IEEE Access
Multidisciplinary Rapid Review Open Access Journal

Received February 9, 2021, accepted February 15, 2021, date of publication February 16, 2021, date of current version March 1, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3059932

Modular Framework and Instances of Pixel-Based Video Quality Models for UHD-1/4K

STEVE GÖRING[1], RAKESH RAO RAMACHANDRA RAO[1], BERNHARD FEITEN[2], AND ALEXANDER RAAKE[1], (Member, IEEE)

[1]Audiovisual Technology Group, Technische Universität Ilmenau, 98693 Ilmenau, Germany
[2]Deutsche Telekom AG, 10117 Berlin, Germany

Corresponding author: Steve Göring (steve.goering@tu-ilmenau.de)

This work was supported by the Open Access Publication Fund of the Technische Universität Ilmenau.

ABSTRACT The popularity of video on-demand streaming services increased tremendously over the last years. Most services use http-based adaptive video streaming methods. Today's movies and TV shows are typically recorded in UHD-1/4K and streamed using settings attuned to the end-device and current network conditions. Video quality prediction models can be used to perform an extensive analysis of video codec settings to ensure high quality. Hence, we present a framework for the development of pixel-based video quality models. We instantiate four different model variants (**hyfr**, **hyfu**, **fume** and **nofu**) for short-term video quality estimation targeting various use cases. Our models range from a no-reference video quality model to a full-reference model including hybrid model extensions that incorporate client accessible meta-data. All models share a similar architecture and the same core features, depending on their mode of operation. Besides traditional mean opinion score prediction, we tackle quality estimation as a classification and multi-output regression problem. Our performance evaluation is based on the publicly available AVT-VQDB-UHD-1 dataset. We further evaluate the introduced center-cropping approach to speed up calculations. Our analysis shows that our hybrid full-reference model (**hyfr**) performs best, e.g. 0.92 PCC for MOS prediction, followed by the hybrid no-reference model (**hyfu**), full-reference model (**fume**) and no-reference model (**nofu**). We further show that our models outperform popular state-of-the-art models. The introduced features and machine-learning pipeline are publicly available for use by the community for further research and extension.

INDEX TERMS Quality Assessment, quality of experience, video quality, full reference, no reference, hybrid video quality models, UHD-1/4K, video streaming, machine learning.

I. INTRODUCTION

Considering the enormous increase of uploaded, watched and shared videos, it is not a surprise that approximately 70% of the overall internet bandwidth is spent for video streaming [14], and this is projected to increase to about 80% to 90% by 2022 [13]. Today's video streaming uses http-based adaptive streaming (HAS) such as dynamic adaptive streaming (DASH) [61] to distribute video contents to the end users. The core idea of HAS is to automatically adapt the played video quality to the used end device and in particular to the available network bandwidth, to avoid stalling of video play out due to buffer depletion, and continuously play out the video at the highest possible quality even in low bandwidth situations. To enable such an adaption, it is required to store several representations on the server. Each representation is usually segmented into smaller portions of the video, with a range of 2-10 seconds [61] each, so that the client can smoothly switch to another representation during play out. Technically it is further required to have meta-data stored to assemble streams, usually done in a manifest file stored on the server. Depending on the used approach, the manifest file can also include representation headers. As another application, DASH is further used for livestreaming of broadcast video content [18], which shows that this technology is quite generic. Moreover, different adaptation strategies or algorithms are investigated to improve quality of experience of users during video streaming [84], especially because the server back-end is based on http, and does not require additional intelligence for adaptation at server level. There are

The associate editor coordinating the review of this manuscript and approving it for publication was Charith Abhayaratne.

efforts to also increase the tasks of the back-end server, e.g. using back channel data to specify different encoding parameters, or to collect and monitor quality-related factors, see [19], to improve streaming efficiency and stability by enabling low latency and faster adaptation to bandwidth fluctuations.

Currently, there has been an increase in the usage of 4K TV screens by end customers, and in addition 8K screens are also available [76]. Furthermore, popular video streaming providers such as Netflix [56], Youtube or Amazon Prime Video are supporting 4K or even higher streaming resolutions. Even at the recommended viewing distance of 1.5 or 1.6 times the height of the display for 4K content (maybe even closer for 8K) and thus with visual angle per pixel below visual acuity of approximately $1'$, see [30]–[32], [82], it can still be quite challenging for users to perceive differences between videos at such very high resolutions, for example between FHD (1920 × 1080) and UHD-1/4K (3840 × 2160 or 4096 × 2160). For this reason, Kara *et al.* [42] analyzed the effects of labels on the perception of 4K content, and showed that most users will not be able to see a difference between FHD and 4K content, with similar results being presented in the study conducted by Berger *et al.* [10]. Moreover, in [27], it is analyzed whether people see a difference between FHD and UHD-1/4K for uncompressed videos without additional labels. In their work, Göring *et al.* [27] show that there is only a perceivable difference for about 50% of the considered videos. This is the result of both the characteristics of the recorded scene and the camera system and production settings used. Since a clear conclusion on the suitability of the usage of 4K based on specific content features was not directly possible, the authors trained a machine learning system using several pixel-based features to classify videos in terms of whether viewing in 4K resolution can be distinguished from the less resource-demanding FHD alternative. Similar analyzes regarding video source resolution have been performed by Katsavounidis *et al.* [44] to evaluate the native video resolutions. In general, 4K or UHD-1 videos show benefits if the scenes are slow and with a lot of details, however, the content has a huge impact on the perceived video quality, which is also the conclusion of VanWallendael *et al.* [86]. Thus, it follows that video quality models should also consider more content diversity, for example for higher resolution videos.

Moreover, in streaming situations with newer video codecs, e.g. AV1 [1] or VVC [34], it is required to have a proper understanding of the video quality subjectively perceived by viewers. It is especially important when taking into account that today's video streaming platforms use more optimized encoding settings, and that viewing strategies and also user's expectations and hence quality perception have changed. The automatic encoding optimization can be performed per title or even per scene or shot of a given video. For example, Netflix now uses a scene-optimized encoding [43]. The main goal of encoding optimization is to deliver high quality video material to users having low internet bandwidth or experiencing strong bandwidth fluctuations. For such optimization, video quality models of high accuracy are required. For example, Netflix uses its own video quality metric VMAF [57] in its optimization pipeline. However, VMAF does not include a dedicated handling of framerate variation [69], and in case of 4K it is not clear for which video codecs it has been trained [58]. Moreover, it also does not include any long-term analysis of video quality suitable for video streaming sessions longer than the 5 to 10 s typically used for video-quality development. Hence, there is room for improved video quality models for the case of 4K adaptive streaming. VMAF is a full reference (FR) model, where the reference and distorted videos are required as input. In practical use cases, e.g. livestreaming, it is not possible to have a proper reference video stored, making such FR models less appropriate for this type of applications. Recently, there has been work within ITU-T Study Group 12 on standardizing a set of different short-term video quality models as the new standards series Rec. P.1204 [37], [67]. There, also a pixel-based, full/reduced reference model and a hybrid no-reference model have been standardized as Recs. P.1204.4 [38] and P.1204.5 [72], respectively. While the mentioned models enable high-accuracy and -precision quality predictions [67], they are not based on a common, modular framework that enables video-quality predictions in a scalable manner, adding features as they are available. This is what is provided with the present paper, as well as an open-source implementation of all model components, which can be used as the basis for further research.

To summarize the open points for video quality models in case of 4K streaming, the following research questions have been identified, which will be addressed in the remainder of the article.

- Can a common feature set and architecture be used to estimate video quality for several application scopes?
- Is it possible to develop no-reference pixel-based video quality models that have a comparable performance to full-reference models?
- Can pixel-based video quality models be extended by meta-data to improve performance?
- Can center cropping be used to speed up calculation with similar overall prediction performance?
- Are such models able to predict more than only mean opinion scores?

To answer the identified questions, we introduce pixel-based video features and a general model framework. We describe four instances of the framework, (1) a no-reference video quality model (**nofu**), (2) a hybrid no-reference video model (**hyfu**), (3) a full-reference model (**fume**), and (4) a hybrid full-reference video quality model (**hyfr**). Here, hybrid models use additional data such as which video codec, framerate, resolution and bitrate of a given distorted video. Such meta-data is typically available in the delivered manifest file that is required to implement the DASH play out. The paper describes the models in detail, as well as a number of evaluation experiments, where we show that our models are able to outperform

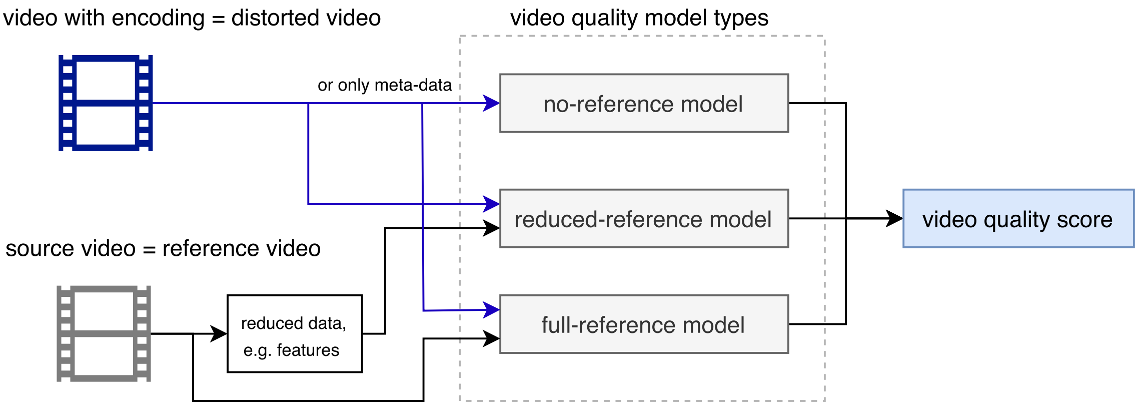


FIGURE 1. Video quality model types with their corresponding input data.

other state-of-the-art video quality models. All models follow the same architecture, thus they share similar or the same features, depending on the available input data, and use a machine learning pipeline to predict video quality. The used machine learning models consist of a feature selection step with an additional applied random forest step, however it should be mentioned that the introduced approach is not limited to the used machine learning algorithms. The modularity of the provided framework enables changing the employed machine learning algorithms. Furthermore, the source code for the features, model architecture,[1] pre-trained models[2] and evaluation datasets[3] are publicly available to enable extensions and usage for the research community. The published framework can be used for various problems in the context of video quality, e.g. genre classification [26] or other classification problems [27]. The main idea of the proposed models, is to evaluate whether such a modular framework can be used for video quality prediction considering UHD-1/4K. Moreover, we analyze to which extent meta-data can improve prediction accuracy, and how center cropping of the videos can be used to speeding up calculations. In contrast to state-of-the-art models, we additionally investigate different prediction targets than the usual mean opinion score.

The article is organized as follows. In the next Section II, we describe the state-of-the-art video quality models and outline limitations or open questions regarding modern video streaming applications. Afterwards, in Section III we describe our proposed models of different types, from no-reference to full-reference hybrid model instances. All models have in common that they use pixel-based data to estimate video quality perceived by end users. To develop more advanced video quality models, it is required to have valid, highly reliable, and carefully designed training and validation databases. For this reason, in the subsequent Section IV we describe the used datasets, detailing e.g. the video encoding conditions and corresponding subjective tests. Furthermore, we evaluate our developed models in several scenarios, e.g. prediction targets (mean opinion score prediction, quality as a classification and a multi-instance regression approach) and the used center cropping. In addition, we compare the model performance with other state-of-the-art video quality models, see Section V. Finally, we conclude the article with a review of our modelling results and of open aspects that are planned for future work.

II. OVERVIEW OF VIDEO QUALITY MODELS

Image or video quality models are typically divided into three main categories [7], [78]: no-reference (NR), reduced-reference (RR) and full-reference (FR), depending on the input data that is available for quality estimation. In Figure 1 an overview of the different video quality model types is shown, where each type has a different input used to predict a video quality score. For example, in case of a full-reference model, the distorted and reference video are fully accessible to the model. On the other hand, for no-reference models, only the distorted video or some meta-/bitstream data is used as input for the model. No-reference models can further be classified into pixel-based or bitstream-based models. In case of bitstream-based models, a full decoding of the given video is not required, consequently only statistics of the data stored in the bitstream itself can be used. A typical example for a bitstream-based video quality model is ITU-T P.1203 [36], [66], [75], where in total four different modes of operation are distinguished. P.1203 is a bitstream based model for adaptive streaming, thus it also requires typical input data of an adaptive video streaming session, i. e., duration of stalling events, quality switches and segment data. The four different modes only change the way how segment data is processed and how a video quality score is predicted for each segment. In the lowest mode 0, only meta-data is used (i.e. framerate, codec, bitrate, resolution). In mode 1, also frame sizes are additionally included, while in mode 3 all bitstream data is available, and for example specifically

[1] https://github.com/Telecommunication-Telemedia-Assessment/quat

[2] https://github.com/Telecommunication-Telemedia-Assessment/pixelmodels

[3] https://github.com/Telecommunication-Telemedia-Assessment/AVT-VQDB-UHD-1

selected QP (Quantization Parameter) values of single video frames are used to predict segment quality. Mode 2 is similar to mode 3 except that only a 2%-subset of all frame data is accessible for prediction of quality. Finally, after applying the mode-specific prediction of each transmitted video segment, P.1203 uses these video quality scores in combination with per-second audio scores, initial loading delay and stalling event information to aggregate an overall audiovisual quality score.

In general, two different aspects can be distinguished for DASH/HAS based video quality estimation. First, how the per segment video quality, which is usually referred to as the short-term video quality, is estimated. And as second, what is the overall audiovisual/video quality after a longer time including stalling, audio quality and more, referred to as long-term video quality. For example, ITU-T P.1203 [36] handles both cases in an integrated framework, where overall audiovisual quality can be estimated up to 5 minutes of video duration.

Moreover, recently the ITU-T P.1204 [37] standard has been approved. Models of this standard consider short-term video quality including H.264, H.265, and VP9 encoded videos up to UHD-1/4K resolution. Raake *et al.* [67] show that the proposed models can also be used for unknown datasets. The P.1204 models can be seen as an extension for the short-term video component of P.1203. In the remainder of the paper, we focus on the per-segment video quality estimation aspect of DASH/HAS.

In general, combinations of several model types are possible, e.g. combining bitstream- and pixel-based models that are usually referred to as hybrid models. In this article, we focus on pixel-based models, in addition we also consider hybrid models, where pixel-based data of a given video is combined with meta-data. For our models we focus only on mode 0 meta-data, where higher modes could be considered, too.

Considering the variety of different DASH/HAS streaming parameters, video quality depends on several factors, starting from various used video codecs, differently optimized encoding settings and corresponding bitrate-ladders, to a large range of video contents that are streamed, in higher resolutions and framerates. The existing set of models are far from comprehensive as yet. For example, Barman *et al.* [7] identified several open points, e.g. privacy, high model complexity, multiple influence factors on video quality perception and a limited handling of all of these, and even more. Thus, it can be concluded that video quality prediction is still a challenging task, based on a number of different influence factors that need to be considered in video quality models and their corresponding development process.

In the following sections, we briefly review some key NR, RR and FR models. We consider models that are capable of handling compression artifacts of modern video codecs especially for higher resolutions (4K or UHD-1) and framerates (up to 60 frames per second), even though not all of these models were explicitly developed for these conditions. We also describe some image quality models, which can be extended or are being used for video quality prediction.

