Correlation Coefficient (SRCC) and Root Mean Square Error (RMSE) after each phase of training. The results are reported based on their performance on the training dataset. For the training, the scale of VMAF was from 1 to 100 and for the Phase-2 and 3 we used 5-point ACR scale and RMSE is reported accordingly.

A. Model Training

1) *Phase-1 (VMAF Training)*: In first phase, we train the model using VMAF scores from three datasets, GVSET, KUGVD, NFLX-PD. The DeneseNet-121 was trained based on the frames extracted from these three datasets, using the VMAF scores as the target labels. The result on the training set showed high performance with RMSE of 5.15 and PCC score of 0.943 at frame level and RMSE of 3.25 and PCC of 0.954 at video level (using average pooling) across all datasets. The result on the two validation datasets is shown in Figure 2.
2) *Phase-2 (Fine-tuning)*: Once the model is trained based on VMAF scores, it is then fine-tuned based on MOS scores from GISET, as it includes scores for both blockiness and blurriness. The model is fine-tuned in two steps, once using the scores for blockiness and once based on scores for blurriness. The same weighting method explained in Phase-1 was applied to the rating of each patch. Since the number of images is quite less, we used a leave one out cross-validation method where we left out video sequences from a game (reference video together with all encoded videos of that video sequence). The process is repeated twelve times for each game in the GISET. The result shows high performance of model for both blockiness, with PCC of 0.94 and RMSE of 0.39, and blurriness with PCC of 0.92 and RMSE of 0.45. Due to small size of dataset, we extracted all possible non-overlapping patches for fine-tuning the model.
3) *Phase-3 (Video-Level)*: In Phase-3, we train the model at video level using four datasets, GVSET, KUGVD, NFLX-PD and a subset of CGVDS consisting of videos of 60 fps (since the other datasets were limited to videos of upto 30 fps). We trained a random forest model based on temporal features and the predicted blockiness and blurriness scores. The features are extracted only from nine patches of a frame and only from every 20th frame. The statistical information of patch features over a video is used in training of the random forest. The result for the training data showed a very high PCC score of 0.941 and RMSE of 0.31.

B. Model Performance Evaluation

The final model at the end of the training process is called DEMI which is then evaluated using two datasets not used in the training process. One each from gaming (CGVDS) and one non-gaming (Live-NFLX-1) are used for testing the model performance. Using NFLX-PD and CGVDS for



(a) Predicted VMAF vs. Actual VMAF scores for NFLX-PD dataset. (b) Predicted VMAF vs. Actual VMAF scores for CGVDS dataset.

Fig. 2: Scatter plots of predicted VMAF vs Actual VMAF scores for the two test datasets.

TABLE II: Comparison of Model Performance

Metrics		NFLX-PD		CGVDS	
		PCC	SRCC	PCC	SRCC
FR Metrics	PSNR	0.64	0.66	0.66	0.67
	SSIM	0.69	0.76	0.64	0.76
	VMAF	**0.93**	**0.91**	0.87	0.87
NR Metrics	BRISQUE	-0.77	-0.76	-0.48	-0.46
	NIQE	-0.83	-0.81	-0.53	-0.53
	PIQE	-0.78	-0.80	-0.41	-0.41
	NDNetGaming	0.89	0.85	0.92	**0.93**
	DEMI	0.89	0.89	**0.93**	0.92

