TECHNISCHE UNIVERSITÄT
ILMENAU

Faculty of Electrical Engineering and Information Technology
Institute for Media Technology
Audio Visual Technology

DISSERTATION

zur Erlangung des Akademischen Grades Doktoringenieur (Dr.-Ing.)

Data-Driven Visual Quality Estimation Using Machine Learning

---

vorgelegt von: Steve Göring
Betreuender Gutachter: Prof. Dr.-Ing. Alexander Raake
Gutachter: Prof. Dr. Patrick Le Callet
Gutachter: Prof. Dr. Sc. Lea Skorin-Kapov
Datum der Einreichung: 05.11.2021
Datum der wissenschaftlichen Aussprache: 18.05.2022

URN: urn:nbn:de:gbv:ilm1-2022000207
DOI: 10.22032/dbt.52210

To my mother.

Acknowledgments

Writing a PhD thesis is a journey of several years. In general, reading, collaborating, discussing, exchanging ideas and presenting work within the research community were important steps along the way.

I'm thankful to Alexander Raake, who supported me while doing research, with whom I had profound discussions, and who even pushed me when necessary to travel more often. I'd also like to thank Patrick Le Callet and Lea Skorin-Kapov for being reviewers of my thesis.

I still remember the first paper deadline, when everyone in the AVT group was working more or less day and night to finish their papers. And this was exactly the spirit of research that I was looking for. Having a last-minute review briefly before the final submission deadline led to another night shift that always paid off in the end. Even though incorporating the changes increased my coffee consumption exponentially.

It should also be mentioned that research, writing publications, and doing experiments is not always a success story, there may be ideas that never work out, papers that get rejected, or experiments that fail. This process may be frustrating, but having the ability to continue working on it is important, and a big help for this was the exchange with people like Rakesh, Stephan, Frank, Dominik, Janto, Ashu, Werner, and everyone else at the AVT group.

Doing research is not a process of a single person sitting in a dark room in the basement, it is teamwork and this is important, so thanks to all of the co-authors of

my publications. Besides the fruitful collaboration within the AVT group or other research groups, such as the work with Saman, Nabajeet, and Steven, I'm happy that I was part of several projects funded by Deutsche Telekom. The discussion about modeling, video quality, video bitstreams, and performing subjective testing with Bernhard, Peter, and Ulf formed an integral part of my path towards my PhD. I would also thank the institute staff members, especially Bernd and Monique for their support, help, and open ears within the last years. Even small talks about some other topics than research are important and were always pushing me to find more motivation.

The overall journey would have also not been possible without my friends Martin, Tim, Erik, Andrea, and Susi who were always motivating me during the last years, even though some Doppelkopf games were not making us happier. Finally, I want to thank my mother for the support throughout my life.

Abstract

Today a lot of visual content is accessible and produced, due to improvements in technology such as smartphones and the internet. This results in a need to assess the quality perceived by users to further improve the experience. However, only a few of the state-of-the-art quality models are specifically designed for higher resolutions, predict more than mean opinion score, or use machine learning. One goal of the thesis is to train and evaluate such machine learning models of higher resolutions with several datasets. At first, an objective evaluation of image quality in case of higher resolutions is performed. The images are compressed using video encoders, and it is shown that AV1 is best considering quality and compression. This evaluation is followed by the analysis of a crowdsourcing test in comparison with a lab test investigating image quality. Afterward, deep learning-based models for image quality prediction and an extension for video quality are proposed. However, the deep learning-based video quality model is not practically usable because of performance constrains. For this reason, pixel-based video quality models using well-motivated features covering image and motion aspects are proposed and evaluated. These models can be used to predict mean opinion scores for videos, or even to predict other video quality-related information, such as a rating distributions. The introduced model architecture can be applied to other video problems, such as video classification, gaming video quality prediction, gaming genre classification or encoding parameter estimation. Furthermore, one important aspect is the processing time of such models. Hence, a generic approach to speed up state-of-the-art video quality models is introduced, which shows that a significant amount of processing time can be saved, while achieving similar prediction accuracy. The models have been made publicly available as open source so that the developed frameworks can be used for further research. Moreover, the presented approaches may be usable as building blocks for newer media formats.

Zusammenfassung

Heutzutage werden viele visuelle Inhalte erstellt und sind zugänglich, was auf Verbesserungen der Technologie wie Smartphones und das Internet zurückzuführen ist. Es ist daher notwendig, die von den Nutzern wahrgenommene Qualität zu bewerten, um das Erlebnis weiter zu verbessern. Allerdings sind nur wenige der aktuellen Qualitätsmodelle speziell für höhere Auflösungen konzipiert, sagen mehr als nur den Mean Opinion Score vorher oder nutzen maschinelles Lernen. Ein Ziel dieser Arbeit ist es, solche maschinellen Modelle für höhere Auflösungen mit verschiedenen Datensätzen zu trainieren und zu evaluieren. Als Erstes wird eine objektive Analyse der Bildqualität bei höheren Auflösungen durchgeführt. Die Bilder wurden mit Video-Encodern komprimiert, hierbei weist AV1 die beste Qualität und Kompression auf. Anschließend werden die Ergebnisse eines Crowd-Sourcing-Tests mit einem Labortest bezüglich Bildqualität verglichen. Weiterhin werden auf Deep Learning basierende Modelle für die Vorhersage von Bild- und Videoqualität beschrieben. Das auf Deep Learning basierende Modell ist aufgrund der benötigten Ressourcen für die Vorhersage der Videoqualität in der Praxis nicht anwendbar. Aus diesem Grund werden pixelbasierte Videoqualitätsmodelle vorgeschlagen und ausgewertet, die aussagekräftige Features verwenden, welche Bild- und Bewegungsaspekte abdecken. Diese Modelle können zur Vorhersage von Mean Opinion Scores für Videos oder sogar für anderer Werte im Zusammenhang mit der Videoqualität verwendet werden, wie z. B. einer Bewertungsverteilung. Die vorgestellte Modellarchitektur kann auf andere Videoprobleme angewandt werden, wie z.B. Videoklassifizierung, Vorhersage der Qualität von Spielevideos, Klassifikation von Spielegenres oder der Klassifikation von Kodierungsparametern. Ein wichtiger Aspekt ist auch die Verarbeitungszeit solcher Modelle. Daher wird ein allgemeiner Ansatz zur Beschleunigung von State-of-the-Art-Videoqualitätsmodellen vorgestellt, der zeigt, dass ein erheblicher Teil der Verarbeitungszeit eingespart werden kann, während eine ähnliche Vorhersagegenauigkeit erhalten bleibt. Die Modelle sind als Open Source veröffentlicht, so dass die entwickelten Frameworks für weitere Forschungsarbeiten genutzt werden können. Außerdem können die vorgestellten Ansätze als Bausteine für neuere Medienformate verwendet werden.

Contents

Chapter 1

Introduction

Today media is consumed nearly everywhere and all the time, especially due to technologies like the internet, television and smartphones that can be used to create, view or share media in all possible situations. In contrast to early photographers and cinematographers everyone can now be a creator of visual content. In addition, such content can be distributed around the world faster than ever before, because internet and sharing platforms enable a global access and distribution within seconds or minutes.

Furthermore, consumption of visual media, i.e. images or videos (including audio), increases tremendously every year [vau19]. For example, statistical evaluations show that about 300k videos are uploaded per day on Youtube [Tur16] resulting in approximately 24 TB of storage. On top of that, 30 million images are uploaded on Flickr [Mic19] per day or 95 million images per day in the case of Instagram [99f20]. In addition, for example, Netflix offers about 15k movie titles [Nee20], or users can select to stream a movie out of about 18k movies accessible on Amazon Prime Video [Spa16]. Considering the enormous amount of accessible audiovisual content, e.g., comparing with the 14 GB compressed text-only data dump of Wikipedia [Wik20], it can be clearly concluded that we live in a world full of audiovisual content that grows heavily each day.