A. NO-REFERENCE MODELS

The first type, no-reference models, are suitable for numerous practical use cases, due to the fact that they do not require any additional input data other than the distorted video. On the other hand, pixel-based no-reference models are usually not able to reach the same prediction performance as full-reference models, because they cannot compensate the missing data of the reference video. This reason also limits some possible applications of pixel-based no-reference models. As a consequence, for example, no pixel-based NR-model has so far been standardized by ITU-T SG12 or the Video Quality Experts Group (VQEG[4]).

1) BITSTREAM BASED MODELS

As already introduced, ITU-T P.1203 [36], [66], [75], is a bitstream based no-reference video quality model developed especially for adaptive streaming use cases. The model is trained on FHD videos of up to 5 minutes of video duration, whereas the encoding was performed using several bitrate, resolution and framerate settings using H.264. Considering that current video streaming providers, e.g. Netflix, Youtube, Amazon Prime video, use more recently developed video codecs for their video streaming and encoding strategies, P.1203 cannot directly be applied to such new codecs. For this reason, Rao *et al.* [68] propose a method to extend P.1203 to modern codecs for mode 0, namely AV1, H.265 and VP9. Besides inclusion of modern video codecs, the extension also enables P.1203 to handle higher resolutions and framerates up to 60 frames per second (fps). The extension only covers the short-term video quality model of P.1203 that predicts video segment scores and assumes that the overall audiovisual integration does not change. Considering that mode 0 models do not have any knowledge about the underlying content, the proposed extension can just be seen as a first starting point for future extensions of the standardization work.

To cover more video codecs, higher resolutions and framerates, the models from the newly standardized ITU-T P.1204 [37], [67] series can be used, which were developed for short-term video quality prediction. Here, ITU-T P.1204.3 is a bitstream based no-reference video quality model [39], with full access to the video bitstream. P.1204.3 uses several statistics that are extracted from the video bitstream [37], [70]. For example, statistics about motion vectors, quantization parameter, and frame sizes, covering H.264, H.265 and VP9. The model itself consists of two parts, a parametric part and a machine learning part. The parametric part is based on degradation-based modeling, similar to P.1203.1 mode 3 [36], [66], whereas the machine learning part uses random forest regression with feature selection to predict the residual not captured by the first, parametric part of the model. Rao *et al.* [70] use the AVT-VQDB-UHD-1 dataset [69] to

[4] https://www.its.bldrdoc.gov/vqeg/vqeg-home.aspx

perform an additional analysis of the model performance, with an implementation of the model being made publicly available.

2) NATURAL SCENE STATISTICS BASED MODELS

Beside bitstream-based no-reference models, pixel-based models have been proposed in the literature. Two examples are **brisque** and **niqe**, which both are part of scikit-video.[5] In scikit-video, only the feature extraction of **brisque** and **niqe** are included, the final model is usually a support vector machine or regressor (SVM/SVR) [53], [54] which uses the extracted features as input. Both methods are independent of distortion-specific assumptions, and focus on measuring differences in naturalness of the given input image. This is realized using statistics of normalized luminance coefficients to measure the differences to undistorted images using a natural scene statistic model. **niqe** only extracts one value, whereas **brisque** extracts 36 different feature values. Using the extracted features, it is possible to train well performing image or video quality models, as it is shown in [22] for images and [25] for 4K videos. Even for streaming quality of gaming videos or sessions, these models can be applied and show promising results [6], [8], [23]. However, to apply them for such video quality prediction, a suitable machine learning model needs to be trained, where in addition ground truth values per video frame are required. At the core, video-specific effects due to motion inside the video or corresponding masking are not captured in these model. In general, **brisque** and **niqe** can also be used as features to develop new models, i.e. combined with motion related measurements. A drawback of such the usage of **brisque** and **niqe** or similar approaches is that a retrained machine learning model requires a suitable ground truth. In addition, the features were also not specifically developed to handle high resolution images or videos. However, it was already shown that both features in combination show promising results even in case of 4K video quality prediction [25]. For this reason we will include a **brisque+niqe** baseline model as comparison in our evaluation, see Section V. Another natural scene statistics based model is BIQI [55]. BIQI is a no-reference distortion independent image quality metric, which uses an SVM similar to **brisque** and **niqe** for final score prediction. However, BIQI is only evaluated on low resolution images based on the LIVE IQA [80] dataset.

3) DNN-BASED MODELS

Beside classical signal driven video quality models, models based on deep learning can also be used to estimate video or image quality or encoding optimization [45], [46], [51]. Most DNN-based quality assessment models share similar approaches. For example, VeNICE [15], the models of Bosse *et al.* [11], [12], Deviq/Deimeq [22], [25], or Wiedemann *et al.* [90] all use some variant of local patch quality estimation. In general, using transfer learning, a pre-trained DNN is applied to perform the quality evaluation task on a per-frame basis. The usage of transfer learning reflects the fact that the ground-truth data typically is too sparse so as to develop a full DNN for image or video quality prediction. For example, in case of VeNICE [15], the VGG16 [81] network is used, similar to Bosse *et al.* [11], [12], whose DNN-based quality model also operates with the VGG network. The model Deviq/Deimeq [22], [25] uses Xeption or Incpetion. Usually these pre-trained DNNs are developed for image classification tasks, and are used in the models as a feature extractor for image quality. In such cases specific layers of the DNNs are used as features and are combined or retrained to predict image quality. It was already analyzed which DNNs are more suitable for image quality evaluation [22]. However, especially for high-resolution videos or images, DNN-based processing is time-consuming, and also retraining is not a straight-forward task, due to the high amount of data that needs to be handled. Moreover, it is not completely clear that for a patch-based training the overall quality score of a frame can be assumed. This is shown for example in [90], indicating that quality scores for local patches can be used to estimate global image quality, however, for some other patch-based models, the opposite conclusion is reported. One no-reference model for video quality is Deviq [25], which handles the mentioned high-resolution problem using hierarchical sub-images to reduce the overall number of patches. In contrast to other approaches, where the last layer of the DNN is replaced by new layers, Deviq's final prediction is performed with an approach based on random forests (RFs) including a feature selection step. The reason for this is mostly due to the fact that RF models are faster to train, and that the DNN is only used as feature extractor. Moreover, a similar approach for no-reference image quality is Deimeq [22], where the main focus is to analyze which DNN is most suitable for image quality prediction. It can be concluded, that the complexity of the DNN has an influence on the ability to transfer the DNN to another image related task, mainly because such models are specifically optimized for the image classification task. Thus, e.g. faster models like Mobilenetv2 [77] or VGG16 [81] are not fully suitable for image quality, and on the other hand, complex models like Xception and Inception are even able to have better performance than signal based models [22]. Today, DNNs are used for several image related tasks and are usually able to outperform traditional methods. However, these DNN-based models are slower for higher resolution images than usual approaches, which is why for our models we focus on traditional signal-based features that perform fast even for higher than 4K resolution videos.

One of the main problems for frame-based video quality models is, that it is hard to obtain subjective video quality scores for individual frames in case of video streaming. A common solution is that image quality models are developed in a first step and later are applied in similarly to video quality prediction. However, it is mostly not fully covered how in such a case motion-related effects change

[5] http://www.scikit-video.org

video quality perception. On the other hand, subjective tests and models based on continuously rated quality scores have been proposed [4], using a slider for the continuous rating of quality over time. It can be assumed that with this setup, several influence factors can lead to different quality scores over time, e.g. if participants are lazy to move the quality rating slider, or if the current quality decision is too biased of previously shown frames. Moreover, rating sliders also cannot directly enable a per-frame quality scoring and hence model-based estimation, because usual videos have several frames per second and rating is performed with temporal delay. For no-reference video quality models, there is another possibility to get ground truth data on a per frame level. For example, per frame scores can be estimated using a suitable full-reference video quality metric, e.g. VMAF [6], [25]. A drawback of this approach is that the scores are based on a different model, and thus the overall performance of the new model depends on the ability of the used full-reference score to measure quality variation over time.

4) MODELS FOR OTHER USE CASES

Beside classical video streaming, there are other video contents streamed using DASH or HAS, for example 360° video or videos of gaming sessions. Due to the fact that such scenarios include different properties of the given content, it is required to develop or use content- or use-case-specific models. In case of 360° video, it was already shown that existing models like VMAF are able to perform quite well [59], if the equi-rectangular projection scheme is used, or that even meta-data and hybrid models can be applied [20]. Similarly for gaming sessions, VMAF has been reported to show good performance [8]. However, especially in the context of gaming, full reference models are hard to apply, due to the specific live-encoding of the gaming content during the gaming session. Thus even though full-reference models could be used, in most application scenarios they are not feasible, because users are not desired to use a lot of additional computing resources, so fast no-reference models would be more suitable.

For example, in [6], Barman *et al.* uses fifteen signal-based no-reference features to build video quality models for gaming video streams. The overall pipeline employs per-frame estimated VMAF-scores as ground truth to train a per-frame quality prediction component. The aggregation of the individual features is performed using a Support Vector Regression (SVR) approach. Moreover, subjective scores are also considered for overall video quality estimation. It is shown that such application- and content-specific models are able to outperform other no-reference models, and reach results comparable with full-reference models. Similarly, the NDNetGaming model [85] proposed by Utke *et al.* uses image-based DNNs to predict image quality at a per-frame level using several patches, where the ground truth for each frame is based on VMAF-scores, combined using a final aggregation to a video quality score.

With a similar goal, we adapted one of our models to the context of gaming QoE. In [23], we propose a gaming-specific version of our **nofu** model (see Section III-D1), which uses a subset of the features of the original **nofu** model to take into account the peculiarities of gaming content, and predict video quality in case of gaming streams. It is shown that **nofu** is able to outperform a **brisque+niqe** retrained baseline model, and that it achieves promising results in comparison with the full-reference VMAF. However, it needs to be noted that especially gaming videos share similar properties, e.g. computer generated textures, different motion patterns, static head up displays. Consequently, it is not clear if such models perform similarly with general 2D video content.

In addition, bitstream based models can also be applied to predict the quality of gaming videos. For example, Rao *et al.* [71] evaluate the performance of the recently standardized ITU-T P.1204.3 model and a retrained variant thereof for several gaming-specific video quality datasets. In addition to GamingVideoSET and KUGVD, also a Cloud Gaming Video Dataset (CGVDS) [97] and a dataset based on Twitch are considered, showing promising results. Moreover, it was shown that the ITU-T P.1203.1 model can be applied to gaming videos [97].

All Gaming-QoE models use similar or even the same underlying dataset, e.g. GamingVideoSET [9] or KUGVD [6], where the used videos have a maximum resolution of FHD with 30 frames per second. This is a limitation due to the specific application use case of such models, because recordings of gaming sessions require more hardware resources, and even many games do not provide higher resolution textures. However, it shows that no-reference models in principle can reach good performance in case of quality prediction for gaming sessions. Moreover, also models have been proposed to bridge traditional videos and gaming videos [96], with Zadtootaghaj *et al.* describing a model consisting of several steps. Here, for example the first step trains a convolutional neural network to estimate blurriness and blockiness, and later it is trained with encoded videos to fine tune the network. Afterwards, a random forest model uses the predictions of the neural network to estimate quality.

B. REDUCED-REFERENCE MODELS

A special case of video quality models are reduced-reference models. They share properties with full-reference models, e.g. that they require access to the source, i.e. reference video. Source video properties are usually extracted before the distorted video is processed. On the other side they are similar to no-reference models, considering that they only have a limited knowledge of the source video, thus a no-reference model could be seen as a reduced reference model without any knowledge of the source video. The approach of a reduced-reference model is that in a first step the source video is processed, and as an output, reduced data of the source video is stored. Such reduced data is based on signal features, sampling or similar characterization of the source video. Accordingly, all models that are based on features extracted from the reference, and not on full pixel information, can be referred to as reduced reference. In general, reduced-reference models increase the prediction accuracy of no-reference models, with their inclusion of side information from the source video. Two examples for such models are SpeedQA [3] and STRRED [83]. SpeedQA [3] is based on spatial efficient entropic differencing for quality assessment and STRRED [83] uses spatial and temporal entropic differences. Another reduced-reference video quality model is ITU-T P.1204.4 [38] that is based on edge statistics of the distorted and reference video to estimate video quality.

Our focus in this article are no-reference, full-reference and hybrid models, however some of our features and the model pipeline can be also used to develop reduced-reference video quality models.

C. FULL-REFERENCE MODELS

Compared to no-reference models, a full-reference model has full access to both the distorted and source video sequence pixel information. The simplest full-reference image quality model is Peak-Signal-To-Noise-Ratio (PSNR), where a pure signal-based difference is estimated. It is well known that PSNR does not match human perception and video quality evaluation, both in general and especially in case of higher resolution [25], [69], [87]. Beside the classical PSNR, a measure that is also used as quality metric in several applications is an extension of PSNR called the PSNR-HVS [17]. PSNR-HVS takes properties of the Human Visual System (HVS) into account. For this, PSNR-HVS is based on a similar fundamental equation as PSNR, however the calculations are done blockwise using DCT coefficients with weighting and correction factors to include contrast perception. With the mentioned extension, PSNR-HVS is able to outperform PSNR and MS-SSIM in case of image quality prediction for several distortion types [17]. However, using PSNR-HVS in case of video does not include specific video motion distortions, or high resolution related aspects. There are other extensions of PSNR available, e.g. X-PSNR [29] or for color CQM [93]. X-PSNR [29] is a low complexity extension of PSNR, that uses a block-wise weighting approach, and CQM [93] is variant of PSNR where the overall score is a weighted sum of PSNR for luminance and chroma channels.

Most video quality models have their origin in image quality estimation, such as Structural Similarity Index Measure (SSIM) [88], [89] or Visual Information Fidelity (VIF) [79]. In spite of their somewhat better representation of the information the HVS extracts from images, VIF and SSIM also show only low prediction performance in case of high resolution videos, as reported in [21], [25], [69].

Netflix's VMAF (Video Multimethod Assessment Fusion) [50], [57] is a video quality model that is based on a combination of different image quality models. It is open source and includes a trained model for 4K video quality prediction [56], [58]. VMAF is based on two full-reference models, namely VIF [79] (4 scales) and ADM2/DLM [49], In addition to per-frame image-based quality features, it also includes a simple motion estimation feature that is based on differences to a previously played video frame. VMAF can be used to estimate 4K video quality, and it shows quite good prediction accuracy even for newly conducted video quality tests [25], [69].

As features, VMAF extracts several image quality scores per frame, and in addition one motion feature. All per-frame values are later aggregated with a Support Vector Regression (SVR) model. The SVR is trained to merge all features into one quality score. The baseline non-4K enabled model is trained on the publicly available Netflix public dataset, including several videos up to FHD resolution with 30 frames per second. In contrast, the 4K videos that are used for training the 4K model version are not available. Based on the per-frame video quality scores provided by VMAF, the overall video quality can be calculated using several methods, from simple averaging to harmonic mean, or running several models to further estimate a prediction confidence interval. Such an approach is suitable for short-term video quality prediction. In turn, for longer-term video quality estimation, where besides a given set of segment quality levels also stalling or quality switches can occur, other integration approaches are required. In general VMAF does not include such aspects and is therefore less suitable for long-term video quality prediction.