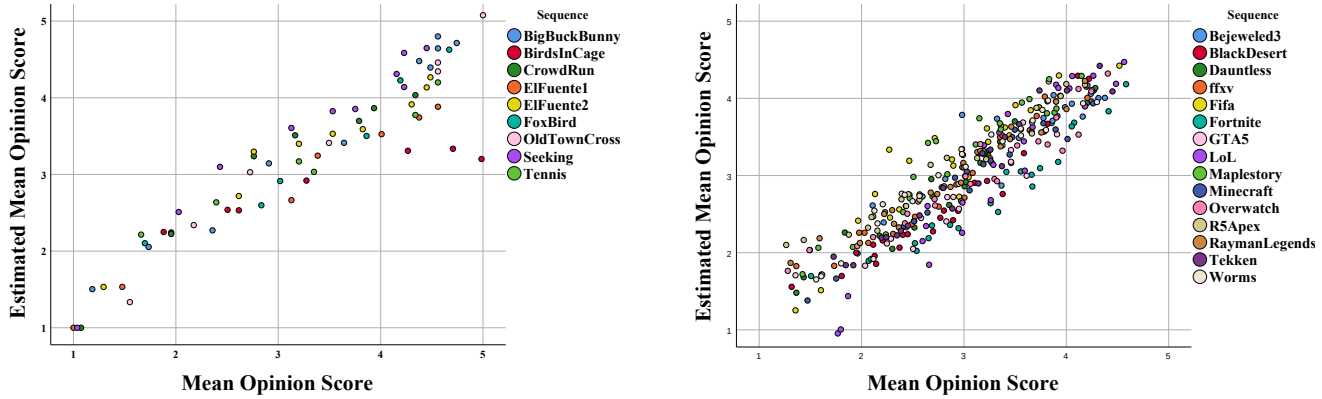
validation, the results perform well which is shown in Table II which is compared with some of the well known FR and NR quality metrics. It can be observed that the result for gaming video dataset, CGVDS, is slightly higher than Live dataset which might be due to the fact that there is a higher number of gaming frames in the training set. Based on the Table II, we can see that NDNetGaming performs slightly higher than DEMI for CGVDS dataset. This is due to the fact that NDNetGaming is trained only based on the gaming video dataset and it is more complex compared to DEMI. However, DEMI outperform NDNetGaming for non-gaming content while still behind VMAF. It has to be noted that VMAF is trained based on the similar dataset to NFLX-PD and the result could be biased for VMAF on this dataset.

In addition, we compared the performance of the metric with MOS on the scatter plot in Figure 3. The scatter plot showing that the model performs very well with gaming video dataset. We can observe a few underestimation for very low complex sequences. For example, we can see that DEMI underestimates a few sequences of the game, League of Legend (LoL), which is recorded from special level named teamfight tactics. We believe this result is due to the training process where more complex video games exist in the training dataset compared to low complex sequences. Similar result can be seen for NFLX-PD.

For testing, we used a PC with 16 GB RAM and NVIDIA graphic card of GTX 1080, on which our model took less than 200 seconds for a 1080p video of 30 seconds duration. The reduced computation time is due to the fact that we sample frames and patches, as explained in Section V-A, to a minimum for reduction in computation complexity. We did not compare our model with existing deep learning video quality models due to the following practical and theoretical reasons. First, the source code 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 selected validation datasets and it is not fair to make such a comparison. In addition, within the same CNN architecture, deeper CNNs typically perform better (e.g. DenseNet201 performs better than DenseNet121 on ImageNet) and such a comparison is valid if we have similar number of trainable parameters.

C. Discussion

In this paper, we presented a deep learning based video quality model which is trained based on gaming and non-gaming content. While the proposed model is more complex during the training phase than the state-of-the-art (NDNetGaming) model, its complexity during test (runtime) phase is greatly reduced due to a much lower sampling rate of frames from the video and hence, reduced number of computations. An exact comparison of complexity is out of the scope of this paper and will be presented in future work. In the Phase-1, we decided to not combine multiple image quality datasets due to subjective bias that could occur and influence the result, and hence instead we used VMAF for training. Using GISET in Phase-2, we augmented the training process for Blockiness and Blurriness artefacts. In Phase-3, we combined the three video datasets in the training process. The reason behind such a approach in Phase-1 and Phase-3 is that for Phase-1 we wanted to combine a huge image quality dataset to allow the deep CNN to learn image compression artifacts. While in the Phase-3, we only combine the three datasets that are similar in terms of methodology of subjective test. One major advantage that the model provides over the existing models is that a service provider can use the quality dimensions output as a



(a) Estimated MOS vs. Actual MOS scores for NFLX-PD dataset.

(b) Estimated MOS vs. MOS scores for CGVDS dataset.

Fig. 3: Scatter plots of predicted MOS vs. MOS scores for the two test datasets.

diagnostic tool to improve the quality of experience of the user by providing improved quality video to the end user.

VI. Conclusion and Future Work

We presented in this work a deep learning based model DEMI for quality prediction of both gaming and non-gaming content. DEMI was trained on four different publicly available datasets and its performance was independently evaluated on two separate datasets. A performance comparison with existing models showed that it achieves similar performance as the state-of-the-art model NDNetGaming but at a much reduced complexity making it suitable for real-world quality estimation tasks. The quality dimensions output can also be used by service providers to adapt the video quality to enhance the quality of experience of the end user, hence making such a model suitable for QoE based measurement and control. Our future work will include assessment of other transfer learning methods, inclusion of more quality dimensions and more relevant datasets.