As a consequence, it is clear that not only media consumption also the creation of content is increasing, because they depend on each other, and moreover user-generated content is an essential part of our daily lifes. In general, more audiovisual content is uploaded and viewed, also because technologies allow high quality recordings

and better viewing experiences, whereas accessing visual media is getting easier and cheaper.

This directly leads to more content watched over the internet than before, resulting in an increase in bandwidth required for data transmission over the internet for audiovisual content in the last years. According to Cisco's forecast in 2018 [Cis18] video streaming will be about 80% to 90% of the total IP traffic in 2022. Whereas UHD-1/4K streamed video will increase up to 22% of the total internet traffic, e.g. because higher resolution screens are available and will replace Full-HD screens. Moreover, newer screens with even higher resolution are developed, e.g., Samsung already sells 8K TV screens since 2018 [Sam18]. In addition, the Japanese broadcast channel NHK delivers 4K or even 8K video content via satellite [NHK20].

Furthermore, not only the amount of visual content increases daily, but also the type, quality, and diversity of images and videos is varying a lot, and this leads to new challenges, for example for quality prediction, as it is shown in [WIA19] for a dataset consisting of user-generated videos from Youtube. Another side effect is that high-quality content requires more storage and also increases processing time, for example in case of pixel based video quality estimation or for performing any kind of per-frame image analysis.

Moreover, during the last years, more advanced methods have been developed to enable video streaming with reduced bandwidth and to increase the perceived quality. Early Youtube videos were streamed progressively, later using adaptive bitrate streaming or other adaptive streaming approaches [You15], which enabled streaming of videos of resolutions up to 8K, or in 360° or 3D format. Similar approaches are used by Vimeo, Netflix, Amazon Prime Video, and other video-on-demand providers. All in all these streaming technologies are usually based on *HTTP based adaptive streaming (HAS)* or *dynamic adaptive streaming using HTTP (DASH)* [PM11; Sto+11] (see also ISO/IEC 23009-1 [ISO19]). There are other streaming protocols such as IPTV [Lee07] or RTSP [SRL98], that differ from *DASH* because they do not using HTTP as the underlying protocol.

HTTP-based methods were more successful and can be even applied for live broadcast streaming [Loh+11]. In general, *DASH* or *HAS* based streaming can be used to stream classical 2D video content, audio (e.g. Spotify [Sch+18; KN10]), 3D videos,

360° videos, and even more (e.g. 3D objects for augmented reality applications, where adaption handles different level of detail settings [Pet+19]). HTTP-based media streaming technologies are more fruitful because usually no advanced setup is required. Especially because each browser can access the media files using HTTP or HTTPS, which are the most widely used protocols of the internet. Furthermore back-end servers only serve files and are not required to have advanced client management (even though currently developed extensions, like SAND are adding server-side control and monitoring mechanisms [Tho+16]) and such servers can easily scale using webserver caching approaches, e.g., in the form of content delivery networks [Tho+15]. To sum up, all these reasons make HTTP-based adaptive video streaming so important and successful for streaming providers and finally also end-users who rely on this technology for media consumption.

Besides streaming methods also video compression is a key technology used to decrease the transmitted bandwidth of our daily consumed visual media while guaranteeing a good overall quality. In general, there exist two compression terms: lossless, where the source information can be reconstructed without losing any information; and lossy, where a specific loss of information is accepted after decompression [BK97]. Lossy-based compression enables video compression to be more effective leading to smaller file sizes for transmission, where the visual loss can be accepted or is nearly not perceivable. For example, usual video frames are highly redundant, e.g. spatially or temporally [SS08, p.4 ff], moreover the human visual system (HVS) does not perceive videos in the same way as a sensor [SS08, p.9 ff], for example, luminance masking can be used to reduce video data tremendously.

Here it should be mentioned, that the majority of the required bandwidth for transmission of the audiovisual content is utilized by the video signal, though audio signals are also transmitted. Usually, typical streaming providers support only a small number of different audio representations, whereas more options for video are provided to enable easy adaption in case of bandwidth changes over time. In addition, the required storage for audio is nearly negligible in comparison to video for typical entertainment content, movies, or television series.

Moreover, such video and image compression algorithms, also referred to as codecs, are in constant development and improvement. For example, an H.265 encoded video can have the same quality as a similar encoded H.264 video that has double the

filesize [ŘE14]. The recently developed codec AV1 [Ope20] can encode videos with similar perceived quality with even lower filesizes than H.265 [AE18; Rao+19b]. In addition, similar bitrate savings can be achieved in the case of higher resolutions or framerates, e.g. UHD-1/4K with 60 frames per second [Rao+19b; Rao+19a], enabling the video providers to also stream in higher resolutions.

Further developments can be observed for image compression as well, e.g. starting from classical JPEG, JPEG-XR, or JPEG-2000 to more advanced image compression methods like HEIF [Lai+16; Nok20] or AVIF [Ope19] that are based on video encoding technology. The main goal is similar to the video case and is the reduction of the required storage size or transmission bandwidth for images, while maintaining high visual quality. However, images require less storage than videos in common applications, technology developments with increasing image sensor resolutions and possibilities to share images automatically lead to a higher transmission bandwidth required for images and thus a need for better compression methods.

Furthermore, developments in compression of 360° video [Sku+17] or point cloud objects in the case of augmented or virtual reality applications [SSG18; GRP18] target the same goal of reducing the required file size for transmission.

From a global point of view, such compression methods are based on several developments, whereas results from human perception are the most important key element to successfully compress an image or video in a lossy manner with a respective high visual quality [Abo+10; Lin10]. An important question directly connected to lossy compression and *DASH* streaming (or in general any multi representation-based adaptive streaming approach) is how typical users perceive the streamed audiovisual content. For this, the person who consumes the media is the most important and thus requires additional investigation and attention.

1.1 Quality of Experience and Linked Areas

How a user rates the quality of a given service leads to a commonly used and defined term called *Quality of Experience (QoE)* where the user is in the center of the analysis, this term originated from an earlier service specific view that is referred to as *Quality of Service (QoS)*. To get a better understanding of the term *QoE*, it is required to

consider as first step some definitions, for example, the widely used Definition 1 proposed by Le Callet, Möller, Perkis, et al. [LMP+12].

> **Definition 1** *"Quality of Experience . . . is the degree of delight or annoyance of the user of an application or service. It results from the fulfillment of his or her expectations with respect to the utility and / or enjoyment of the application or service in the light of the user's personality and current state." [LMP+12]*

> **Definition 2** *"Quality of Experience . . . is the degree of delight or annoyance of a person whose experiencing involves an application, service, or system. It results from the person's evaluation of the fulfillment of his or her expectations and needs with respect to the utility and / or enjoyment in the light of the person's context, personality and current state" [RE14]*

An extension of the first introduced definition can be found in [RE14]. The main extension of the Definition 2 is to include a broadened view of the person involved in the overall quality formation process.

These definitions of QoE highlight that the user is more in the focus of the measurement or estimation of quality, thus the service or application should be designed in a way to satisfy the expectations of the end-user for the given use case of the service.

Especially because humans play a central role for *QoE*, several influence factors can be observed and form the fundamental basis of all *QoE* analysis [Bru+13; LMP+12; Rei+14]. In general, compare Figure 1.1, three main factors can be identified, first *system factors*, second *context factors* and third *human factors* [Bru+13; LMP+12; Rei+14].

The category *system factors* are mainly related to the typical end-to-end video processing chain, starting from the production of the video to the encoding, transmission, storage, and final reception of the end-user. In each of the mentioned steps, quality changes can occur and moreover may also be required. For example, the uncompressed video signal is important for cutting and post-production of the video before the content is released. Assuming the uncompressed material is not compressed, it will lead to a high amount of required bandwidth and more powerful hardware during the reception at the end-users side. Here a compression with a state-of-the-art video codec will enable less storage and lower hardware requirements.