D. HYBRID MODELS

Besides pure bitstream- or pixel-based video quality models, combinations of models are possible, that are usually summarized as hybrid models [5], [92]. For example, it is possible to use a no-reference pixel-based video quality model and extend the available input by using meta-data that pertains to bitstream-based models. To describe the additional bitstream data, it is possible to use the modes that are defined for bitstream based models in the series ITU-T P.1203 and P.1204. For example, part of P.1204 is a hybrid no-reference mode 0 model (P.1204.5 [72]), which uses meta-data, that are accessible at the client side, and combines such features with a pixel-based, no-reference video complexity feature. The complexity feature uses a recorded version of the played video and is based on the file-size of the re-encoded recording. In a similar approach Yamagishi *et al.* [94] proposed a model for IPTV, extending a meta-data based model by content complexity, using the Spatial and Temporal Information (SI, TI) described in [41].

E. SUMMARY

We briefly described several no-, reduced- and full-reference models. While most of the models were not developed explicitly for UHD-1/4K resolution, they can still be applied for such higher resolutions. Accordingly, different studies have found that some of these models also show a good prediction performance. However, it is also clear that models specifically addressing the target of higher resolutions will perform

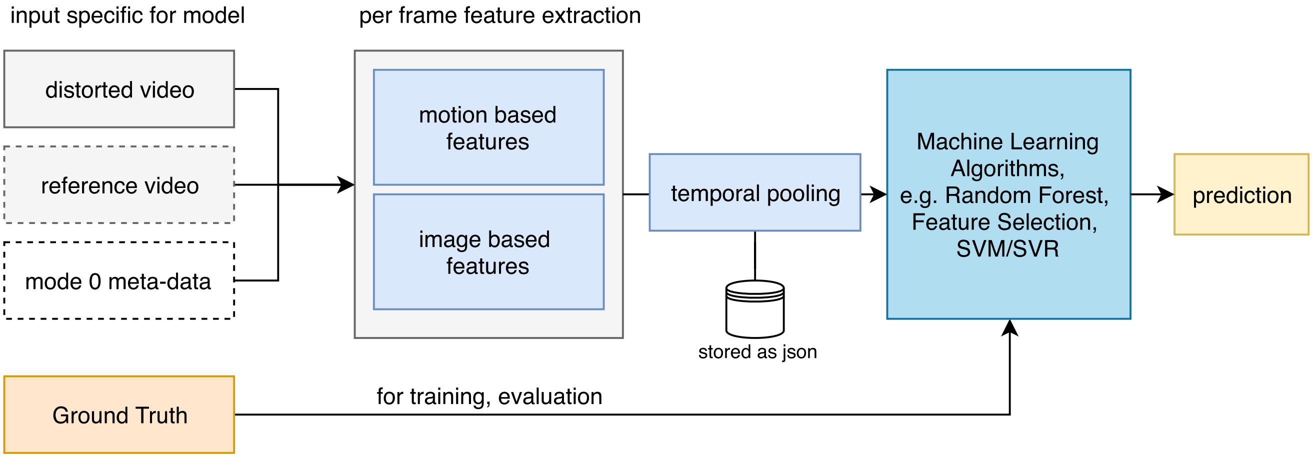


FIGURE 2. **General Video Quality Model structure consisting of feature extraction, temporal pooling and machine-learning-based model training or prediction.**

better in predicting subjective quality. In addition, only a few models, such as VMAF, are capable to predict more than a pure quality score. For example, VMAF can be used to predict confidence intervals of several trained predictors of the model, to further evaluate the prediction accuracy or the underlying individual user ratings of the given video. However, there are additional approaches possible, for example the prediction of a rating distribution or only a quality class. Both extensions are possible with our introduce model framework and will be described later in detail. In addition, it should be mentioned that our framework includes even more features and approaches that are used and described within this paper, to enable researchers to develop models for various research problems in the context of video quality.

III. PROPOSED VIDEO QUALITY MODELS

To tackle the problem of video quality estimation with different types of available input data, we developed several pixel-based video quality models. All models behave similarly, moreover, they share specific features and conceptual parts in a common framework. In Figure 2, the general structure of the video quality models is illustrated. Usually the distorted video and reference video have the same input resolutions, pixel format and framerates, otherwise before applying our model a conversion is performed to ensure this condition. First, depending on the given input data that can be accessed, features are calculated only from the distorted video (no-reference), from distorted and reference (full-reference), or including some additional meta-data. In general, the features can be categorized into two groups, first, motion-based features, and second, image-based features. All implemented features and training code are part of *quat*[6] and the specific instances are part of *pixelmodels*.[7] Both the general framework and the instances are publicly available. Most features are calculated on a per-frame basis, which leads to the requirement of pooling to estimate a time-independent set of feature values. For this reason we select advanced temporal pooling, a method that includes several statistical pooling approaches, and that we already used to solve different video quality research problems [23], [27].

As a last general step, all pooled features are used to train a machine learning algorithm. In our case we use a random forest model (120 trees for a no-reference and 240 for a full-reference model) with a previously applied feature selection step using the ExtraTreesRegressor algorithm. The number of trees for all models has been evaluated using 10-fold-cross validation in several additional training runs. Our implementation is based on Python 3 and uses scikit-video[8] for video processing and scikit-learn [62] for all machine learning parts. However, it should be mentioned that our introduced models are not restricted to the used machine learning algorithms. We further analyzed different algorithms, e.g. SVR, Gradient Boosting Regression (GBR), …, and all lead to a similar performance. Here, RF models showed stable performance for all four model instances. After training the machine learning model using the subjective scores included in the database, we are able to analyze the prediction accuracy of our model. To this aim, we use several commonly evaluation performance metrics, e.g. for the MOS prediction scenario, i.e. Pearson Correlation Coefficient (P or PCC), Spearman's Rank Correlation Coefficient (S), Kendall Rank Correlation coefficient (K) and root mean square error (RMSE).

In the following subsection, we describe the individual parts of our model structure in more detail. We start with the pixel-based features, describe further details regarding speedup of calculations, temporal pooling, and finally conclude with different instances of our general model pipeline.

A. FEATURES AND MOTIVATION

Considering that video distortions introduced in the video signal are heavily dependent on specific encoding settings and the used codec, it is required to also have several features

[6] https://github.com/Telecommunication-Telemedia-Assessment/quat

[7] https://github.com/Telecommunication-Telemedia-Assessment/pixelmodels

[8] http://www.scikit-video.org/stable/

handling such effects. In addition, also masking effects can have a strong influence on perceived video quality [73]. To describe the effects that are the reasons for the final quality rating of a user, we group our features into two general sets, namely motion-based (**mov**) and image-based no-reference features (**img**). Further, we include several other features, e.g. image full-reference features (**img-fr**). To enable our models to use bitstream or meta-data, we include bitstream specific features (**bs**). Table 1 summarizes all features of our model pipeline, moreover also references to the source of the given features are provided. Features marked with **own** are features we have developed ourselves. It is noted that each feature produces either per-sequence values (e.g. in case of bitstream features) or per-frame values. Further, we added **brisque** as additional features in our table, it will only be used for one specific model.

Some of our own implemented features were already used in different video quality related research directions, for example for gaming video quality [23] or automatic estimation of the perceivable differences of UHD and HD [27].

1) PER-FRAME NO-REFERENCE FEATURES

We developed or re-implemented several features that are calculated on a per-frame basis. For example, *colorfulness* [28], *tone* [2], and *saturation* [2] are features that are already used in image aesthetics prediction, which we reimplemented based on the published work. The rationale behind including aesthetics features is that usual video content is getting more and more diverse, so especially liking aspects are also influencing user's perception. Moreover, a similar argumentation follows for our *contrast* feature, that we estimate using histogram equalization. We use the normalized average difference before and after correction of the histogram based on the cumulative distribution function (CDF). Furthermore, spatial and temporal information are additional factors influencing video quality, for example comparing UHD with HD, usually spatial information is increased. For this, we use our implementation[9] of the SI and TI measure, in the following referred to as *si* and *ti*, that is based on ITU-T Recommendation P.910 [41].

Beside *si* or *ti*, videos are rescaled during encoding to lower resolutions to save bandwidth, such rescaling introduces degradations in sharpness, or adds additional blurriness. Usually users rate lower, if the images or videos lack sharpness. For this reason, we implemented a blurriness feature *blur* that is based on Laplacian variance. Each frame is converted to a grayscale image and afterwards a bilateral filter is applied to remove some noise. As the last step, a convolution with a 2D Laplacian filter kernel is performed. Based on the result, we estimate our blurriness score. As another way to recover some information about rescaling, we re-implemented an *fft* feature, that is based on [44]. With a similar motivation, especially for models that have no access to the native distorted video resolution, we measure the similarity to the rescaled HD frame as *uhdhdsim*, using PSNR as criterion. Here, for example a UHD-1/4K frame is rescaled to HD resolution (half of the input resolution) and upscaled to 4K (to the origin resolution), afterwards PSNR is calculated for the rescaled and non-rescaled frame. In addition to typical blurriness degradations, also blockiness can be observed in case of a badly selected encoding setting or a "fast" preset of the used encoder, for example in case of livestreaming. To measure block artifacts introduced due to high or suboptimal compression as in a live context, we developed a measure for *blockiness*. There are already features to measure blockiness reported in the literature [63], [65], however these features usually assume a fixed block size and are developed for JPEG compression. To overcome these limitations, we decided to develop an own feature, that shares some of the general ideas of the aforementioned blockiness estimation approaches. In general our feature checks commonly used block sizes and

TABLE 1. Overview of all included Features; # of values are either per frame (/F) or per video sequence (/S); a * marks features that are re-implemented.

Feature	Feature Type	Source	#Values
contrast	img	own [27]	1/F
fft	img	[44]*	1/F
blur	img	own [27]	1/F
colorfulness	img	[28]*	1/F
tone	img	[2]*	1/F
saturation	img	[2]*	1/F
scene_cuts	mov	own	1/F
movement	mov	own [27]	1/F
temporal	mov	own [27]	1/F
si	img	[41]	1/F
ti	mov	[41]	1/F
blockmotion	mov	own [23, 27]	3 /F
cubrow.0	mov	own [23]	1/F
cubcol.0	mov	own [23]	1/F
cubrow.1.0	mov	own [23]	1/F
cubcol.1.0	mov	own [23]	1/F
cubrow.0.3	mov	own	1/F
cubcol.0.3	mov	own	1/F
cubrow.0.6	mov	own	1/F
cubcol.0.6	mov	own	1/F
cubrow.0.5	mov	own	1/F
cubcol.0.5	mov	own	1/F
staticness	mov	own [23, 27]	1/F
uhdhdsim	img	own [27]	1/F
blockiness	img	own [23]	1/F
noise	img	[16]	1/F
PSNR	img-fr		1/F
SSIM	img-fr	[88, 89]	1/F
VIF	img-fr	[79]	4/F
fps_est	mov-fr	own	1/F
framerate	bs		1/S
bitrate	bs		1/S
codec	bs		1/S
resolution	bs		1/S
bpp	bs		1/S
bitrate_log	bs		1/S
framerate_log	bs		1/S
resolution_log	bs		1/S
framerate_norm	bs		1/S
resolution_norm	bs		1/S
brisque	img-nofu	[53]	36/F

[9] https://github.com/Telecommunication-Telemedia-Assessment/SITI

for a given blocksize b it estimated edges of the current frame. Based on the edges it measures mean differences in horizontal and vertical orientation, assuming that if there are blocks in a frame, that each b-th row/column has a different edge distribution compared to the overall frame. A more detailed description is presented in [23].

Video shots or scenes are mostly characterized by including some kind of motion, for example a moving object, or resulting from a moving camera. Hence, we also include motion-related features in our model pipeline. As a first feature, we use a motion estimation approach that calculates the RMSE to the previously played frame. This feature is referred to as *temporal*. It shows a similar behavior as *ti*, however still some differences can be observed. Moreover, to handle foreground and background motion, we use a foreground background segmentation algorithm of OpenCV (see [99], [100]). Focusing on the foreground object, the percentage of the moving area is used as motion indicator in our *movement* feature. Similar to a video codec, we also use a blockmotion estimation algorithm – *blockmotion*, that is part of scikit-video. In our implementation, we use the SE3SS search method, and use 10% of the video height as blocksize to speedup calculations. Moreover, after extraction of moving blocks, we count, for all directions, how often a moving block was identified [23], [27].

Similar to what is described in [52], we further developed motion features with a more global view. To this aim, we use a sliding window of 60 frames, that usually corresponds to about 1 second of the given video. This window is then later handled as a cuboid, where we slide several planes to estimate motion aspects. For example, the *cubrow* features handle row slices of the cuboid, where *cubrow.p* refers to the used single pixel p percent height of the cuboid. Accordingly, *cubcol* is defined in an analogue way for columns.

We considered videos that include motion. However, some videos are quite static, and to handle such cases, we include a staticness measure *staticness*. For this reason, we calculate a mean frame based on all currently played frames. If the video is mostly static, the estimated mean frame includes a lot of spatial information. That is why we use, as final feature value, the SI measure of the current mean frame.

In addition to staticness of the video, we further calculate the amount of noise within a given video frame as *noise*. This feature uses a wavelet-based estimator for noise [16].

To further analyze a given video, we check how many scene cuts a video shot has. Our feature *scene_cuts* uses resized 360p views of the given video frames and performs a threshold-based detection for scene cuts, similar to the method implemented in scikit-video, see [60], [95].

All features that we described so far are classical no-reference features, thus a reference video is not required to perform the calculation. In case of a full-reference model, such features can be applied on the distorted and reference video. Moreover, also differences of feature values comparing distorted and reference video are considered in our model pipeline.

2) FULL-REFERENCE FEATURES

To include typical full-reference aspects, we further use some traditional full-reference image metrics, namely PSNR, SSIM [88], [89] and VIF [79]. In the development stage, we used a higher number of full-reference metrics, however there was no noticeable increase in performance. In a pure full-reference scenario, where the distorted video is e.g. recorded with a fixed framerate the model does not know which framerate the transmitted distorted video has. To handle this missing information, we developed a framerate estimation feature *fps_est*. It compares frames of the distorted and reference video in a sliding window of $w = 60$ frames, assuming that in case of a distorted lower fps, there are duplicated frames stored. Using RMSE of two consecutive processed frames for the distorted and references video as indicator, we check for the given window how many duplicated frames are presented. The final estimated number of frames is calculated using Equation 1, with ref_0 and dis_0 corresponding to the vector of RMSE values that are zero. In the beginning, the window size w is not fixed, and as overall feature we later pool several statistics so that the feature *fps_est* becomes quite robust.

$$fps(w) = |w| - |dis_0| + |ref_0| \tag{1}$$

3) BITSTREAM FEATURES

To handle hybrid mode 0 models, additional bitstream or meta-data-based features are required. For this reason, we extract meta-data of a given video file using ffprobe. Most important meta-data are framerate, bitrate, video height and width (resolution) and the video codec used. Including these features, we calculate some additional values, starting with resolution as height times width, logarithm of resolution, bits-per-pixel (bpp), see Equation 2, logarithm of bitrate and framerate, normalized values for framerate, see Equation 3, and resolution, see Equation 4. Here, the normalization is based on the maximum values for framerate and resolution.

Most of these additional feature values are inspired by P.1203, whereas similar calculations are performed in the mode 0 parametric model part [36], [66].

$$bpp = \frac{bitrate}{framerate \cdot resolution} \tag{2}$$

$$framerate\_norm = \frac{framerate}{60} \tag{3}$$

$$resolution\_norm = \frac{resolution}{2160 \cdot 3840} \tag{4}$$

B. TEMPORAL POOLING OF FEATURE VALUES

In our machine learning pipeline, we train several models for video quality prediction. Due to the fact that some of our features are time-dependent, e.g. having per-frame values, it is required to transform such features to time-independent values, using temporal pooling of feature values. In contrast to other models, we include more than mean values as statistics in our pooling strategy, since this enables a better reflection of the temporal change of feature values.