Acknowledgment

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

References

[1] Globalwebindex, "The world of gaming." https://www.globalwebindex.com/reports/gaming-report, 2020. [Online: accessed 5-Aug-2020].

[2] N. Barman and M. G. Martini, "QoE Modeling for HTTP Adaptive Video Streaming–A Survey and Open Challenges," *IEEE Access*, vol. 7, pp. 30831–30859, March 2019.

[3] N. Barman, *An objective and subjective quality assessment for passive gaming video streaming*. PhD thesis, Kingston University London, 2019.

[4] S. Zadtootaghaj, N. Barman, S. Schmidt, M. G. Martini, and S. Möller, "NR-GVQM: A No Reference Gaming Video Quality Metric," in *2018 IEEE International Symposium on Multimedia (ISM)*, pp. 131–134, IEEE, 2018.

[5] S. Göring, R. R. R. Rao, and A. Raake, "nofu — A Lightweight No-Reference Pixel Based Video Quality Model for Gaming Content," in *2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*, pp. 1–6, 2019.

[6] N. Barman, S. Zadtootaghaj, S. Schmidt, M. G. Martini, and S. Möller, "GamingVideoSET: A Dataset for Gaming Video Streaming Applications," in *2018 16th Annual Workshop on Network and Systems Support for Games (NetGames)*, pp. 1–6, 2018.

[7] N. Barman, E. Jammeh, S. A. Ghorashi, and M. G. Martini, "No-reference video quality estimation based on machine learning for passive gaming video streaming applications," *IEEE Access*, vol. 7, pp. 74511–74527, 2019.

[8] M. Utke, S. Zadtootaghaj, S. Schmidt, S. Bosse, and S. Möller, "Ndnetgaming-development of a no-reference deep cnn for gaming video quality prediction," *Multimedia Tools and Applications*, pp. 1–23, 2020.

[9] S. Bosse, D. Maniry, K.-R. Müller, T. Wiegand, and W. Samek, "Deep neural networks for no-reference and full-reference image quality assessment," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 206–219, 2018.

[10] S. Bosse, S. Becker, K.-R. Müller, W. Samek, and T. Wiegand, "Estimation of distortion sensitivity for visual quality prediction using a convolutional neural network," *Digital Signal Processing*, vol. 91, pp. 54–65, 2019.

[11] R. R. R. Rao, S. Göring, P. Vogel, N. Pachatz, J. J. V. Villarreal, W. Robitza, P. List, B. Feiten, and A. Raake, "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*, 2019.

[12] S. Zadtootaghaj, S. Schmidt, S. Shafiee Sabet, S. Möller, and C. Griwodz, "Quality estimation models for gaming video streaming services using perceptual video quality dimensions," in *Proceedings of the 11th International Conference on Multimedia Systems*, ACM, 2020.

[13] "Netflix Public Dataset." https://github.com/Netflix/vmaf/blob/master/resource/doc/datasets.md. [Online: Accessed 06-June-2020].

[14] C. G. Bampis, Z. Li, I. Katsavounidis, T.-Y. Huang, C. Ekanadham, and A. C. Bovik, "Towards Perceptually Optimized End-to-end Adaptive Video Streaming," *CoRR*, vol. abs/1808.03898, 2018.

[15] ITU-T Rec. P.918, "Dimension-based subjective quality evaluation for video content," Jan 2020.

[16] N. Barman, S. Schmidt, S. Zadtootaghaj, M. G. Martini, and S. Möller, "An evaluation of video quality assessment metrics for passive gaming video streaming," in *Proceedings of the 23rd Packet Video Workshop*, pp. 7–12, ACM, 2018.

[17] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

[18] F. Chollet, "Keras." https://keras.io, 2015.

[19] N. Barman, S. Zadtootaghaj, M. G. Martini, S. Möller, and S. Lee, "A Comparative Quality Assessment Study for Gaming and Non-Gaming Videos," in *Tenth International Conference on Quality of Multimedia Experience (QoMEX)*, (Sardinia, Italy), May 2018.

[20] ITU-T Rec. P.910, "Subjective video quality assessment methods for multimedia applications," April 2008.

[21] scikit video, "Video processing in python." http://www.scikit-video.org/stable/. [Online: accessed 5-Aug-2020].