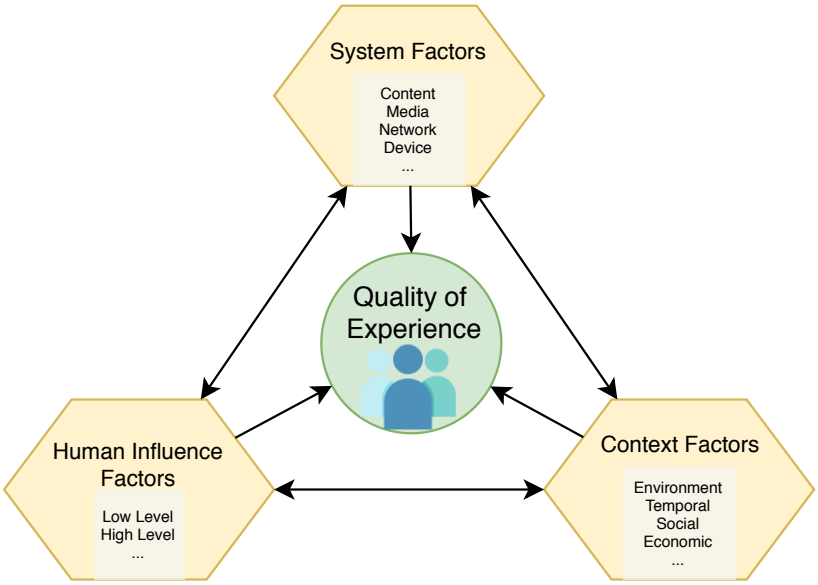


Figure 1.1: General influence factors for Quality of Experience [Bru+13; LMP+12; Rei+14].

The second main influence to the overall *QoE* are *context factors*. They subsume for example social, environmental, temporal, economical dependencies, that have an impact on the perception of users.

As of last, *human factors* are distinguished, they correspond to demographic, mental, emotional properties of the user. Usually, they are classified into low and high-level processing factors, for example, low-level factors would refer to the mental constitution of the user, whereas high-level can be related to any interpretation of the shown content or personal preference.

All these factors have an influence on the perceived quality of a user and furthermore, depend on and affect each other. For example, in the case of a rating in a subjective video quality test, a user can favor specific movie genres. Here, typical human factors can have an influence on such test results, for example, when the test needs to be finished quickly because the participant has an appointment after the test, or is less interested in the test and more in the final payment. Also, the test design itself has an influence, e.g., an unbalanced design of quality distortions could lead to range equalization effects in the results of the test, as it is discussed in [ZRB08]. In such cases, the interpretation of collected results is the most challenging part. In addition, for prediction models, it is also important to know the scope of the model for, e.g.,

quality estimation, so that training and validation data can be adequately defined or collected with perception tests.

However, *QoE* is also related to *QoS*. The term *QoS* is defined as follows in Definition 3.

> **Definition 3** *"Totality of characteristics of a telecommunications service that bear on its ability to satisfy stated and implied needs of the user of the service." [ITU08a]*

Comparing the *QoE* and *QoS* definitions, *QoE* can be seen as an extension of *QoS* with additional influence factors and shift towards the user as being more important than the service. *QoS* is an assessment of *system factors* primarily from a technical standpoint. This can also be explained from a business point of view, when a service is not able to satisfy the expected needs of a user, the user will probably not pay or use the service in the future. Sackl et al. [Sac+12] evaluate the connection of QoE and economic aspects, where they observe a willingness of participants to pay own money for enhancing quality.

In the general definition of *QoE*, the system influence factor is highly related to *QoS*, however, this is not the only defining part of user's perception.

For example, the combination of human and system influence factors of *QoE* leads to the importance of the content that is provided by the service. Especially the final human process of assessing a quality score in for example a lab test depends further on factors related to the presented stimuli. An evaluation of *QoE* should therefore also include aspects of content likability or aesthetics.

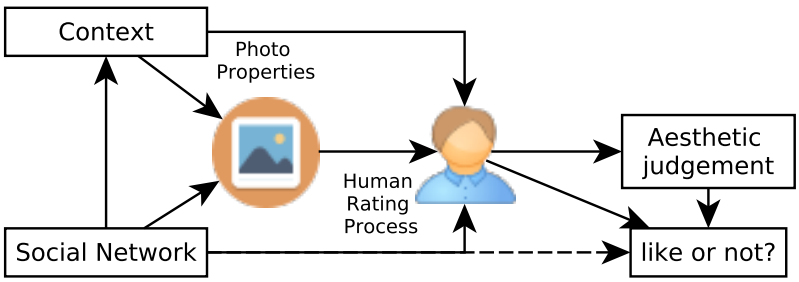


Figure 1.2: How humans rate aesthetic and decide liking, based on [Led+04] used in [GBR18].

For example, in Figure 1.2 a summarized and extended view of Leder et al.'s model of aesthetics ratings is presented, where the specific inclusion of the final liking decisions [GBR18] is in focus. Three main factors that influence each other are important for human aesthetics judgment or the final liking decision: the photo or likewise a video (thus including the quality), the context, and the social influences [Led+04].

Such factors can be interpreted in terms of the introduced *QoE* influence factors, leading to the need to include aesthetics also in *QoE*-related research questions. For example, if a service only provides content that is of low quality or low acceptance, there is also no need to pay or use the provided service in the normal case, however there are exceptions where low-quality content becomes famous or viral.

Additional research fields that are also linked to the field of *QoE* are Big Data, Computer Vision, Machine or Deep Learning. The main reason for such an inclusion of areas is that content diversity, the number of users, and more are increasing, which automatically goes along with larger data that needs to be analyzed.

In the following, whenever the terms image or video quality are used they usually are related to quality perceived by users that is a part of the overall media experience thus *QoE*. Moreover, it is also clear that in usually conducted video quality tests, which form the ground truth for developed prediction models, not all known influence factors can be included, e.g., because they cannot be estimated or measured.

1.2 DASH Streaming and Encoding

As mentioned before, streaming technologies evolved from video downloads to progressive streamed videos to more sophisticated approaches. One famous and mostly used approach in this context is *DASH*.

In general, a *DASH* streaming setup, consists of a web server storing different encoded representations of the media stream, i.e., a video, subtitles, or an audio stream [PM11; Sto+11; ISO19]. Such servers can be replicated in an easy way to enable lower latencies and scalable access to different streams around the world in so-called content delivery networks [Tho+15]. Starting from a given media stream, using specific encoding strategies (for YouTube compare with [Che+17]), e.g., the

simplest would be a fixed bitrate ladder with different (bitrate, resolution, framerate, codec)-combinations for video and similar for audio, several representations will be created. After encoding, these representations are chunked to segments with lengths varying from 2 to 10 seconds, depending on the service, the use case, and the targeted client devices [PM11]. On the server, the encoded segments and a manifest file that includes the assignment of each segment to a representation and additional meta-data are stored.

A user would now request the media stream using a specific client, that handles the *DASH* playout, i.e., downloading the manifest file, initial segments, and video segments. Moreover, the client handles which segment is required based on the player behavior considering, e.g. bandwidth fluctuations, buffer changes, or user preference, whether it is required to perform a switch to lower or higher quality representation.

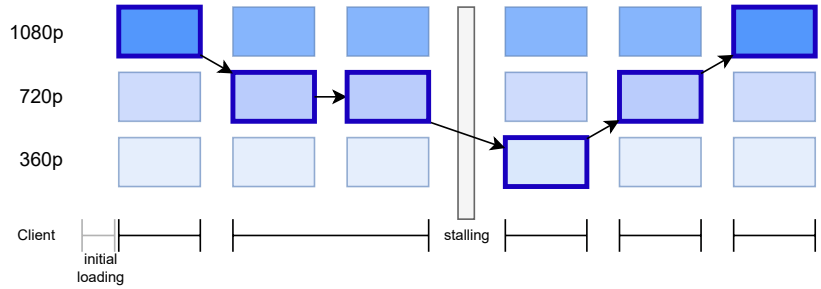


Figure 1.3: Example of a DASH client streaming different segments, starting from 1080p resolution to 720p, with stalling, then to 360p, 720p and 1080p; resolutions are used to represent different quality representations.