The approach taken is similar to the method used in [23], [27], [70]. For example, let us assume that f is such a per-frame-estimated feature vector for a given video and a single feature. In case a feature includes several values per frame, we convert it to individual vectors and perform for each of the vectors the following calculations. For f we calculate: mean value, standard deviation, skewness, kurtosis, inter-quartile range, quantiles ([0, 1] with 0.1 stepsize), and the last and first value of f. Here, the last and first values are used to frame the feature values. In addition, we split the values of f into 3 equidistant temporal groups, and for each group we calculate mean and standard deviation. With this method, for each feature we extract 25 statistical values in total that are time-independent and are later fed into our machine learning pipeline.

C. SPEEDUP AND ERROR COMPENSATION

There are several ways to speed up calculation of software in general. Besides vectorization, parallelization that better utilizes modern hardware and approximations could be used. Considering the amount of data for uncompressed 4K video, it is clear that processing will require cpu-time. For example, in case of 4:2:2-10bit 4K uncompressed video, a frame has a size of ≈ 20 MByte, with usually 60 frames played in a second. Moreover, classical pixel-based video quality models are not specifically tuned to be fast. Two possible types of sampling-based reduction can be performed, e.g. sub-sampling of frames, and per frame sub-sampling. In this paper, we consider only the reduction of per-frame information, to not interfere with temporal/motion related properties of the video. Our general idea is based on the approach presented in [21], where a center crop of the video is used to estimate video quality.

It is clear that such an approach has a stronger content dependency than the full-frame calculated model version. However, for example it was shown [21] that a center crop of 360p introduces only a rather small error compared to full-frame estimated VMAF-scores. The introduced error was below the error that occurs while repeating a same subjective test at different labs [64]. Moreover, the models instances from our framework are able to compensate some center cropped errors due to the used machine learning model, and using some more features than would be required.

D. MODEL INSTANCES

Using the introduced general model framework, that includes various features, it is possible to create several model instances. Each specific example model instance has a different application scope, which we will also highlight in the following description. Our model instances focus on pixel-based and hybrid models. For all models, as the default we use a 360p center crop. In addition, we evaluate larger crops and uncropped model variants (see Section V-D).

1) **nofu** –NO-REFERENCE

The first model instance is a no-reference model, referred to as **nofu**. It uses all **img**, **mov** and **img-nofu** features shown in Table 1. In total 64 feature values per frame are estimated. The *brisque* feature that is part of **img-nofu** is only used in this model, because here it showed an improvement in performance, while for the other models no improvement was found. All other parts of our introduced model pipeline are the same, such as the temporal pooling method. No-reference pixel-based video quality models are required in case a reference video is not accessible, and also additional meta-data cannot be extracted, for example for a given client session. Thus, the typical application for no-reference models is quality estimation for screen recordings of third-party services, or in case such a model is fast enough for real time quality monitoring [74]. Example applications include quality monitoring in case of live-streaming of broadcasting channels, or streaming of gaming sessions. We already successfully applied a reduced variant of **nofu** to estimate gaming video quality [23]. In our evaluation experiments it outperformed the unmodified VMAF model. For the considered case of gaming-video streaming prediction, we used a reduced feature set and a lightweight temporal pooling method, because gaming videos have different properties compared to the wider range of common videos.

2) **hyfu** –HYBRID NO-REFERENCE

As another model instance based on our features, a hybrid model is proposed, referred to as **hyfu**. **hyfu** uses all **img**, **mov** and bitstream **bs** features listed in Table 1. Thus, **hyfu** is an extension of **nofu** with meta-data-based bitstream features, and removing the *brisque* feature. The main application of **hyfu** is client-side video quality estimation if meta-data can be accessed, using screen recording, while the reference video is unknown. For example, in case of YouTube, Netflix and Amazon Prime Video, it is possible to estimate the required meta-data based on the DASH manifest file.

3) **fume** –FULL-REFERENCE

Especially in encoding optimization approaches, the source video is accessible, and enables the application of full-reference video quality models. We introduce a model called **fume** that is based on all **img**, **mov**, **img-fr** and **mov-fr** features described in Table 1. **fume** is a combination of pure no-reference pixel-based features with full-reference features, similar for example to the combination of full-reference features with motion features in case of Netflix's VMAF. The no-reference features are calculated for the distorted and source videos, whereas also differences of both feature values are stored as additional values. It is noted that the application scope of full-reference models is not limited to encoding optimization, since also at the production side the reference video often is available. In addition, it is also possible to use a high-quality encoded version of a given video as reference, considering that the resulting error for the final prediction is much smaller than the quality-impact introduced due to lower-bitrate encoding and processing.

4) **hyfr**–HYBRID FULL-REFERENCE

As last model instance, we developed a hybrid full-reference model called **hyfr**. It includes all features (**img**, **mov**, **img-fr**, **mov-fr** and **bs**) that are listed in Table 1. **hyfr** can be applied to monitoring or encoding optimization tasks, especially in cases where also knowledge of the underlying bitstream is accessible, in our case using meta-data. Especially to not fully focus the model on the used encoding schemes, we decided to only include some basic meta-data based features as bitstream features.

5) EXTENSIONS

We described four baseline models, that use our introduced modelling and feature approach. However, further video quality models can be developed using the features. For example, a reduced reference model could perform no-reference feature extraction on the reference video and use these features similar to **fume**, except the full-reference features, here with differences regarding these no-reference features used for estimation. Also, other prediction targets or analyses can be performed. For example, for gaming videos we already evaluated a **nofu** variant [23] or an algorithm for classification of gaming genres [26]. Accordingly, also video encoding estimation as a classification task can be performed [24]. Moreover, additional bitstream-based features could be used to enable higher modes of hybrid model variants, for example mode 3, according to ITU-T P.1203.1 [36] using QP values, or in addition using motion statistics similar to ITU-T P.1204.3 [39].

E. PREDICTION TARGETS

For developing video quality prediction models, usually a set of subjective video quality tests is performed. In such tests, a number of videos with different levels of distortion are shown, and after each video, the test user ("subject") is asked to rate the video quality based on a given rating scheme. In most cases, a single-stimulus test paradigm is used with subsequent individual videos being shown. Here, in many tests a 5-point absolute category rating (ACR) scale [35], [40] is used, where 1 means bad quality and 5 excellent, however also different other schemes are possible. In total at least 24 participants are required to yield statistically reliable quality scores from such a video quality test, according to ITU-R BT.500-13 [35]. In general, mean opinion scores (MOS) are calculated averaging the individual ratings for each stimulus v over the subjects. Those MOS values can be directly used as prediction target in our introduced video quality pipeline, in the following named as $VQ_{mos}(v) \mapsto float$ reflecting a continuous value, so that the resulting model can be conceived as regression-based. Providing predictions on a MOS-type scale in a form $VQ_{mos}(v)$ is the most common case for video quality models.

However, even other prediction targets are possible and will enable a more detailed understanding of the underlying individual ratings of participants. Preference could be another prediction target. In this case, pairwise ratings and a corresponding overall MOS score can be transformed with high correlations, depending on the video content, where some additional influences can be observed [47], [48], [91], [98].

In addition, based on majority or rounded mean or on median ratings per stimulus v, the given video quality prediction problem can be modeled as a classification task, in the following noted as $VQ_{class}(v) \mapsto int$. The VQ_{class} variant of video quality prediction is a different version of VQ_{mos} considering only discrete values. It still can be applied in cases where users' acceptance is required or a less granular quality monitoring is appropriate. For example, if a faster model with lower accuracy is used, the classification view can be a first indicator whether quality drops or other technical problems occurred in a streaming provider scenario. Moreover our classification scenario for quality just represents any kind of video classification problem using the described features within our proposed framework.

Another possibility is to model the video quality prediction task as multi-output regression problem. In such a case, for each video, a distribution of ratings based on individual subjects' scores is predicted. To this aim, the following assumptions are made her, which can be extended depending on the scope and available subjective data. In a subjective video quality test with the typical within-subject design, n participants were asked to rate the quality of the presented videos using the 5 point ACR scheme. It is noted that this approach can be extended to other rating schemes as well. Thus, it follows that for each video in the subjective test, n ratings are available. We define all ratings for a given video v as $ratings(v)$, see Equation 5.

$$ratings(v) = [rating(v, u_1), \ldots, rating(v, u_n)], \tag{5}$$

where $rating(v, u_i) \in [1..5]$ represents the categorical rating of user u_i for the video v. Using the individual ratings, a distribution can be calculated counting the frequency of each possible rating and normalizing it by n, see Equation 6.

$$prob(v) = [(r, |rating(v, u_i) = r|/n); \forall i \in 1..n \wedge r \in [1..5]] \tag{6}$$

If only a specific rating should be analyzed, the notation in Equation 7 is used.

$$prob_{=r}(v) = |rating(v, u_i) = r|/n; \;\; \forall i \in 1..n \tag{7}$$

$prob_{=r}(v)$ is the probability that a given user will rate the video v with the rating r.

Here, we focus on predicting the value of $prob_{=r}(v)$ for a given video and all possible ratings r. For example, a video v was rated by 3 participants, with the ratings $ratings(v) = [2, 5, 3]$. In addition, it can be calculated that $prob(v) = [(2, 1/3), (3, 1/3), (5, 1/3)]$, and respectively $prob_{=1}(v) = prob_{=4}(v) = 0$, $prob_{=2}(v) = prob_{=3}(v) = prob_{=5}(v) = 1/3$.

We can use these probability values as video quality prediction targets, in the following referred to as

$VQ_{prop}(v) \mapsto [prob_{=1}(v), prob_{=2}(v), prob_{=3}(v), prob_{=4}(v), prob_{=5}(v)]$. Our general idea is that for each possible rating r a separate regression algorithm is trained to predict the corresponding $prob_{=r}(v)$ values for all possible ratings $r = 1..5$, meaning the video quality prediction task is modeled as a multi-output regression problem. It is not required to always train the same type of regression algorithm, though we consider the same machine learning method for all possible ratings r.

IV. SUBJECTIVE VIDEO QUALITY DATASETS

To train the proposed and presented video quality models, we use the four subjective tests that we conducted as part of the P.NATS Phase 2 competition that resulted in the ITU-T Rec. P.1204 series of standards [37], [67]. These will be referred to as the AVT-PNATS-UHD-1 dataset in the remainder of the paper. These models are further validated and evaluated using the superset of our publicly available dataset AVT-VQDB-UHD-1 [69]. This superset comprises additional source videos employed in the tests that cannot be shared. All tests used the ACR methodology. The test session was preceded by a visual acuity test conducted for each participant using Snellen charts, as recommended in ITU-T P.910 [40] and ITU-R BT.500-13 [35]. A viewing distance of 1.5 × H was used in all tests, with H being the height of the screen. The test was conducted in a controlled lab environment following distances, lighting and other conditions according to ITU-T P.910 [40] and ITU-R BT.500-13 [35], more details are presented in [69]. The ratings were performed using the AVRateNG[10] software. The suitability of the test participants was checked by performing outlier detection. A participant was categorized as an outlier if that participant's individual ratings had a Pearson Correlation Coefficient (PCC) lower than 0.75 with the mean ratings across all participants. This method has been widely used in literature, most notably for developing ITU-T Recs. P.1203 and P.1204 [36], [37], [67]. We will briefly describe the conducted tests underlying the AVT-VQDB-UHD-1 dataset, and also provide an overview of the AVT-PNATS-UHD-1 dataset that is used to train our models instances.

A. TRAINING DATASET: AVT-PNATS-UHD-1

Four subjective tests that were designed and conducted within the P.NATS Phase 2 competition form the AVT-PNATS-UHD-1 dataset and are used to train the proposed models. Each of the four tests used more than 50 source contents of 7–9 s duration with 3 sources being common across all databases. These sources were used in combination with 5 common encoding conditions also referred to as the hypothetical reference circuits (HRCs) to form the anchor conditions across the 4 tests. The rationale behind using such a high number of sources is to have content variation across tests so that the models submitted as part of the P.NATS Phase 2 competition were capable of handling contents of different genres and complexities. The framerates of the source contents between 24 fps to 60 fps. All tests used HRCs with framerates in the range from 15 fps to 60 fps with a condition that the framerate of the encoded video was never higher than the source framerate. For each HRC, one encoding bitrate selected from the range 100 kbps to 50000 kbps and one resolution between 360p and 2160p was selected and several such HRs are used in all the tests to cover the full range of possible distortions.

Three different codecs, namely, H.264, H.265 and VP9 were used in all the 4 tests. In addition to the offline encoding of videos, segments from services such as YouTube and Bitmovin were used to include real-world encoding settings in the tests. Due to the high number of sources used in the tests, a full-factorial test design was infeasible, and hence every source was repeated only between 3 and 5 times with different HRCs. All the four tests used a 55" LG OLED screen to present the videos.

The first test in this dataset used 52 sources in combination with different HRCs, resulting in a total of 187 video stimuli or processed video sequences (PVSs) being rated by 27 participants. 2 outliers were detected using the defined criterion. In the second test, 53 different sources were used with 187 PVSs being rated by 36 participants, with 2 detected outliers. For the third test, 52 different sources were encoded with various HRCs, resulting in 185 different PVSs rated by 30 participants, with 5 outliers being detected. The fourth and final test used 53 sources with a total of 191 PVSs that were rated by 28 participants. Following the defined outlier criterion, 3 outliers were detected for this test.

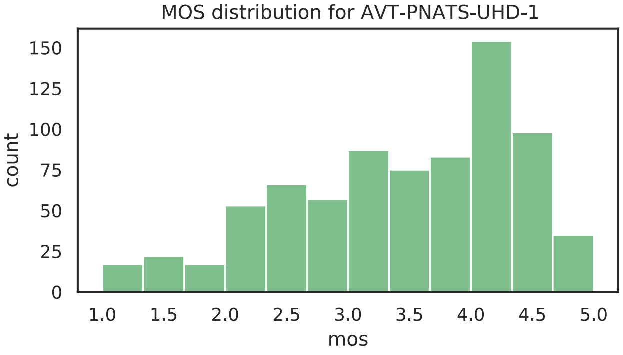


FIGURE 3. MOS distribution of all video quality tests used for training.

The quality rating distribution of all the tests is as shown in Figure 3. Here, it can be observed that mostly high-quality conditions are included within the test, e.g. the majority of ratings are between 3.5 and 5.0. Only a few conditions are rated as bad quality, e.g. with MOS values below 2.0. To further inspect the individual test subject ratings for the AVT-PNATS-UHD-1 dataset, we created boxplots for each possible rating as depicted in Figure 4. The mentioned probability refers to the VQ_{prop} problem formulation, see Section III-E. Similar to the MOS distribution it can be

[10] https://github.com/Telecommunication-Telemedia-Assessment/avrateNG

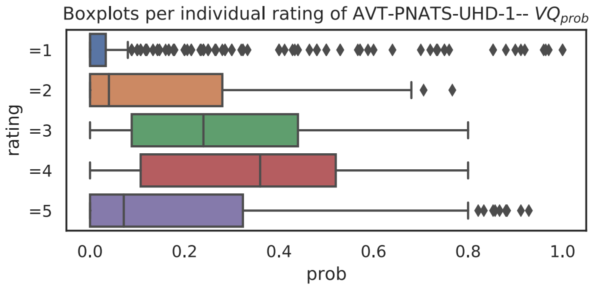


FIGURE 4. **Boxplots of individual user ratings and the corresponding distribution for training.**

concluded that high quality ratings are the majority within this dataset.