In Figure 1.3 such a typical playout is illustrated. In general, the time required from the moment where the user requests the video to the first played video frame is defined as *initial loading delay* [ITU17], whereas moments during playout where the player buffer is empty and needs to request more segments resulting in a frozen video are called *stalling* [ITU17; ITU16a], where the time of occurrence and lengths are important characteristics. The *DASH* client starts playing segments of the initial representation as long as network bandwidth is available, and if measures show sufficient remaining bandwidth, a higher quality representation is requested and further segments of this representations are played. In case the bandwidth is not enough the client selects a lower quality representation leading to follow-up requests and playing of such segments. In general, modern *DASH* clients avoid stalling, as it

is for example shown in [WWC19] for YouTube and smartphones for the year 2018, where about 80% of video playbacks have less than 1 stalling event. In addition, Wassermann, Wehner, and Casas [WWC19] show that within the last years the number of stalling events while playing a video decreased. In addition, a usual *DASH* player also adapts based on previously played videos [Rob+18b]. Moreover, typical encoding settings for *DASH* representations are not static anymore, e.g., Youtube uses a per-title optimized encoding strategy, or Netflix even uses a per scene optimized encoding approach [Kat18].

1.3 Image and Video Quality Estimation

There are several possible application areas for image and video quality estimation methods, for example in Figure 1.4 a typical end-to-end delivery chain is shown for the video streaming application case.

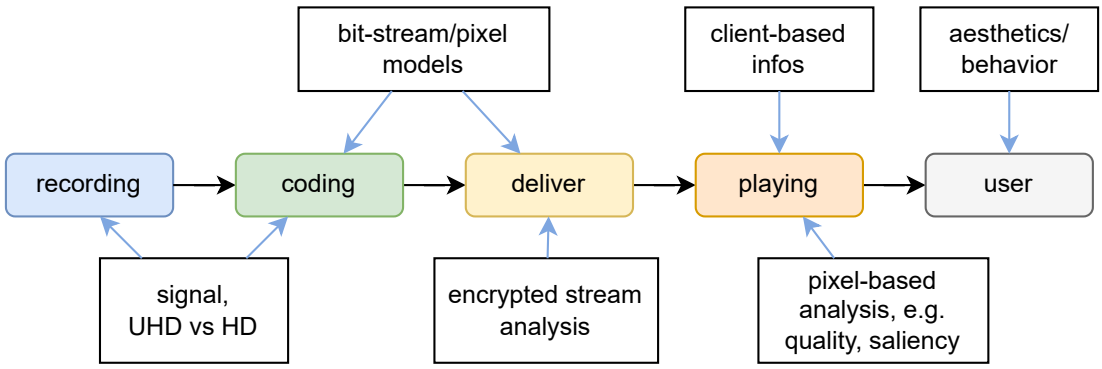


Figure 1.4: A typical end-to-end delivery chain for visual content, possible intermediate steps are highlighted, here QoE or related predictions can be applied.

Here, in the first part of the chain, video quality estimation can be used to analyze recorded videos to increase the quality already before delivering the content. In addition based on *QoE* estimation, encoding strategies can be optimized to reduce file storage and the transmitted bandwidth of streaming providers. Furthermore, image quality estimation can be used to optimize compression of images or to enable better

sorting of search results in the case of image sharing platforms. Moreover, general *QoE* monitoring can be used for encrypted [OS20] or unencrypted video or audio streaming to enable internet service providers to increase the customer's satisfaction in using specific streaming services. Finally, due to the fact that user-based quality rating depends also on the liking or aesthetic of the shown content, user behavior can be evaluated, e.g. based on the analysis of played content considering the region of interest or quality aspects.

1.4 QoE Modeling using Machine Learning

In general, the mentioned increase of content diversity or higher pixel densities for sensors and thus higher resolutions leads to the goal of developing robust *QoE* models. Moreover, also the need for adaption to newer contents or visual distortion is required for modern *QoE* estimation.

Besides traditional *QoE* models, machine learning can be used to analyze large datasets, where patterns that have been used in the past are not clearly visible for non-machine learning modeling approaches. Here, the increase of quality and diversity of the available visual contents creates a need for more sophisticated and scalable models.

Typical machine learning models can be used for such quality estimations, e.g., based on defined and well-motivated feature sets [Hua+18]. Those models can be trained using ground truth data and finally evaluated. Such ground truth data is based on conducted subjective tests or uses synthetically generated data. In case new or different visual contents are occurring, e.g., higher resolutions, different rating schemes, new compression methods, such models can easily be re-trained, and re-validated. However, also newer features could be required, which can be included in such a machine learning pipeline.

1.5 Research Questions and Goals

Considering the previously mentioned problems in the current visual quality research, the following questions will be handled in this thesis.

> **Research Question 1** *To which extent can machine learning models be used to develop robust video and image quality models, capable of handling content diversity? Which features are best suitable for quality prediction? How good are such models, considering unknown datasets and limitations in training data?*

The main focus in Research Question 1 is the development of robust video and image quality models especially for high-quality content (higher resolutions, framerates, ...), where one important part will be the evaluation that is performed on unknown datasets to prove the generalizability of the developed concepts. Moreover, several machine learning approaches will be considered, to validate which are the best and most suitable ones. Furthermore, the introduced model framework can be easily extended to newer distortion types, using re-training or new features, and are applicable to several other video-related problems.

> **Research Question 2** *How can the overall processing time of state-of-the-art quality models be reduced, without introducing a large error?*

Besides a high correlation or low error with the user ratings, it is also important to consider the computation time of such models. Here, Research Question 2 reflects how much computation time is required and whether it is possible to reduce this time with a low overall prediction error. Moreover, the concepts used to speed up such models will be included in the developed models, that are addressed in Research Question 1.

> **Research Question 3** *Where are the limits of perception in the case of higher quality content and is it possible to predict when there is no perceivable difference?*

Especially because of the increase of resolution for several visual content types, e.g., UHD-1/4k or UHD-2/8K videos, or super-high-resolution images, the limits of perceptions will be addressed and investigated in Research Question 3. Moreover, using machine learning it will be analyzed whether such limits can be predicted or not. Such a prediction system could be used for encoding optimization or source content quality checks.

Research Question 4 *Can machine learning models be used to predict more than mean opinion scores of subjectively rated visual content?*

QoE and quality prediction of visual contents cannot be fully represented in a single decimal value for a given stimulus, several influence factors are not considered in a *mean opinion score (MOS)*. Machine learning models will be used to tackle the visual quality prediction problem as classification, single regression, and multi-instance regression problems, to demonstrate that such models can be used beyond classical mean opinion score prediction.

Research Question 5 *Can video quality models/compression be applied to images?*

Finally, the Question 5 is interconnected to all aforementioned questions, because features, model structures, and approaches will be evaluated for videos and images. The core idea here is that approaches developed for videos can be used in a similar fashion also for image-related prediction problems.

1.6 Relevant Contributions

In the following Sections, contributions of the author to the video quality and *QoE* research field are summarized and categorized. As a hint for the reader, such own contributions are color-coded in the remaining text, e.g., [GSR18], external references are color-coded as [BM19]. First, the relevant publications for this thesis are highlighted. Second, to increase reproducibility the author created several open-source software projects, they can be used for further analysis or new research in the

field of *QoE*. The published software was also used for, e.g., performing subjective tests that are presented and used in this thesis, for training and validation of machine learning approaches, and more.

1.6.1 Publications

The following publications are grouped according to their relevance for this thesis, thus first the main publications are listed, later highly related publications, and finally publications that cover different scopes of *QoE* or are related to quality.

Pixel-based image/video quality models, image appeal, features, or analysis.

[GSR18] **Steve Göring**, Janto Skowronek, and Alexander Raake. "DeViQ – A deep no reference video quality model". In: *Electronic Imaging, Human Vision Electronic Imaging* 2018.14 (2018), pp. 1–6

[GBR18] **Steve Göring**, Konstantin Brand, and Alexander Raake. "Extended Features using Machine Learning Techniques for Photo Liking Prediction". In: *Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. Sardinia, Italy, May 2018

[GR18] **Steve Göring** and Alexander Raake. "deimeq – A Deep Neural Network Based Hybrid No-reference Image Quality Model". In: *7th European Workshop on Visual Information Processing (EUVIP)*. IEEE. Nov. 2018, pp. 1–6. DOI: 10.1109/EUVIP.2018.8611703

[Gör+19] **Steve Göring**, Julian Zebelein, Simon Wedel, Dominik Keller, and Alexander Raake. "Analyze And Predict the Perceptibility of UHD Video Contents". In: *Electronic Imaging, Human Vision Electronic Imaging* 2019.12 (2019)

[GRR19] **Steve Göring**, Rakesh Rao Ramachandra Rao, and Alexander Raake. "nofu - A Lightweight No-Reference Pixel Based Video Quality Model for Gaming Content". In: *Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*. Berlin, Germany, June 2019

[GR19] **Steve Göring** and Alexander Raake. "Evaluation of Intra-coding based image compression". In: *8th European Workshop on Visual Information Processing (EUVIP)*. IEEE. 2019, pp. 1–6

[GKR19] **Steve Göring**, Christopher Krämmer, and Alexander Raake. "cencro – Speedup of Video Quality Calculation using Center Cropping". In: *21st IEEE International Symposium on Multimedia (IEEE ISM)*. Dec. 2019, pp. 1–8

[Rao+19a] Rakesh Rao Ramachandra Rao, **Steve Göring**, Werner Robitza, Bernhard Feiten, and Alexander Raake. "AVT-VQDB-UHD-1: A Large Scale Video Quality Database for UHD-1". In: *21st IEEE International Symposium on Multimedia (IEEE ISM)*. Dec. 2019, pp. 1–8

[Gör+20] **Steve Göring**, Robert Steger, Rakesh Ramachandra Rao Rao, and Alexander Raake. "Automated Genre Classification for Gaming Videos". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6

[Zad+20a] Saman Zadtootaghaj, Nabajeet Barman, Rakesh Ramachandra Rao Rao, **Steve Göring**, Maria G. Martini, Alexander Raake, and Sebastian Möller. "DEMI: Deep Video Quality Estimation Model using Perceptual Video Quality Dimensions". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6

[RGR21a] Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Enhancement of Pixel-based Video Quality Models using Meta-data". In: *Electronic Imaging, Human Vision Electronic Imaging*. 2021

[Gör+21a] **Steve Göring**, Rakesh Rao Ramachandra Rao, Bernhard Feiten, and Alexander Raake. "Modular Framework and Instances of Pixel-based Video Quality Models for UHD-1/4K". in: *IEEE Access* 9 (2021), pp. 31842–31864. DOI: 10.1109/ACCESS.2021.3059932. URL: https://ieeexplore.ieee.org/document/9355144

[RGR21b] Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Towards High Resolution Video Quality Assessment in the Crowd". In: *13th International Conference on Quality of Multimedia Experience (QoMEX)*. 2021

[GR21] **Steve Göring** and Alexander Raake. "Rule of Thirds and Simplicity for Image Aesthetics using Deep Neural Networks". In: *2021 IEEE 23st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2021, pp. 1–6

[Gör+21b] **Steve Göring**, Rakesh Rao Ramachandra Rao, Stephan Fremerey, and Alexander Raake. "AVRate Voyager: an open source online testing platform". In: *2021 IEEE 23st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2021, pp. 1–6

ITU-T Rec. P.1203, P.1204, or bitstream-related publications.

[GRF17] **Steve Göring**, Alexander Raake, and Bernhard Feiten. "A framework for QoE analysis of encrypted video streams". In: *Ninth International Conference on Quality of Multimedia Experience (QoMEX)*. May 2017, pp. 1–3. DOI: 10.1109/QoMEX.2017.7965640

[Raa+17] Alexander Raake, Marie-Neige Garcia, Werner Robitza, Peter List, **Steve Göring**, and Bernhard Feiten. "A bitstream-based, scalable video-quality model for HTTP adaptive streaming: ITU-T P.1203.1". In: *Ninth International Conference on Quality of Multimedia Experience (QoMEX)*. May 2017, pp. 1–6. DOI: 10.1109/QoMEX.2017.7965631

[Rob+18b] Werner Robitza, Dhananjaya G Kittur, Alexander M Dethof, **Steve Göring**, Bernhard Feiten, and Alexander Raake. "Measuring YouTube QoE with ITU-T P. 1203 under Constrained Bandwidth Conditions". In: *Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2018, pp. 1–6

[Rob+18a] Werner Robitza, **Steve Göring**, Alexander Raake, David Lindegren, Gunnar Heikkilä, Jörgen Gustafsson, Peter List, Bernhard Feiten, Ulf Wüstenhagen, Marie-Neige Garcia, Kazuhisa Yamagishi, and Simon Broom. "HTTP Adaptive Streaming QoE Estimation with ITU-T Rec. P.1203 – Open Databases and Software". In: *9th ACM Multimedia Systems Conference*. Amsterdam, 2018. ISBN: 9781450351928. DOI: 10.1145/3204949.3208124

[Rao+19b] Rakesh Rao Ramachandra Rao, **Steve Göring**, Patrick Vogel, Nicolas Pachatz, Juan Jose Villamar Villarreal, Werner Robitza, Peter List, Bernhard Feiten, and Alexander Raake. "Adaptive video streaming with current codecs and formats:

Extensions to parametric video quality model ITU-T P.1203". In: *Electronic Imaging* (2019)

[Rob+20] Werner Robitza, Alexander M. Dethof, **Steve Göring**, Alexander Raake, Tim Polzehl, and Andre Beyer. "Are You Still Watching? Streaming Video Quality and Engagement Assessment in the Crowd". In: *Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. Athlone, Ireland, May 2020

[GRR20] **Steve Göring**, Rakesh Rao Ramachandra Rao, and Alexander Raake. "Prenc – Predict Number Of Video Encoding Passes With Machine Learning". In: *Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. Athlone, Ireland, May 2020

[Rao+20b] Rakesh Rao Ramachandra Rao, **Steve Göring**, Peter List, Werner Robitza, Bernhard Feiten, Ulf Wüstenhagen, and Alexander Raake. "Bitstream-based Model Standard for 4K/UHD: ITU-T P.1204.3 – Model Details, Evaluation, Analysis and Open Source Implementation". In: *Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. Athlone, Ireland, May 2020

[Rao+20a] Rakesh Ramachandra Rao Rao, **Steve Göring**, Robert Steger, Saman Zadtootaghaj, Nabajeet Barman, Stephan Fremerey, Sebastian Möller, and Alexander Raake. "A Large-scale Evaluation of the bitstream-based video-quality model ITU-T P.1204.3 on Gaming Content". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6

[Raa+20] Alexander Raake, Silvio Borer, Shahid Satti, Jörgen Gustafsson, Rakesh Rao Ramachandra Rao, Stefano Medagli, Peter List, **Steve Göring**, David Lindero, Werner Robitza, Gunnar Heikkilä, Simon Broom, Christian Schmidmer, Bernhard Feiten, Ulf Wüstenhagen, Thomas Wittmann, Matthias Obermann, and Roland Bitto. "Multi-model standard for bitstream-, pixel-based and hybrid video quality assessment of UHD/4K: ITU-T P.1204". In: *IEEE Access* 8 (2020), pp. 193020–193049. DOI: 10.1109/ACCESS.2020.3032080. URL: https://ieeexplore.ieee.org/document/9234526?source=authoralert

▷ Contributions to standardization: ITU-T Rec. P.1203.1 [ITU17], P.1204, especially P.1204.3 [ITU19a; ITU19b]

Other publications, mostly related to virtual reality.