B. VALIDATION DATASET: AVT-VQDB-UHD-1

The publicly available AVT-VQDB-UHD-1 [69] dataset including the sources that could not be shared as part of the original publication is used to validate and evaluate the proposed model. This dataset consists of four different subjective tests with each test following a full-factorial test design unlike the training dataset. A total of 17 different sources of 8–10 s duration were used in the four conducted subjective tests. It is noted that in our evaluation, due to processing issues, we excluded stimuli using the 10 s water_netflix sequence (this holds only for test_1). All the sources have a framerate of 60 fps. A wide range of encoding conditions have been used in the tests, with resolutions ranging from 360*p* to 2160*p*, framerates between 15 fps and 60 fps and the encoding bitrates between 200 kbps and 40000 kbps. In the following, we will briefly present each of the four subjective tests that make up the AVT-VQDB-UHD-1 dataset. A more detailed description is presented in [69]. Like in case of the training dataset, a PCC of 0.75 was used to detect outliers. Test_1, 2 and 3 were tests with different codecs and encoding settings as in case of the training dataset AVT-PNATS-UHD-1, while test_4 was conducted to analyze the effect of different framerates on the perceived video quality.

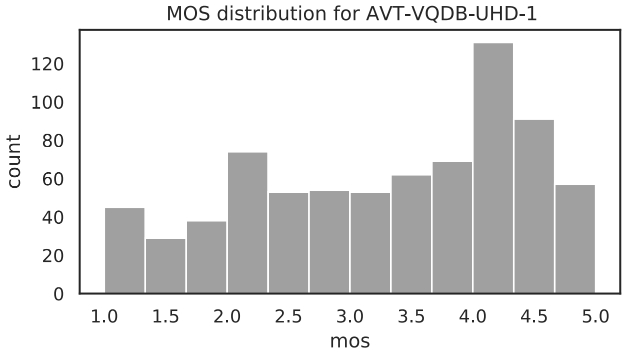


FIGURE 5. **MOS distribution of all video quality tests used for validation.**

The quality rating distribution is as shown in Figure 5 for all four tests within the AVT-VQDB-UHD-1 dataset. In contrast to the training database (AVT-PNATS-UHD-1), the distribution shows that there are more low-quality conditions included, however the majority of the stimuli are still of high quality. In Figure 6, boxplots of per-user ratings are shown for the AVT-VQDB-UHD-1 dataset. The overall dataset is more balanced considering the different rating groups.

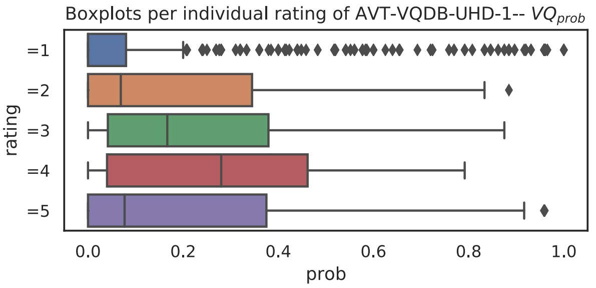


FIGURE 6. **Boxplots of individual user ratings and the corresponding distribution for the dataset used for model validation.**

1) TEST_1

In this test, the HRCs were based on varying bitrates across different resolutions. A total of six different source contents were used, each of them being encoded at four different resolutions, namely, 360*p*, 720*p*, 1080*p* and 2160*p*. The videos were encoded using two different bitrates for resolutions from 360*p* and 720*p* resolutions and three different bitrates for resolutions of 1080*p* and 2160*p*. In total all videos were encoded with three different codecs, namely, H.264, H.265 and VP9. All source videos have a framerate of 60 fps and no framerate variation was included in the test. This resulted in a total of 180 PVSs, which were rated by 29 participants. A 65'' Panasonic screen was used for video play out. There were no outliers detected for this test.

2) TEST_2

This test follows a bits-per-pixel (bpp) approach for HRC design with four different bpp values used for the four different resolutions employed in the test. As in test_1, four different resolutions, namely, 360*p*, 720*p*, 1080*p* and 2160*p* were considered and the framerate was kept constant at 60 fps, which reflects the framerate of the applied source contents. In total six different source contents were used in this test, out of which three were repeated from test_1. Owing to the higher number of HRCs and the usage of four bpp values for each resolution, only two codecs, namely, H.264 and H.265 were considered for encoding videos in this test. A total 192 PVSs were played out on a 55'' LG OLED screen for each subject. They were rated by 24 participants, with no outliers being detected.

3) TEST_3

Test_2 and test_3 together form a subset within the AVT-VQDB-UHD-1 dataset which follow a bpp approach to HRC design. Same bpp and resolutions were used as in test_2 but with H.265 and VP9 codecs to encode the video with the source contents being the same as in test_2. The H.265 encoded videos act as the anchor conditions between test_2 and test_3 thus enabling the comparison of all three codecs across the two tests. As in test_2, there were a total of 192 PVSs in this test. 26 participates took part in the test and there were no outliers. As in test_2, a 55” LG OLED screen was used to play out the videos.

4) TEST_4

Since test_4 is a test to compare the effect of different framerates on the perceived video quality, the HRC design was based on a variety of framerates, and hence only one codec, namely H.264 was used for video encoding. In total eight different source contents with no repetition from the previous tests were used in this test. The source contents were encoded in four different framerates, namely, 15 fps, 24 fps, 30 fps and 60 fps, along with six different resolutions between 360*p* and 2160*p*. This resulted in a total of 192 PVSs being rated by 25 participants. In this test, the videos were played out on the 55” OLED screen also used in test_2 and test_3. In test_4, two outliers were detected using the criterion of 0.75 PCC.

V. EVALUATION

In the following section, we will present the results of the presented four models, namely **nofu**, **hyfu**, **fume**, and **hyfr**, considering different prediction targets.

Moreover, we will perform an in-depth analysis of how the proposed center cropping approach will affect the model performance. Our training and validation does not have overlapping source videos. This enables a critical view on the performance of our models, because the model will be evaluated with unknown data.

For training we use all 764 stimuli included in the AVT-PNATS-UHD-1 dataset. Whereas the validation is based on the videos of our publicly available database AVT-VQDB-UHD-1, with a total number of 756 stimuli. The trained models are part of the open source software to enable reproducibility of our evaluation.

In the following we will evaluate the performance, for all models, first for the classification problem, then the regression problem (classical video quality evaluation), and finally the distribution prediction (multi-output regression problem). All three different prediction targets have different applications. For all models, we use 360*p* center cropping to speed up the feature extraction. A more detailed evaluation of the center crop used will also be performed in this section, even considering the computation time.

A. CLASSIFICATION PROBLEM: VQ_{class}

In contrast to the regression problem formulation, VQ_{class} uses rounded MOS values as target. Thus, this problem formulation is a classification problem and different performance metrics are required, e.g., we consider accuracy, precision, recall, f1-score (f1) and Matthews correlation coefficient (mcc) to evaluate the final classification models.

In Figures 7, normalized confusion matrices for all models considering the full validation data are shown. The best model clearly is **hyfr**, followed by **hyfu**, **fume**. The worst performing model is **nofu**, here it is visible that many cases are wrongly classified. In general, all models have in common that the quality classes with $class = 5$ and $class = 1$ are hard to predict, which is visible in the shift in the confusion matrix from the optimal diagonal line. The reason for this is that in the training dataset such ratings are rare, whereas in the validation dataset such cases occur more often.

TABLE 2. Performance values for VQ_{class} for all models; sorted by tests and mcc, rounded to 3 decimal places.

model	test	*accuracy*	*precision*	*recall*	*f1*	*mcc*
hyfr	test_1	0.660	0.661	0.660	0.595	0.514
fume	test_1	0.613	0.596	0.613	0.561	0.435
hyfu	test_1	0.580	0.597	0.580	0.519	0.367
nofu	test_1	0.513	0.421	0.513	0.430	0.242
hyfr	test_2	0.589	0.538	0.589	0.516	0.428
hyfu	test_2	0.583	0.512	0.583	0.518	0.420
fume	test_2	0.573	0.500	0.573	0.510	0.404
nofu	test_2	0.443	0.335	0.443	0.359	0.196
fume	test_3	0.599	0.546	0.599	0.543	0.420
hyfu	test_3	0.573	0.523	0.573	0.503	0.384
hyfr	test_3	0.562	0.426	0.562	0.483	0.359
nofu	test_3	0.469	0.376	0.469	0.378	0.219
hyfr	test_4	0.526	0.465	0.526	0.483	0.355
hyfu	test_4	0.484	0.419	0.484	0.407	0.285
fume	test_4	0.438	0.430	0.438	0.406	0.246
nofu	test_4	0.422	0.423	0.422	0.328	0.188
hyfr	all	0.580	0.629	0.580	0.519	0.409
fume	all	0.552	0.614	0.552	0.508	0.370
hyfu	all	0.554	0.618	0.554	0.488	0.363
nofu	all	0.459	0.497	0.459	0.377	0.206

A detailed view of performance values per subjective test that are included in the AVT-VQDB-UHD-1 dataset is presented in Table 2. The lowest performing test is test_4, here models reach a maximum *mcc* of ≈ 0.35. In contrast to test_1, with the best *mcc* of ≈ 0.52 in case of the **hyfr** model. The general problem formulation as VQ_{class} seem to be more. This can also be argued by the fact that the underlying video quality tests were targeted to cover video quality as mean opinion score and not as classification. Here a specifically designed test with a reduced number of classes (e.g. only high, medium and low quality) would lead to a better performance of the models.

B. REGRESSION PROBLEM: VQ_{mos}

As second prediction target, we introduced the quality prediction task as a regression problem VQ_{mos}.

In Figure 8, scatter plots for all four models are shown and in Table 3 a detailed view. For both the scatter plots and Table 3, a linear fit of the predicted and ground truth ratings was performed, according to ITU-T P.1401 [33]. The best model for this task is **hyfr**, followed by **hyfu** and **fume**. The performance of **nofu** is the worst, reflecting that the no-reference video quality prediction task is also the hardest. An important factor to be mentioned here is that the validation data and encoding is completely unknown to the models, and

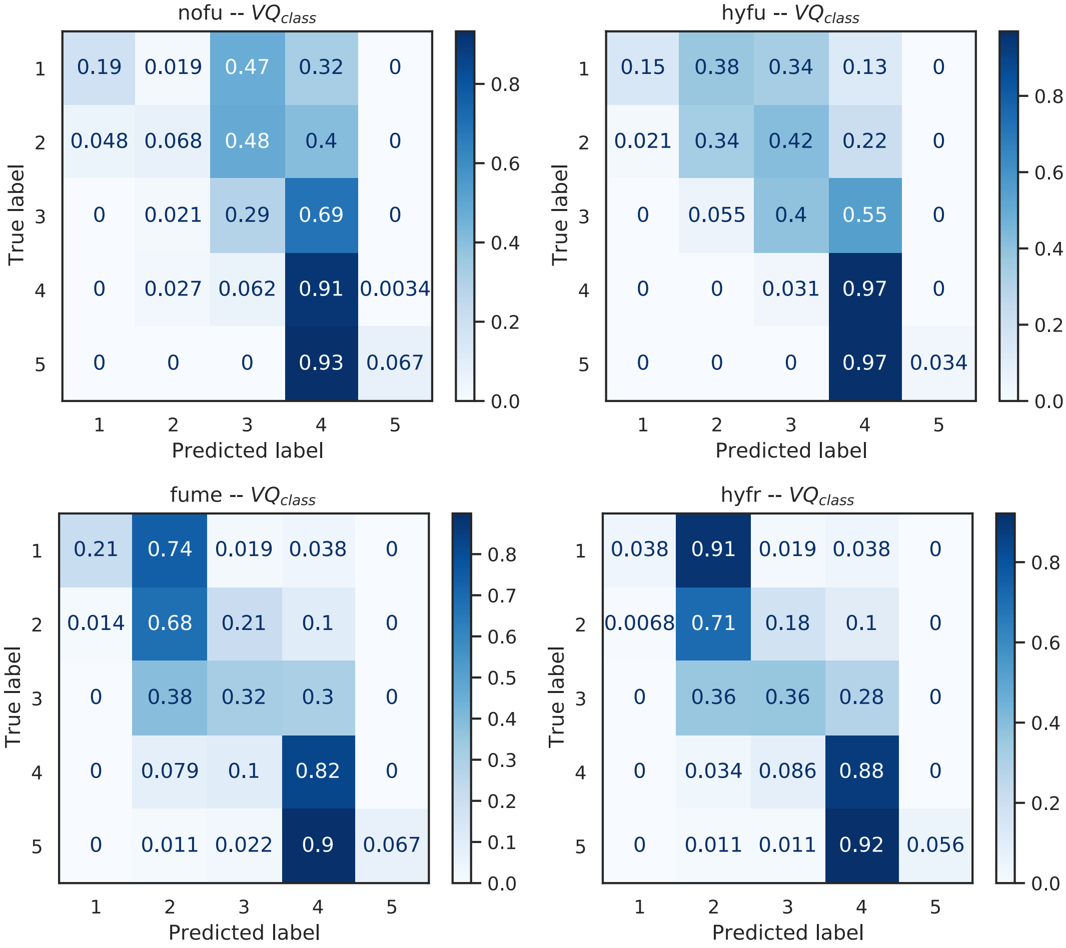


FIGURE 7. Confusion matrices for all models for VQ_{class}.

nofu will perform better if it is specifically trained on the encoding and content type that is used for prediction. Such training specific to the application scope can improve the performance of **nofu**. We already evaluated such a specialized model in case of **nofu** for gaming videos [23], where the performance of **nofu** was comparable to the performance of VMAF. Furthermore, it can be seen that the included mode 0 knowledge (bitrate, framerate, resolution) of the distorted video is a benefit for developed models, increasing the performance from e.g. ≈ 0.84 pearson correlation in case of **fume** to ≈ 0.92 in case of **hyfr**, where the only difference between these two models is the inclusion of such meta-data. Similar performance boosts can be observed for the models **hyfu** and **nofu**, even though **nofu** includes one additional no-reference feature (the inclusion of this specific feature to **hyfu** showed no performance improvement).

In addition to the evaluation of our models and because the usual video quality problem is handled as VQ_{prob}, it is possible to compare our results with different state-of-the-art models.