[Fre+19b] Stephan Fremerey, Rachel Huang, **Steve Göring**, and Alexander Raake. "Are people pixel-peeping 360° videos?" In: *Electronic Imaging, Human Vision Electronic Imaging* (2019)

[Sin+19a] Ashutosh Singla, **Steve Göring**, Alexander Raake, Rob Koenen, Britta Meixner, and Thomas Buchholz. "Subjective Quality Evaluation of Tile-based Streaming for Omnidirectional Videos". In: *10th ACM Multimedia Systems Conference*. Amherst, MA, USA, 2019

[Sin+19b] Ashutosh Singla, Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Assessing Media QoE, Simulator Sickness and Presence for Omnidirectional Videos with Different Test Protocols". In: *26th IEEE Conference on Virtual Reality and 3D User Interfaces*. Osaka, Japan, Mar. 2019

[Fre+19a] Stephan Fremerey, Frank Hofmeyer, **Steve Göring**, and Alexander Raake. "Impact of Various Motion Interpolation Algorithms on 360° Video QoE". in: *Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*. June 2019, pp. 1–3. DOI: 10.1109/QoMEX.2019.8743307

[Fre+20a] Stephan Fremerey, **Steve Göring**, Rakesh Ramachandra Rao Rao, Rachel Huang, and Alexander Raake. "Subjective Test Dataset and Meta-data-based Models for 360° Streaming Video Quality". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6

[Fre+20b] Stephan Fremerey, Frank Hofmeyer, **Steve Göring**, Dominik Keller, and Alexander Raake. "Between the Frames - Evaluation of Various Motion Interpolation Algorithms to Improve 360° Video Quality". In: *22st IEEE International Symposium on Multimedia (IEEE ISM)*. Dec. 2020, pp. 1–8

[Sin+21] Ashutosh Singla, **Steve Göring**, Dominik Keller, Rakesh Rao Ramachandra Rao, Stephan Fremerey, and Alexander Raake. "Assessment of the Simulator Sickness Questionnaire for Omnidirectional Videos". In: *IEEE Virtual Reality and 3D User Interfaces (VR)*. 2021. URL: https://conferences.computer.org/vrpub/pdfs/VR2021-2AyvgnPUHcYon9QQHz6BPD/255600a198/255600a198.pdf

[Rob+21] Werner Robitza, Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Impact of Spatial and Temporal Information on Video Quality and Compressibility". In: *13th International Conference on Quality of Multimedia Experience (QoMEX)*. 2021

[Kel+21] Dominik Keller, Markus Vaalgamaa, Erkki Paajanen, Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Groovability: Using Groove as a Novel Measure for Audio QoE with the Example of Smartphones". In: *13th International Conference on Quality of Multimedia Experience (QoMEX)*. 2021

1.6.2 Open Source Software and Open Data

Besides the aforementioned publications, another important part for reproducible research is that the used software and or the data are available for other researchers, for these reasons some software tools that are used in the listed publications were published as open source.

- *AVRateNG*[1]: Software for collecting subjective ratings for videos and images; used for all own conducted classic subjective tests in this thesis
- *AVrateVoyager*[2]: Extension of AVRateNG for online or crowdsourcing based tests, used and described in [RGR21b; Gör+21b].
- *ITU-T Rec. P.1203 open source implementation*[3]: Reference implementation of ITU-T Rec. P.1203; see [Rob+18a].
- *ITU-P.1203 codec extension*[4]: Extension of P.1203 to support codecs HEVC and VP9; was extended in [Rao+19b].
- *cencro*[5]: A center cropped variant of Netflix's VMAF to speedup quality calculations with low error; see [GKR19].

[1] https://github.com/Telecommunication-Telemedia-Assessment/avrateNG
[2] https://github.com/Telecommunication-Telemedia-Assessment/AVrateVoyager
[3] https://github.com/itu-p1203/itu-p1203/
[4] https://github.com/Telecommunication-Telemedia-Assessment/itu-p1203-codecextension
[5] https://github.com/Telecommunication-Telemedia-Assessment/cencro

- *quat*[6]: Framework for quality analysis experiments, including features, machine learning pipelines and more, it is mostly used in this thesis to develop the mentioned models; see [Gör+21a].
- *pixelmodels*[7]: Open source implementation of the developed video quality models, using *quat*; see [Gör+21a].
- *avc bitstream parser*[8], *hevc bitstream parser*[9] and *av1 bitstream parser*[10]: Parser for AVC (H.264), HEVC (H.265) and AV1 encoded videos to develop bitstream based-models.
- *image compression*[11]: Evaluation of intra-coding based image Compression; see [GR19].
- *AVT-VQDB-UHD-1*[12]: A database consisting of data for conducted UHD-1/4K video quality tests, presented in [Rao+19a].
- *ITU-P.1204.3 reference implementation*[13]: P.1204.3 reference implementation, presented in [Rao+20b].
- *ITU-P.1204.3 video bitstream parser*[14]: Contributions to the reference vidoe parser for the P.1204.3 prediction model; see [Rao+20b].

1.7 Thesis Structure

The main goal of this thesis is to handle the defined research questions that are in the scope of visual quality estimation.

[6] https://github.com/Telecommunication-Telemedia-Assessment/quat
[7] https://github.com/Telecommunication-Telemedia-Assessment/pixelmodels
[8] https://github.com/stg7/avc_bitstreamparser
[9] https://github.com/stg7/hevc_bitstreamparser
[10] https://github.com/stg7/av1_bitstreamparser
[11] https://github.com/Telecommunication-Telemedia-Assessment/image_compression
[12] https://github.com/Telecommunication-Telemedia-Assessment/AVT-VQDB-UHD-1
[13] https://github.com/Telecommunication-Telemedia-Assessment/bitstream_mode3_p1204_3
[14] https://github.com/Telecommunication-Telemedia-Assessment/bitstream_mode3_videoparser

To enable a good understanding of the answers to these questions it is thus required to introduce some general concepts of current state-of-the-art *QoE* models and approaches, starting from a proper understanding of high-quality content, basic principles of machine learning for *QoE* to a review of *QoE* models for images and videos in Chapter 2 "QoE Models and Approaches".

Afterwards the main contributions of the author to the field of image and video quality estimation are highlighted in Chapter 3 "High Resolution Image Quality Evaluation" for image related problems, for video quality in Chapter 4 "Models for Video Quality Prediction" and other applications of the introduced architecture in Chapter 5 "Other Applications of the Model Pipeline". All three chapters are based on machine learning methods combined with classical computer vision or deep learning-based features. One focus is further, to have a deeper investigation into the limits of perception for users, e.g., comparing UHD-1/4K with Full-HD videos, because higher image and video resolutions are streamed and shared by users. Moreover, the presented machine learning algorithms require carefully selected or created datasets for training and validation that are presented in detail in each of the chapters. Some datasets are based on subjective tests conducted in collaboration with or by the author, some are based on large downloaded datasets that are partially available or already shared with the research community. Considering prediction systems, a robust evaluation is required and will be further discussed in detail, with several variations to enable a proper and wide use case of the resulting models. In addition, other applications and extensions of the introduced models, the features, and the general prediction approaches are presented and discussed to enable the usage of the presented methods, tools, and software parts for future experiments and research.

Finally, in Chapter 6 "Conclusion and Future Work", a brief conclusion of the thesis is presented, where also future work aspects are highlighted and open challenges in the context of machine-learning-based visual quality prediction and *QoE* are summarized.

Chapter 2

QoE Models and Approaches

The following chapter will provide an in-depth overview of several QoE models and approaches to predict quality ratings for images or videos. It is further required to start with some basic principles, e.g., to clarify what a UHD-1/4K video is, or what kind of machine learning methods are suitable for visual quality prediction.

The chapter is partially based on the following publications:

[Gör+21a] **Steve Göring**, Rakesh Rao Ramachandra Rao, Bernhard Feiten, and Alexander Raake. "Modular Framework and Instances of Pixel-based Video Quality Models for UHD-1/4K". in: *IEEE Access* 9 (2021), pp. 31842–31864. DOI: 10.1109/ACCESS.2021.3059932. URL: https://ieeexplore.ieee.org/document/9355144

[GR19] **Steve Göring** and Alexander Raake. "Evaluation of Intra-coding based image compression". In: *8th European Workshop on Visual Information Processing (EUVIP)*. IEEE. 2019, pp. 1–6

[Gör+19] **Steve Göring**, Julian Zebelein, Simon Wedel, Dominik Keller, and Alexander Raake. "Analyze And Predict the Perceptibility of UHD Video Contents". In: *Electronic Imaging, Human Vision Electronic Imaging* 2019.12 (2019)

[GR18] **Steve Göring** and Alexander Raake. "deimeq – A Deep Neural Network Based Hybrid No-reference Image Quality Model". In: *7th European Workshop on Visual Information Processing (EUVIP)*. IEEE. Nov. 2018, pp. 1–6. DOI: 10.1109/EUVIP.2018.8611703

2.1 High Resolution Images and UHD-Videos

Considering the increase of pixel densities for sensors and displays and the availability of devices for recording and presentation, it is clear that modern *QoE* models should address higher resolutions. Smartphones and recent cameras are able to take photos with at least 16 *megapixels (MP)*. For example, the Samsung S20 smartphone [Sam20] from 2020 has several cameras ranging from 12 *MP* to 108 *MP*. A typical smartphone has a 16 *MP* camera sensor which leads, for example, to an image resolution of 4920 × 3264. Accordingly, in recently published datasets similar high image resolutions are included [GR19].

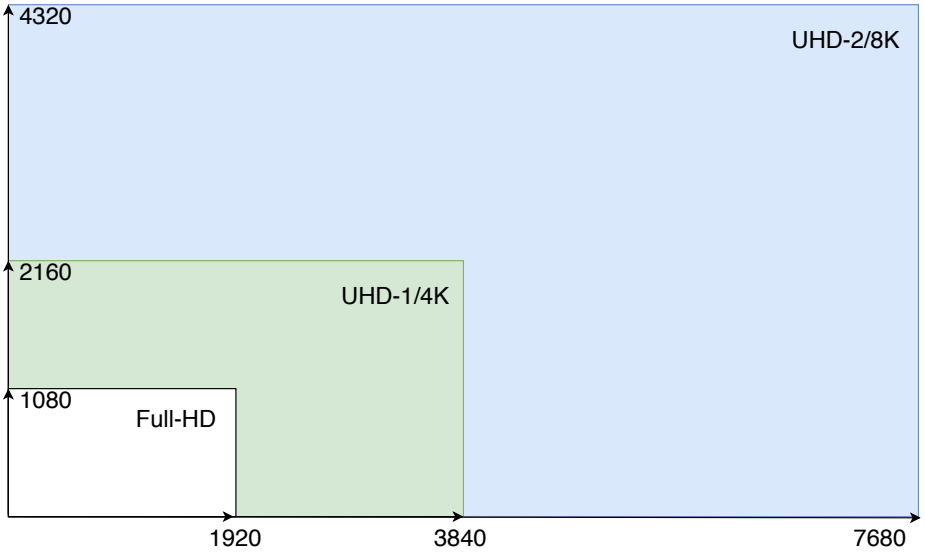


Figure 2.1: Visual comparison Full-HD, UHD-1/4K and UHD-2/8K, based on [ITU15].

Besides larger digital image resolutions, also videos can be recorded with higher resolutions and more frames per second. For example, a UHD-1/4K video has usually 60 frames per second, where each frame has a resolution of 3840 × 2160 in case of UHD-1, or according to 4K a resolution of 4000 × 2160 pixels [Uni12; ITU15]. In the following, a UHD-1/4K video refers to a video with 3840 × 2160 pixels and at least 50 frames per second. Therefore, a UHD-1/4K video has about 4 times more pixels than a Full-HD (1920 × 1080) video, with about double the number of frames per second. Even more pixels per frame are handled in the specification of UHD-2/8K, in this case, videos have a resolution of 7680 × 4320 with framerates up to 120

frames per second [Uni12; ITU15]. A visual comparison of Full-HD, UHD-1/4K, and UHD-2/8K is shown in Figure 2.1. Assuming 60 frames per second with a chroma sub-sampling of 4:2:2 with 10 bits as bit-depth, a 10 second video in UHD-1/4K uncompressed would result in a file with the size of approximately 12 GB. Such an uncompressed video has roughly the size of the compressed text-only data dump of Wikipedia [Wik20]. Besides the increase of resolution for UHD videos, which furthermore also leads to more processing time for any pixel-based analysis, other extensions are also considered for Ultra High Definition content, for example, wide color gamut, higher bit-depth, high dynamic range, and high framerates [Eri15].

Furthermore, the question arises whether UHD videos are required, or how users perceive these compared to traditional Full-HD content, for example, addressed in [Gör+19; Ber+15], or considering UHD and HD contents including resolution-related labels with matching and non-matching indications in [Kar+19; Kar+17]. Here, typical variables for quality perception are the upscaling and frame interpolation algorithms, screen sizes, viewing distances, user's vision and expectations, and more.

In the following, UHD-1/4K is assumed to be the main application case of models and quality prediction within this thesis. However, all models developed in this work can be extended to handle UHD-2/8K or more, especially when training data with subjective quality annotations and uncompressed UHD-2/8K video sources are publicly available. Moreover, it should be mentioned that considering the transmission of such high-resolution video signals, also additional requirements for *DASH*-servers, the end devices, and algorithms are required.

2.2 Machine Learning Basic Principles for QoE

Traditionally visual quality, which forms a part of *QoE*, is assessed using a discrete or continuous rating scheme [ITU06; Rec08; ITU14b; MR14], where ratings are gathered in subjective tests with several participants to enable statistical stability of the results. For example, in the case of video quality, a simple setup could be the following, a participant who attends a perception test will see a visual stimulus presented on a display and afterwards answers a question regarding the overall perceived

visual quality. This directly leads to a mapping for each stimulus in such a test to a user-specific rating value, that is either discrete or continuous. However, when a more stable view on the rating is required, usually a mean value, *MOS*, is calculated based on several ratings collected from different participants. Here, the *MOS* is a continuous value in a defined range, depending on the used rating scheme.

Moreover, for this reason, *QoE* can be formulated as a traditional machine learning problem, where for a given stimulus, for example, a video, a discrete or continuous value q is required to be predicted by an unknown function. The needed value q is therefor a function $q(\ldots) = f(\ldots) + \epsilon = f(\vec{x}) + \epsilon$ of several possible input parameters $\vec{x}$ (in the following simplified to x), that are for example properties of the stimulus, external conditions, user's preference or more. Such input parameters, in the following referred to as features, are either explicitly given or derived from the input and are a characterization of the quality distortion or factors of this stimulus.

The prediction of discrete or continuous values is a typical supervised or unsupervised classification or regression problem, where a machine learning algorithm will approximate the unknown function f. Due to the approximative nature of the underlying machine learning algorithms and also because of noise and inconsistencies in the training and or validation data, it clearly can be stated that the value q cannot be estimated perfectly using f, leading to an error ϵ in such predictions In the context of visual quality prediction VQ is used for q.

In general, the unsupervised case of machine learning methods is not so relevant for direct prediction of *QoE*, because usually ground truth data can be created by conducting subjective tests, so that supervised learning can be applied. In general, such tests are also required to reflect human perception because this is not directly given in an unsupervised manner for machine learning algorithms. However, unsupervised machine learning algorithms can be used to define features or in a semi-labeled training approach.