In Table 4, performance metrics for the AVT-VQDB-UHD-1 dataset for **VMAF**, **ADM2**, **MSSSIM**, **SSIM** and **PSNR** are shown. We only considered full-reference state-of-the-art models, because they are included in the public implementation of Netflix's VMAF and they have already been evaluated for UHD-1/4K content showing good results. Moreover, even though it is possible to re-train, for example VMAF, using our training databases, we only consider unmodified versions of the models, to enable reproducibility. Further, we used the objective model values that are included in the AVT-VQDB-UHD-1 dataset, here a similar linear fit was performed to ensure comparability. The best models for all tests included in the validation database are **VMAF** followed by **ADM2**. **VMAF** reaches a pearson correlation of ≈ 0.81 across all tests, and a maximum value of ≈ 0.94 in case of test_1. In comparison to **VMAF**, our best performing model **hyfr** has a pearson correlation of ≈ 0.92 for all tests and as best ≈ 0.94 for test_1. So **VMAF** and **hyfr** have similar performance, except that **VMAF** has a higher error in case of test_4, where more framerate variations are included, which the model was not specifically developed for. In general, test_4 seems to be the hardest for all models, and it should be mentioned that the training data does not cover a similar range of framerate

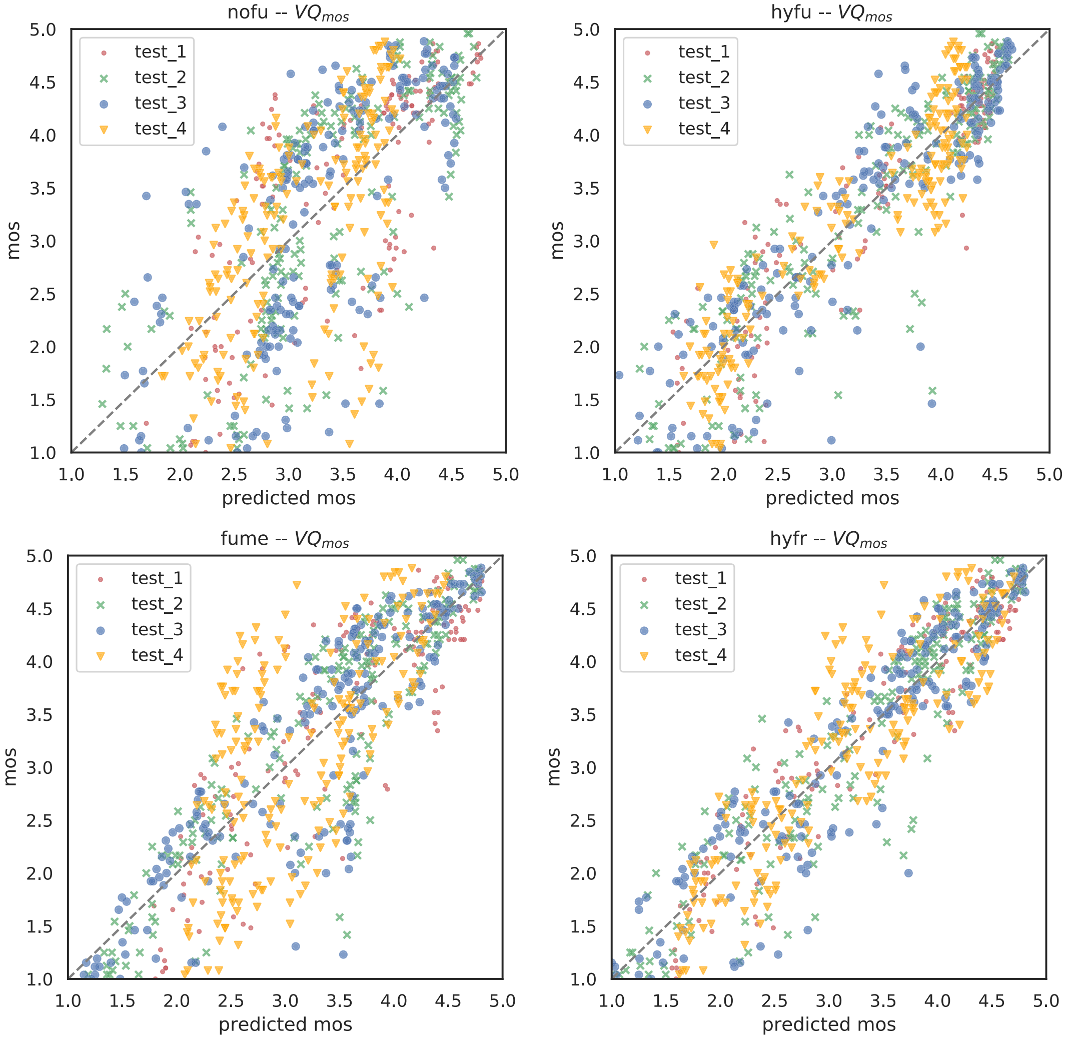


FIGURE 8. Scatter plots for all models for VQ_{mos}. For each subjective test a linear fit was performed.

variations. It can further be observed that the hybrid models predict the video quality for test_4 more precisely. However, comparing all of our models to **VMAF**, it can be stated that **hyfr**, **hyfu** and **fume** outperform **VMAF** considering all four tests. **fume** has a pearson correlation of $\approx$ 0.84 for all tests compared to **VMAF** with $\approx$ 0.81. Comparing **fume** and **VMAF** they are both full-reference models using several atom features for the overall quality estimation, however **fume** includes more temporal specific features, that cover motion related aspects, where on the contrary **VMAF** just includes a basic motion feature similar to *ti*. The model **hyfu** also outperforms **VMAF** for all tests, without having access to the source video. Our worst performing model **nofu** has a similar performance as **PSNR** all tests, and also shows better results for e.g. test_4 compared to other models. The performance of **nofu** can even be improved if larger center crops are used, as it is shown in Figure 10. However, **PSNR** is a full-reference metric compared to **nofu** that just uses the distorted video for prediction. Thus, the overall performance of **nofu** can be considered as relatively good.

C. MULTI-OUTPUT REGRESSION PROBLEM: VQ_{prob}

Besides the prediction problems formulated as classification VQ_{class} and regression problems VQ_{mos}, respectively, we further introduced the multi-output regression problem VQ_{prob}. Here, for a given video sequence, the prediction consists of several values, one for each possible rating category ($r \in [1, 2, 3, 4, 5]$). For each rating category, that one value represents the probability of users selecting that rating.

In Figure 9, for all models the prediction performance in terms of pearson correlation is shown for each possible rating r, considering all tests of the validation dataset AVT-VQDB-UHD-1. Similar to the VQ_{mos} problem, the best model is **hyfr**, followed by **hyfu**, **fume** and **nofu**. The lowest performance for prediction for all models is in case of the rating $r = 3$. Here, a possible reason may be that the training database mainly consists of high quality ratings above 3.5 in terms of MOS.

Additional performance measures are summarized in Table 5. For each rating target r, we include Pearson, Kendall and Spearman correlation values with regard to the ground

TABLE 3. Performance values for VQ_{mos} for all models; sorted by test and pearson, rounded to 3 decimal places.*all* refers to the linear fit for each database and calculating the metrics after this normalization thus is not an average of the individual test performance values.

model	test	*pearson*	*kendall*	*spearman*	*rmse*
hyfr	test_1	0.942	0.741	0.907	0.357
hyfu	test_1	0.924	0.738	0.911	0.406
fume	test_1	0.865	0.669	0.852	0.533
nofu	test_1	0.745	0.613	0.798	0.709
hyfr	test_2	0.928	0.778	0.931	0.415
hyfu	test_2	0.900	0.739	0.908	0.485
fume	test_2	0.887	0.730	0.893	0.514
nofu	test_2	0.746	0.603	0.795	0.741
hyfr	test_3	0.930	0.774	0.928	0.414
hyfu	test_3	0.900	0.725	0.894	0.489
fume	test_3	0.877	0.724	0.889	0.539
nofu	test_3	0.682	0.557	0.748	0.823
hyfu	test_4	0.916	0.735	0.912	0.403
hyfr	test_4	0.881	0.685	0.868	0.475
fume	test_4	0.660	0.485	0.652	0.754
nofu	test_4	0.600	0.472	0.632	0.803
hyfr	all	0.922	0.744	0.915	0.421
hyfu	all	0.910	0.726	0.905	0.450
fume	all	0.835	0.651	0.841	0.597
nofu	all	0.701	0.536	0.731	0.774

TABLE 4. Performance values for VQ_{mos} for state-of-the-art models; sorted by test and pearson, rounded to 3 decimal places. *all* refers to the linear fit for each database and calculating the metrics after this normalization thus is not an average of the individual test performance values.

model	test	*pearson*	*kendall*	*spearman*	*rmse*
VMAF	test_1	0.934	0.738	0.895	0.380
ADM2	test_1	0.930	0.716	0.877	0.391
SSIM	test_1	0.793	0.595	0.762	0.658
MSSSIM	test_1	0.772	0.566	0.726	0.677
PSNR	test_1	0.745	0.544	0.706	0.708
VMAF	test_2	0.923	0.782	0.930	0.429
ADM2	test_2	0.919	0.768	0.922	0.440
PSNR	test_2	0.805	0.638	0.813	0.663
MSSSIM	test_2	0.769	0.630	0.815	0.714
SSIM	test_2	0.753	0.637	0.823	0.742
VMAF	test_3	0.910	0.745	0.909	0.466
ADM2	test_3	0.904	0.739	0.908	0.483
PSNR	test_3	0.780	0.625	0.793	0.706
MSSSIM	test_3	0.734	0.597	0.783	0.765
SSIM	test_3	0.713	0.578	0.765	0.793
ADM2	test_4	0.799	0.615	0.806	0.603
VMAF	test_4	0.789	0.624	0.811	0.617
MSSSIM	test_4	0.558	0.421	0.581	0.833
PSNR	test_4	0.509	0.353	0.494	0.864
SSIM	test_4	0.494	0.418	0.580	0.873
VMAF	all	0.816	0.625	0.817	0.627
ADM2	all	0.792	0.586	0.786	0.663
MSSSIM	all	0.785	0.584	0.782	0.672
SSIM	all	0.765	0.559	0.759	0.699
PSNR	all	0.731	0.538	0.723	0.741

truth data. The best prediction is clearly the case where $r = 5$. This is due to the mainly high quality ratings that are part of the training and validation datasets. Further, for such high-quality cases with $MOS \approx 5$, almost all subjects must have rated $r = 5$, to achieve such a high mean rating. As can be seen from Figure 9, the values for Kendall and Spearman correlation behave similarly as the Pearson Correlation does, thus the worst performing prediction target is $r = 3$. Here, it should be mentioned that the used multi-output regression approach trains separate models for each rating $r \in [1, 2, 3, 4, 5]$, for this reason there is no connection between the individual prediction targets given. A different machine learning pipeline or algorithm that takes into account such hidden connections could improve the prediction performance.

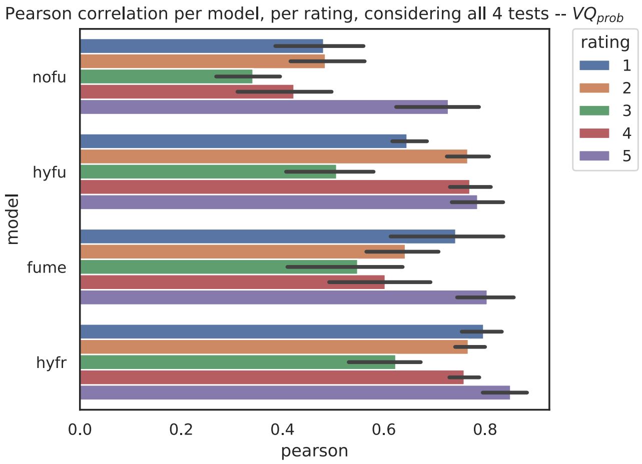


FIGURE 9. Performance across all tests in case of VQ_{prob} considering all four models, with 95% confidence intervals.

TABLE 5. Mean performance values for VQ_{prob} for all tests; sorted by rating and Pearson, rounded to 3 decimal places.

model	rating r	*pearson*	*kendall*	*spearman*
nofu	1	0.486	0.373	0.492
hyfu	1	0.638	0.547	0.700
fume	1	0.724	0.542	0.681
hyfr	1	0.784	0.600	0.747
nofu	2	0.487	0.396	0.555
fume	2	0.638	0.516	0.704
hyfu	2	0.749	0.561	0.751
hyfr	2	0.757	0.605	0.798
nofu	3	0.325	0.239	0.343
hyfu	3	0.496	0.353	0.501
fume	3	0.545	0.408	0.569
hyfr	3	0.622	0.471	0.645
nofu	4	0.437	0.290	0.436
fume	4	0.592	0.410	0.580
hyfr	4	0.748	0.522	0.715
hyfu	4	0.761	0.514	0.706
nofu	5	0.693	0.512	0.678
hyfu	5	0.770	0.660	0.843
fume	5	0.811	0.619	0.795
hyfr	5	0.855	0.711	0.883

D. CENTER CROP EVALUATION

As mentioned in Section III-C and III-D our model instances use a center cropped version of the input videos to calculate features. This approach is similar to the **cencro** approach proposed in [21]. However, in that previous work, several full-reference models were applied on full-frames, and an additional evaluation using center cropped frames was performed. In the present paper, we want to further evaluate the proposed center cropping approach and its impact on the performance and feature calculation speed. In total, we selected five different center cropping settings namely 240*p*, 360*p*, 720*p*, 1080*p* and 2160*p*, where the last setting refers to the full-frame, thus no center cropping being used. For each of the center cropping settings, we trained all four models with the training dataset described in Section IV. In Figure 10, the performance values for all models and cropping settings are shown, considering 10-fold-cross validation of the employed training data. We performed 32 training repetitions. In this part, we only focus on the evaluation of the VQ_{mos} problem formulation. Similar results can be observed with the other variants and also using the validation dataset.

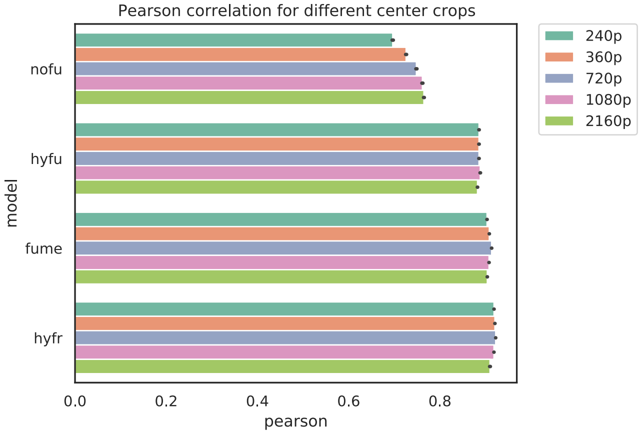


FIGURE 10. Prediction performance evaluation of different center cropping values, based on 32 training runs for each model and each center cropping value.

First, it is notable that there is only a small improvement for the models **hyfu**, **fume** and **hyfr** in case of different center crop values. In contrast to **nofu**, here the performance can be slightly improved using a larger center crop. A 360*p* center crop for **nofu** results in a pearson correlation value of around 0.73, whereas the center crop setting of 720*p* improves it to 0.75, 1080*p* ≈ 0.76 and 2160*p* results in 0.76. The worst performance of around 0.70 is in case of a 240*p* center cropping setting. All the other models have nearly the same rounded performance considering the introduced center crop variations. However, to have a uniform structure of all models we decided to also use a 360*p* center crop for **nofu**, even if the performance is slightly lower than a for 720*p* center crop, pearson correlation of 0.75 vs. 0.73.

The processing time is an important factor in addition to the overall performance of all models considering the used center cropping parameter. For this reason, we measured the overall model prediction time, including the conversion of the distorted video to the center cropped variant, the time required for feature extraction and model prediction time. Especially the feature extraction time is the major part of the overall processing time for our introduced model instances.

We selected one video sequence (american footbal, 360*p* resolution target encoding resolution, bitrate = 200 kbit/s, video codec vp9) as test sequence, and measured the overall processing time of all center crop variants for 32 repetitions, where each run removes all cached files of the previously performed run. Different videos will end up with slightly different processing time that is required because the features are content dependent. However, the overall connection of different center crops will be similar, as it has been already shown in [21]. Here, it should be mentioned that all of our steps are single core optimized (except the conversion of the distorted video, here several cores are used). The introduced and published framework allows for parallel processing considering different videos in a data parallelization manner. All measurements were performed on the same computer, with a Intel Core i7-9700 CPU (3.00 GHz) with 64 GB of main memory and local file access (SSD).

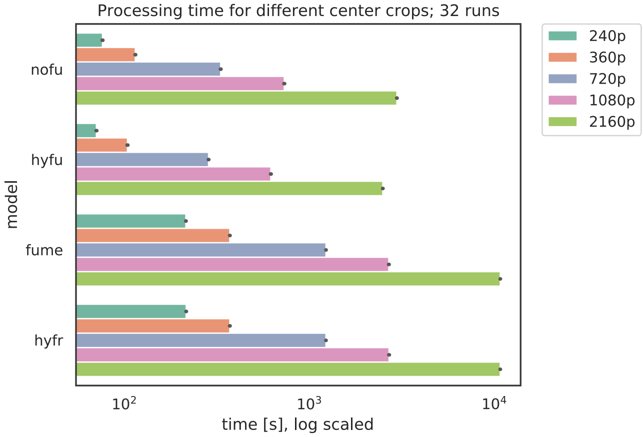


FIGURE 11. Overall processing time for quality prediction considering different center cropping values. Shown are mean values and 95% confidence intervals across 32 repetitions each.

In Figure 11 mean values with 95% confidence intervals for each center cropping parameter and each model are shown respectively. The fastest two models are clearly the no-reference models (**nofu** and **hyfr**), with the hybrid model being slightly faster, due to the fact that it does not include the **img-nofu** feature. In addition, it clearly can be seen that there is an exponential relationship between processing time and used center crop setting, compare also Table 6. For example, the **hyfu** model requires about 70 s for 240*p* and ≈ 2466 s for 2160*p*, thus 9 times the center cropping height results in about ≈ 35 times the processing time. The other models behave similarly across several center crop values. In general the full-reference models need about four 3-4 times the processing time, e.g. for **hyfr** in case of 720*p* it takes around 1216 s, compared to ≈ 282 s for **hyfu**.

Considering the speedup that we can achieve using a center crop and the negligible performance reduction for most

TABLE 6. Mean processing time [s] for each model for different center crop settings; values are rounded to integers.

center crop	nofu	hyfu	fume	hyfr
240p	75	70	213	214
360p	114	103	368	368
720p	328	282	1216	1216
1080p	724	613	2657	2665
2160p	2938	2466	10633	10626

of our models (except **nofu**), we selected a center crop of 360p as the best trade-off between speed and prediction performance. This conclusion is along with our results of other full-reference models [21], where a 360p center crop was able to speedup calculation time significantly, while still preserving high prediction accuracy of the models.

VI. DISCUSSION

We introduced our proposed framework for video quality prediction and furthermore instantiated four different models for three prediction targets.

The first prediction target handles video quality as a classification task VQ_{class}. Here it is notable, that especially for this formulation of the quality prediction problem seems to be hard for our models. A main reason for this is that for such a formulation a more uniformly distributed training dataset is required. A more suitable training dataset should also target classification for video quality, e.g. including only three main classes, low, medium and high quality. From the analysis of the used databases it can be seen that the lowest and highest quality classes are not well predicted and also not represented frequently enough in the training dataset.

Furthermore, our model **nofu** has a low performance compared to the other model variants. An example reason for this is the diversity of the underlying video content, and it was reported that a more constrained **nofu**-based model variant already shows better performance for gaming content [23]. Here the general challenge of pixel-based no-reference video quality estimation is still an open and hard task, especially when unknown video content is considered. As second prediction target, we focus on the commonly used problem formulation, namely video quality as a single continuous score VQ_{mos}, thus our approach considers it as a regression problem. Here, we show that three of our models (**fume**, **hyfu** and **hyfr**) are able to outperform state-of-the-art models, e.g. Netflix's VMAF, considering the used evaluation metrics. Even though the model **nofu** shows a lower overall performance compared to VMAF, it still shows a comparable performance to PSNR and SSIM, which are also commonly used video quality models. The evaluation shows in addition, that the defined features are capable for the prediction tasks.

As last prediction target, we handle the video quality task as a multi-output regression problem VQ_{prob}, where several models are trained and predict a distribution of ratings. All models show similar performance compared to the VQ_{mos} formulation. However, the prediction of individual ratings r could benefit of knowledge of the other ratings, thus further analysis is required.

In addition to the three different video quality prediction variants, we evaluate the used center cropping approach, enabling us to speed up our feature calculation significantly, with only a minor increase in prediction error in comparison to the ground truth subjective scores. It is shown that the introduced error is comparable to the error that would occur when a subjective video quality test is repeated in a different lab, according to [21], [64]. Only the model **nofu** could benefit from a larger used center crop, however we decided to even use for this model a 360p center crop to have a unified model architecture. Beside the model performance, we also evaluated the required processing time, and it can be seen that there is a huge cpu-time saving when center-cropping is used, this confirms and extends our observations in [21].

VII. CONCLUSION AND FUTURE WORK

We started with the observation that there are only a few video quality models available and specifically trained for UHD-1/4K video contents. Moreover generally there is a wide range of features and subsequent integration approaches described in the literature, without these being available in a collection of tools suitable for developing own models. To overcome these limitations, the paper introduces a general video quality modelling pipeline, which is made available as open source. Our model pipeline includes a set of features that are image- or motion-based, and a temporal feature pooling method. This allows for the evaluation of several machine learning algorithms for the generic task of video quality prediction. Besides the traditional modeling of video quality using mean opinion scores in a regression scenario, we described two further approaches, namely a classification and a multi-output regression variant. Both new variants can be used to further extend the application of video quality models, for example considering different applications such as prediction of uncertainties in user's ratings or other video classification applications beyond quality prediction.

Based on the model architecture, we instantiate four different video quality models that are publicly available. Two out of the four models are pure pixel-based models (a no-reference and a full-reference model – **fume** and **nofu**). In addition, for each of these we describe a hybrid model extension, **hyfu** and **hyfr**, incorporating additional video metadata about the codec used, resolution, bitrate and framerate. Such meta-data is typical accessible during play out of a given video, while other bitstream related data requires specifically designed extractors.

To properly train and validate the models, we describe a set of subjective quality tests conducted by our group that we used for training and validation, where the validation database is publicly available. As we further publish the code of our models and their trained instances, it ensures that our validation experiments are reproducible. In our evaluation, we show that our models have a similar or even better performance than state-of-the-art models, whereas the hybrid models outperform the non-hybrid models. Moreover, we evaluated three different prediction targets for the underlying video quality estimation problem. For each of the problem formulations we evaluated four model instances, whereas the hybrid (**hyfr** and **hyfu**) and full-reference model (**fume**) show the best results. Furthermore, we evaluated the introduced center cropping approach regarding the prediction error, it is shown that there is only a small negligible error introduced, for this reason we used a 360*p* center crop for all instantiated models.

Our introduced pipeline can even be used for different video analyses, as we already showed in several of our previously conducted work, e.g. video classification [27], genre classification for games [26], estimation of encoding parameters [24] or using the center cropping approach for 360° video quality [20]. Promising extensions of our models could include further knowledge of the bitstream itself, similar to the P.1204.3 model, where e.g. QP values and motion statistics are extracted from the bitstream [67], [70]. In addition, the video quality problem formulations as classification and multi-output regression tasks need to be further investigated, e.g. including specifically designed video quality tests.

ACKNOWLEDGMENT

The authors would like to thank the Deutsche Telekom AG Team, namely Peter List and Ulf Wüstenhagen, for their inputs over the course of the work.

REFERENCES

[1] Aomedia. (2020). *Av1 Overview*. Accessed: Nov. 30, 2020. [Online]. Available: https://aomedia.org/av1-features/

[2] T. O. Aydin, A. Smolic, and M. Gross, "Automated aesthetic analysis of photographic images," *IEEE Trans. Vis. Comput. Graphics*, vol. 21, no. 1, pp. 31–42, Jan. 2015.

[3] C. G. Bampis, P. Gupta, R. Soundararajan, and A. C. Bovik, "SpEED-QA: Spatial efficient entropic differencing for image and video quality," *IEEE Signal Process. Lett.*, vol. 24, no. 9, pp. 1333–1337, Sep. 2017.

[4] C. G. Bampis, Z. Li, I. Katsavounidis, and A. C. Bovik, "Recurrent and dynamic models for predicting streaming video quality of experience," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3316–3331, Jul. 2018.

[5] M. Barkowsky, I. Sedano, K. Brunnström, M. Leszczuk, and N. Staelens, "Hybrid video quality prediction: Reviewing video quality measurement for widening application scope," *Multimedia Tools Appl.*, vol. 74, no. 2, pp. 323–343, Jan. 2015.

[6] 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.

[7] 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, 2019.

[8] 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 *Proc. 23rd Packet Video Workshop*, Jun. 2018, pp. 7–12.

[9] N. Barman, S. Zadtootaghaj, S. Schmidt, M. G. Martini, and S. Moller, "GamingVideoSET: A dataset for gaming video streaming applications," in *Proc. 16th Annu. Workshop Netw. Syst. Support Games (NetGames)*, Jun. 2018, pp. 1–6.

[10] K. Berger, Y. Koudota, M. Barkowsky, and P. Le Callet, "Subjective quality assessment comparing UHD and HD resolution in HEVC transmission chains," in *Proc. 7th Int. Workshop Qual. Multimedia Exper. (QoMEX)*, May 2015, pp. 1–6.

[11] S. Bosse, D. Maniry, K.-R. Muller, T. Wiegand, and W. Samek, "Deep neural networks for no-reference and full-reference image quality assessment," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 206–219, Jan. 2018.

[12] S. Bosse, D. Maniry, K.-R. Muller, T. Wiegand, and W. Samek, "Neural network-based full-reference image quality assessment," in *Proc. Picture Coding Symp. (PCS)*, 2016, pp. 1–5.

[13] *Cisco Visual Networking index: Forecast and Trends, 2017–2022*. Accessed: Feb. 17, 2021. [Online]. Available: https://davidellis.ca/wp-content/uploads/2019/12/cisco-vni-mobile-data-traffic-feb-2019.pdf

[14] Cisco. *Cisco Visual Networking Index: Forecast and Methodology, 2015–2020*. Accessed: Feb. 17, 2021. [Online]. Available: http://www.audentia-gestion.fr/cisco/white-paper-c11-738085.pdf

[15] P. P. Dash, A. Wong, and A. Mishra, "VeNICE: A very deep neural network approach to no-reference image assessment," in *Proc. IEEE Int. Conf. Ind. Technol. (ICIT)*, Mar. 2017, pp. 1091–1096.

[16] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, Sep. 1994.

[17] K. Egiazarian, J. Astola, N. Ponomarenko, V. Lukin, F. Battisti, and M. Carli, "New full-reference quality metrics based on HVS," in *Proc. 2nd Int. Workshop Video Process. Qual. Metrics*, vol. 4, 2006, pp. 1–4.

[18] A. E. Essaili, T. Lohmar, and M. Ibrahim, "Realization and evaluation of an end-to-end low latency live DASH system," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2018, pp. 1–5.

[19] O. El Marai, T. Taleb, M. Menacer, and M. Koudil, "On improving video streaming efficiency, fairness, stability, and convergence time through client–server cooperation," *IEEE Trans. Broadcast.*, vol. 64, no. 1, pp. 11–25, Mar. 2018.

[20] S. Fremerey, S. Göring, R. Rao, R. Huang, and A. Raake, "Subjective test dataset and meta-data-based models for 360° streaming video quality," in *Proc. IEEE 22nd Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2020, pp. 1–6.

[21] S. Goring, C. Krammer, and A. Raake, "Cencro–Speedup of video quality calculation using center cropping," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2019, pp. 1–8.

[22] S. Goring and A. Raake, "Deimeq–A deep neural network based hybrid no-reference image quality model," in *Proc. 7th Eur. Workshop Vis. Inf. Process. (EUVIP)*, Nov. 2018, pp. 1–6.

[23] S. Göring, R. Rao, and A. Raake, "nofu—A lightweight no-reference pixel based video quality model for gaming content," in *Proc. 11th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, Berlin, Germany, Jun. 2019, pp. 1–9.

[24] S. Göring, R. Rao, and A. Raake, "Prenc—Predict number of video encoding passes with machine learning," in *Proc. 12th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, Athlone, Ireland, May 2020, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9123108

[25] S. Göring, J. Skowronek, and A. Raake, "DeViQ–A deep no reference video quality model," *Electron. Imag., Hum. Vis. Electron. Imag.*, vol. 2018, no. 14, pp. 1–6, 2018. [Online]. Available: https://www.ingentaconnect.com/content/ist/ei/2018/00002018/00000014/art00017

[26] S. Goring, R. Steger, R. Rao Ramachandra Rao, and A. Raake, "Automated genre classification for gaming videos," in *Proc. IEEE 22nd Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2020, pp. 1–6.

[27] S. Göring, J. Zebelein, S. Wedel, D. Keller, and A. Raake, "Analyze and predict the perceptibility of UHD video contents," *Electron. Imag.*, vol. 2019, no. 12, p. 215, 2019. [Online]. Available: https://www.ingentaconnect.com/content/ist/ei/2019/00002019/00000012/art00009

[28] D. Hasler and S. E. Suesstrunk, "Measuring colorfulness in natural images," *Proc. SPIE*, vol. 5007, pp. 87–96, Jun. 2003.

[29] C. R. Helmrich, M. Siekmann, S. Becker, S. Bosse, D. Marpe, and T. Wiegand, "Xpsnr: A low-complexity extension of the perceptually weighted peak signal-to-noise ratio for high-resolution video quality assessment," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 2727–2731.

[30] *Comparison of Quality Assessment of UHD and HD Videos at Two Different Viewing Distances*, NTT, Tokyo, Japan, 2017. [Online]. Available: https://www.itu.int/md/T17-SG12-C-0005/_page.print

[31] *HDTV and UHDTV Including HDR-TV Test Materials for Assessment of Picture Quality*, International Telecommunication Union, Geneva, Switzerland, document ITU-R BT.2245-3, 2017.

[32] *The Present State of Ultra-High Definition Television*, International Telecommunication Union, Geneva, Switzerland, document ITU-R BT.2246-6, 2017.

[33] *Methods, Metrics and Procedures for Statistical Evaluation, Qualification and Comparison of Objective Quality Prediction Models*, Int. Telecommunication Union, document ITU-T-P.1401, 2014.

[34] *Recommendation H.266 (08/20)–Versatile Video Coding*, ITU Rec., ITU-T, 2020.

[35] *Recommendation ITU-R BT.500-13–Methodology for the Subjective Assessment of the Quality of Television Pictures*, ITU Rec., 2014.

[36] *Recommendation P.1203–Parametric Bitstream-Based Quality Assessment of Progressive Download and Adaptive Audiovisual Streaming Services Over Reliable Transport*, ITU Rec., 2016.

[37] *Recommendation P.1204–Video Quality Assessment of Streaming Services Over Reliable Transport for Resolutions up to 4K*, ITU Rec., 2019.

[38] *Recommendation P.1204.4: Video Quality Assessment of Streaming Services Over Reliable Transport for Resolutions Up to 4K With Access to Full and Reduced Reference Pixel Information*, ITU Rec., ITU-T, 2019.

[39] *Recommendation P.1204.3: Video Quality Assessment of Streaming Services Over Reliable Transport for Resolutions up to 4K With Access to Full Bitstream Information*, ITU Rec., 2019.

[40] *Subjective Video Quality Assessment Methods for Multimedia Applications*, document ITU-T.P.910, 2008.

[41] *Subjective Video Quality Assessment Methods for Multimedia Applications. Serie P: Telephone Transmission Quality, Telephone Installations, Local Line Networks*, International Telecommunication Union. Geneva, document ITU-T P.910, 2008.

[42] P. A. Kara, W. Robitza, A. Raake, and M. G. Martini, "The label knows better: The impact of labeling effects on perceived quality of HD and UHD video streaming," in *Proc. 9th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, May 2017, pp. 1–6.

[43] I. Katsavounidis. (2018). Dynamic Optimizer—A Perceptual Video Encoding Optimization Framework. The Netflix Tech Blog. Accessed: Feb. 17, 2021. [Online]. Available: https://netflixtechblog.com/dynamic-optimizer-a-perceptual-video-encoding-optimization-framework-e19f1e3a277f

[44] I. Katsavounidis, A. Aaron, and D. Ronca, "Native resolution detection of video sequences," in *Proc. SMPTE Annu. Tech. Conf. Exhib.*, Oct. 2015, pp. 1–20.

[45] S. Kuanar, C. Conly, and K. R. Rao, "Deep learning based HEVC in-loop filtering for decoder quality enhancement," in *Proc. Picture Coding Symp. (PCS)*, Jun. 2018, pp. 164–168.

[46] S. Kuanar, K. R. Rao, M. Bilas, and J. Bredow, "Adaptive CU mode selection in HEVC intra prediction: A deep learning approach," *Circuits, Syst., Signal Process.*, vol. 38, no. 11, pp. 5081–5102, Nov. 2019.

[47] P. Lebreton, A. Raake, M. Barkowsky, and P. Le Callet, "Perceptual preference of S3D over 2D for HDTV in dependence of video quality and depth," in *Proc. IVMSP*, Jun. 2013, pp. 1–4.

[48] J. Li, M. Barkowsky, and P. Le Callet, "Analysis and improvement of a paired comparison method in the application of 3DTV subjective experiment," in *Proc. 19th IEEE Int. Conf. Image Process.*, Sep. 2012, pp. 629–632.

[49] S. Li, F. Zhang, L. Ma, and K. N. Ngan, "Image quality assessment by separately evaluating detail losses and additive impairments," *IEEE Trans. Multimedia*, vol. 13, no. 5, pp. 935–949, Oct. 2011.

[50] J. Y. Lin, T.-J. Liu, E. C.-H. Wu, and C.-C.-J. Kuo, "A fusion-based video quality assessment (FVQA) index," in *Proc. Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA), Asia–Pacific*, Dec. 2014, pp. 1–5.

[51] D. Liu, Y. Li, J. Lin, H. Li, and F. Wu, "Deep learning-based video coding: A review and a case study," *ACM Comput. Surv.*, vol. 53, no. 1, pp. 1–35, 2020.

[52] H. Men, H. Lin, and D. Saupe, "Spatiotemporal feature combination model for no-reference video quality assessment," in *Proc. 10th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, May 2018, pp. 1–3.

[53] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Trans. Image Process.*, vol. 21, no. 12, pp. 4695–4708, Dec. 2012.

[54] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a 'completely blind' image quality analyzer," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 209–212, Mar. 2013.

[55] A. K. Moorthy and A. C. Bovik, "A two-stage framework for blind image quality assessment," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 2481–2484.

[56] Netflix. (2018). *4K Support*. Accessed: Jul. 9, 2018. [Online]. Available: https://help.netflix.com/en/node/13444

[57] Netflix. *Netflix Vmaf*. Accessed: Feb. 17, 2021. [Online]. Available: https://github.com/Netflix/vmaf

[58] Netflix. (2018). *Vmaf 4K Included*. Accessed: Jul. 9, 2018. [Online]. Available: https://github.com/Netflix/vmaf

[59] M. Orduna, C. Díaz, L. Muñoz, P. Pérez, I. Benito, and N. García, "Video multimethod assessment fusion (VMAF) on 360 VR contents," 2019, *arXiv:1901.06279*. [Online]. Available: http://arxiv.org/abs/1901.06279

[60] K. Otsuji and Y. Tonomura, "Projection-detecting filter for video cut detection," *Multimedia Syst.*, vol. 1, no. 5, pp. 205–210, Mar. 1994.

[61] R. Pantos. (2011). *HTTP Live Streaming*. [Online]. Available: https://tools.ietf.org/html/draft-pantos-http-live-streaming-13

[62] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.

[63] C. Perra, "A low computational complexity blockiness estimation based on spatial analysis," in *Proc. 22nd Telecommun. Forum Telfor (TELFOR)*, Nov. 2014, pp. 1130–1133.

[64] M. H. Pinson and S. Wolf, "Comparing subjective video quality testing methodologies," *Proc. SPIE*, vol. 5150, pp. 573–582, Jun. 2003.

[65] M. T. Qadri, K. T. Tan, and M. Ghanbari, "Frequency domain blockiness measurement for image quality assessment," in *Proc. 2nd Int. Conf. Comput. Technol. Develop.*, Nov. 2010, pp. 282–285.

[66] A. Raake, M.-N. Garcia, W. Robitza, P. List, S. Göring, and B. Feiten, "A bitstream-based, scalable video-quality model for HTTP adaptive streaming: ITU-T P.1203.1," in *Proc. 9th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, Erfurt, May/Jun. 2017, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/document/7965631/

[67] A. Raake, S. Borer, S. Satti, J. Gustafsson, R. Rao, S. Medagli, P. List, S. Göring, D. Lindero, W. Robitza, G. Heikkilä, S. Broom, C. Schmidmer, B. Feiten, U. Wüstenhagen, T. Wittmann, M. Obermann, and R. Bitto, "Multi-model standard for bitstream-, pixel-based and hybrid video quality assessment of UHD/4K: ITU-T P.1204," *IEEE Access*, vol. 8, pp. 193020–193049, 2020. [Online]. Available: https://ieeexplore.ieee.org/document/9234526?source=authoralert

[68] R. Rao, S. Göring, N. P. Patrick Vogel, J. J. V. Villarreal, W. Robitza, P. List, B. Feiten, and A. Raake, "Adaptive video streaming with current codecs and formats: Extensions to parametric video quality model ITU-T P. 1203," *Electron. Imag.*, vol. 2019, no. 10, p. 314, 2019. [Online]. Available: https://www.ingentaconnect.com/content/ist/ei/2019/00002019/00000010/art00015

[69] R. R. Ramachandra Rao, S. Goring, W. Robitza, B. Feiten, and A. Raake, "AVT-VQDB-UHD-1: A large scale video quality database for UHD-1," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2019, pp. 1–8. [Online]. Available: https://ieeexplore.ieee.org/document/8959059

[70] R. Rao, S. Göring, W. Robitza, A. Raake, B. Feiten, P. List, and U. Wüstenhagen, "Bitstream-based model standard for 4K/UHD: ITU-T P.1204.3—Model details, evaluation, analysis and open source implementation," in *Proc. 12th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, Athlone, Ireland, May 2020, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/9123110

[71] R. Rao, S. Göring, R. Steger, S. Zadtootaghaj, N. Barman, S. Fremerey, S. Müller, and A. Raake, "A large-scale evaluation of the bitstream-based video-quality model ITU-T P.1204.3 on gaming content," in *Proc. IEEE 22nd Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2020, pp. 1–6.

[72] *Video Quality Assessment of Streaming Services Over Reliable Transport for Resolutions up to 4K With Access to Transport and Received Pixel Information*, International Telecommunication Union, document P.1204.5, 2019.

[73] S. Rimac-Drlje, D. Zagar, and G. Martinovic, "Spatial masking and perceived video quality in multimedia applications," in *Proc. 16th Int. Conf. Syst., Signals Image Process.*, Jun. 2009, pp. 1–4.

[74] W. Robitza, A. Ahmad, P. A. Kara, L. Atzori, M. G. Martini, A. Raake, and L. Sun, "Challenges of future multimedia QoE monitoring for Internet service providers," *Multimedia Tools Appl.*, vol. 76, no. 21, pp. 22243–22266, Nov. 2017, doi: 10.1007/s11042-017-4870-z.

[75] W. Robitza, S. Göring, A. Raake, D. Lindegren, G. Heikkilä, J. Gustafsson, P. List, B. Feiten, U. Wüstenhagen, M.-N. Garcia, K. Yamagishi, and S. Broom, "HTTP adaptive streaming QoE estimation with ITU-T rec. P. 1203: Open databases and software," in *Proc. 9th ACM Multimedia Syst. Conf.*, Amsterdam, The Netherlands, 2018, pp. 466–471.

[76] Samsung. *Future of Display*. [Online; 07.09.2018]. Accessed: Sep. 7, 2018. [Online]. Available: https://news.samsung.com/global/ifa-docent-series-part-1-tv-as-the-lifestyle-screen-the-future-of-display

[77] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[78] M. Shahid, A. Rossholm, B. Lövström, and H.-J. Zepernick, "No-reference image and video quality assessment: A classification and review of recent approaches," *EURASIP J. Image Video Process.*, vol. 2014, no. 1, p. 40, Aug. 2014, doi: 10.1186/1687-5281-2014-40.

[79] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430–444, Feb. 2006.

[80] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3440–3451, Nov. 2006.

[81] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556

[82] R. Sotelo, J. Joskowicz, M. Anedda, M. Murroni, and D. D. Giusto, "Subjective video quality assessments for 4K UHDTV," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2017, pp. 1–6.

[83] R. Soundararajan and A. C. Bovik, "Video quality assessment by reduced reference spatio-temporal entropic differencing," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 4, pp. 684–694, Apr. 2013.

[84] K. Spiteri, R. Sitaraman, and D. Sparacio, "From theory to practice: Improving bitrate adaptation in the DASH reference player," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 15, no. 2, p. 67, Jul. 2019.

[85] 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 Appl.*, pp. 1–23, 2020, doi: 10.1007/s11042-020-09144-6.

[86] G. Van Wallendael, P. Coppens, T. Paridaens, N. Van Kets, W. Van den Broeck, and P. Lambert, "Perceptual quality of 4K-resolution video content compared to HD," in *Proc. 8th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, Jun. 2016, pp. 1–6.

[87] Y. Wang, "Survey of objective video quality measurements," EMC Corp., Hopkinton, MA, USA, 2006, vol. 1748, p. 39.

[88] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[89] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. 37th Asilomar Conf. Signals, Syst. Comput.*, vol. 2, 2004, pp. 1398–1402.

[90] O. Wiedemann, V. Hosu, H. Lin, and D. Saupe, "Disregarding the big picture: Towards local image quality assessment," in *Proc. 10th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, May 2018, pp. 1–6.

[91] W. Hagen, C. Hold, and A. Raake, "Listener preference for wave field synthesis, stereophony, and different mixes in popular music," *J. Audio Eng. Soc.*, vol. 66, no. 5, pp. 385–396, May 2018.

[92] S. Winkler and P. Mohandas, "The evolution of video quality measurement: From PSNR to hybrid metrics," *IEEE Trans. Broadcast.*, vol. 54, no. 3, pp. 660–668, Sep. 2008.

[93] Y. Yalman and I. Ertürk, "A new color image quality measure based on YUV transformation and PSNR for human vision system," *Turkish J. Elect. Eng. Comput. Sci.*, vol. 21, no. 2, pp. 603–612, 2013.

[94] K. Yamagishi, T. Kawano, and T. Hayashi, "Hybrid Video-Quality-Estimation model for IPTV services," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Nov. 2009, pp. 1–5.

[95] R. Zabih, J. Miller, and K. Mai, "A feature-based algorithm for detecting and classifying scene breaks," in *Proc. 3rd ACM Int. Conf. Multimedia*, 1995, pp. 189–200.

[96] S. Zadtootaghaj, N. Barman, R. R. R. Rao, S. Goring, M. G. Martini, A. Raake, and S. Moller, "DEMI: Deep video quality estimation model using perceptual video quality dimensions," in *Proc. IEEE 22nd Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2020, pp. 1–6.

[97] S. Zadtootaghaj, S. Schmidt, S. S. Sabet, S. Möller, and C. Griwodz, "Quality estimation models for gaming video streaming services using perceptual video quality dimensions," in *Proc. 11th ACM Multimedia Syst. Conf.*, May 2020, pp. 213–224.

[98] E. Zerman, V. Hulusic, G. Valenzise, R. K. Mantiuk, and F. Dufaux, "The relation between MOS and pairwise comparisons and the importance of cross-content comparisons," *Electron. Imag.*, vol. 2018, no. 14, pp. 1–6, 2018.

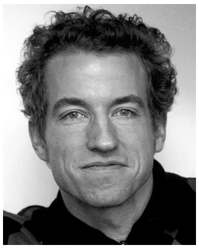
[99] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, vol. 2, Aug. 2004, pp. 28–31.

[100] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recognit. Lett.*, vol. 27, no. 7, pp. 773–780, May 2006.

STEVE GÖRING studied with TU Ilmenau and graduated the B.Sc. and M.Sc. degrees in computer science, from 2008 to 2013. He is currently a Computer Scientist working with the Audiovisual Technology Group, TU Ilmenau. His is also focus on data analysis problems for video quality models and video streams. Before he started 2016 with the Audiovisual Technology Group, he was working with the Big Data Analytics Group, Bauhaus University Weimar. His specializations are data analytics/machine learning, video quality, and distributed communication/information systems. His research focus in Weimar was improving search engines (using axiomatic re-ranking approaches), argumentation analysis, and analyzing large unstructured datasets using machine learning approaches.

RAKESH RAO RAMACHANDRA RAO received the M.Sc. degree in communications engineering from RWTH Aachen, in 2017, with focus on image content analysis and millimeter wave transmission systems. He has been an Electrical Engineer working with the Audiovisual Technology (AVT), TU Ilmenau, since 2017. His main focus is on video quality analysis and modeling. Before joining AVT, he worked as an intern with HEAD acoustics, where he worked on reference-based noise estimation. His specializations include video quality and image content analysis.

BERNHARD FEITEN received the D.Sc. degree in electronic engineering from Technische Universität Berlin in the field of psychoacoustics and audio bit rate reduction. He worked as an Assistant Professor with the Technische Universität Berlin in the field of communication science, digital signal processing, and computer music with "Elektronisches Studio". Since 1996, he has been with Deutsche Telekom (currently known as Technology and Innovation), working as a Senior Expert and a Project Manager for Innovative Multimedia Services, Quality of Experience, and Network Analytics. His research and development activities comprise audio and video coding quality, broadcasting applications, high quality Internet media distribution and streaming, QoE monitoring, and optimization.



ALEXANDER RAAKE (Member, IEEE) received the Dr.-Ing. degree from the Faculty of Electrical Engineering and Information Technology, Ruhr-Universität Bochum, in 2005, with the book *Speech Quality of VoIP*. From 2004 to 2005, he was a Postdoctoral Researcher with LIMSI-CNRS, Orsay, France. From 2005 to 2015, he held a Senior Researcher, an Assistant, and later an Associate Professor positions with TU Berlin's An-Institut T-Labs, a joint venture between Deutsche Telekom AG and TU Berlin, heading the Assessment of IP-based Applications Group. In 2015, he has joined TU Ilmenau as a Full Professor, where he heads the Audiovisual Technology Group. Since 1999, he has been involved with the ITU-T Study Group 12's standardization work on QoS and QoE assessment methods. His research interests include audiovisual and multimedia technology, speech, audio and video signals, human audiovisual perception, and Quality of Experience. He is a member of the Acoustical Society of America, the AES, VDE/ITG, and DEGA.