2.2.1 Parametric Models

As mentioned before, the main goal of a quality model is to estimate the unknown function f based on the given input, while maintaining a low overall error ϵ. Therefore, such a model can be just an approximation of the quality value q.

Some *QoE* models are based on parametric functions, where, e.g., the parameters of a polynomial, exponential or logarithmic function are optimized using curve fitting methods, with various optimization possibilities. One possible optimization method could be, for example, the least-squares method or the Levenberg-Marquardt algorithm [Lev44].

For instance, in [FHT10; Rei+10], the link between *QoS* parameters and *QoE* is modeled as an exponential function, thus $f \sim e^x$, which is referred as "IQX hypothesis" in the literature. The input x are technical *QoS* properties of the underlying system or service, and it is shown that this kind of relationship is suitable for several *QoE* research questions, e.g. web surfing [FHT10]. Moreover, in [Cha+18], a multidimensional extension of the IQX hypothesis is presented, where x is a vector of several *QoS* indicators. Another exponential or logarithmic approach to model *QoE* in the case of video is shown in [KNK12]. The origin of the exponential or logarithmic approaches to estimate human perception is the Weber-Fechner Law [Fec48].

The ITU P.1203 [ITU17] (short term audio and video quality) and P.1204.5 [ITU19d] are other examples for parametric video quality models. One of the models that is part of ITU P.1203 [ITU17], the meta-data model (mode 0), estimates video quality on a segment level with an exponential function ($f \sim e^x$), where x depends on several pre-processing steps of the meta-data. Similarly, P.1204.5 [ITU19d] uses a sigmoid-like exponential function as the final aggregation step.

2.2.2 Support Vector Machine

VMAF [Net], brisque [MSB13], niqe [MMB12] are just some of several examples of video/image quality models that rely on specifically designed features, that are later used in combination with a *Support Vector Machine (SVM)* or *Support Vector Regressor (SVR)* to predict quality scores. In general, a *SVM* estimates a hyperplane

in a given feature space to separate labeled data-points [SV08]. For the regression case, this basic concept is extended. Here, the hyperplane is estimated so that the training data is within a specific tolerance level around the hyperplane [SV08].

2.2.3 Decision Tree based Algorithms

Similar to the *SVM* approach, decision tree-based algorithms are used for image and video quality prediction, e.g. in [ITU17] (overall quality) or [Ul +20]. In general, a decision tree can be seen as a collection of several rules, each path from a leaf to the root node within the tree corresponds to one rule, and a rule includes different conditions [MRT18]. For a decision tree $f \sim tree$, where *tree* is a nested combination of if-then-else statements. The rules follow a hierarchical structure and such trees can be trained using various algorithms, e.g. the ID 3 algorithm [Qui86].

Random forest models (RF) [Bre01] are based on decision trees and extended to overcome a too specific fitting of the generated tree to the training data. In the case of *RF*, several decision trees are trained [Bre01] using random sub-sampled parts of the input data also known as bootstrapping [TE93]. All trained decision trees form a forest, and later the prediction is performed for all trees. Because each tree will produce an individual prediction, that is not required to be matching the other trees, either majority voting, mean aggregation, or other methods are used to estimate the final predicted value. This also depends whether the *RF* is used for regression or classification. The ensemble of trees model f as $f \sim agg(n \times tree)$ similar to the decision tree, with *agg* being any kind of final aggregation method. In general, the individual trees of an RF model can be trained independently, thus the training can be performed in parallel, which leads to an overall faster training process compared to other regression models.

2.2.4 Deep Neural Networks

Recently, many prediction problems in computer vision and image processing are using *deep neural network (DNN)* based methods. For example, *DNN*s can be used for image classification [Rus+15], image appeal prediction [Lu+15], quality prediction [TM18], and more. Thus, also video and image quality models are taking

advantage of such neural networks [GR18; GSR18; DWM17; Bos+16b]. Especially for quality prediction, usually *convolutional neural networks (CNNs)* are used. Most of the published models apply transfer learning. Transfer learning is a process where an already trained network is re-trained to a new task, for example, a given image classification *DNN* is re-trained for the quality prediction task. In general, such *DNN*s contain several layers [GBC16], e.g., convolutional layers, fully-connected layers, drop-out layers, ..., and in case of transfer learning, only a subset of the layers (what could be the last layer only) is re-trained [Cho; TS10]. In a simplified view, the *DNN* outcome for f can be seen as $f \sim agg(nestedlayers)$, where agg is the final aggregation, e.g., a sigmoid activation function, and $nestedlayers$ are the different layers of the *DNN* that are connected according to the definition of the network. However, even though *DNN*s show promising results for quality prediction, higher resolutions and higher framerates lead to more processing time and may lead to the non-applicability of *DNN*s in these cases, depending on the available processing resources and requirements. One major advantage of *DNN*s is that they do not need hand-crafted features, whereas in turn more data for training and time is required. Especially the quality and amount of the data is important, considering that the *DNN*s require to have a wide range of possible input images or videos for training.

2.2.5 Other Machine Learning Methods

Besides the aforementioned machine learning approaches to model video or image quality as regression or classification tasks, also other methods can be used, for example to reduce feature dimensions or to cluster input data. Dimension reduction approaches, like PCA, T-SNE [LH08], autoencoders [Kra91], can be used to reduce the number of given feature values, based on hidden dependencies within the data. For example, PCA assumes a linear connection between the given input data, while T-SNE is an extension that can also handle non-linear dependencies. Moreover, autoencoders are *DNN*s consisting of an encoder and decoder with a bottleneck as connection. This bottleneck reflects the dimension reduction. In addition to dimension reduction, data clustering can help to balance datasets, or to filter out noise or outliers within the data. Examples for such methods are K-Means [Mac+67] or DBSCAN [Est+96]. Besides data clustering and data reduction, in several cases,

some features are just not required to be used for the final model prediction. A reduction of features can help to make a given model simpler, faster, and to be less over-fitted to specific datasets. In general, feature selection [BLR14, p. 3] can be realized using a machine learning model, e.g., a decision tree-based model as a first step within the model pipeline. The feature selection model will be trained on the input data, and based on the resulting used features a reduction can be performed. For example, the overall used features can be reduced using statistics about the usage of features within the feature-selection model (feature scoring) and a given threshold.

2.2.6 Beyond Mean Opinion Score Prediction

Besides the well-known and used approach of modeling visual quality based on mean opinion scores that originate from perception tests, noted as $VQ_{mos}(v) \mapsto float$ for the given stimulus v, also other approaches should be considered.

In the following, it is assumed, that all individual ratings for a given stimulus v are accessible and are in the range $[1, 5]$, thus using the 5-point *Absolute Category Rating (ACR)* scale. However, it should be mentioned that not in all cases individual ratings are available, for example some publicly available datasets only publish *MOS* values, or *MOS* and *confidence interval (CI)* values.

Based on majority or rounded mean or median ratings per stimulus v the given video quality prediction problem can be modeled as a classification task, noted as $VQ_{class}(v) \mapsto int$.

The VQ_{class} variant of video quality prediction is a simpler version of VQ_{mos} considering only discrete values. It still can be applied in cases where users' acceptance is required or less granular quality monitoring is appropriate. For example, if a faster model with lower accuracy is used, the classification view can be the first indicator of whether quality drops or other technical problems occurred in a streaming provider scenario.

Another possibility is to model the video quality prediction task as a 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 here, 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 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. All ratings for a given video v are defined as $ratings(v)$, see Equation 2.1.

$$ratings(v) = [rating(v, u_1), \ldots, rating(v, u_n)], \tag{2.1}$$

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 2.2.

$$prob(v) = [(r, |rating(v, u_i) = r|/n) \forall i \in [1..n] \wedge r \in [1..5]] \tag{2.2}$$

If only a specific rating should be analyzed, the notation in Equation 2.3 is used.

$$prob_{=r}(v) = |rating(v, u_i) = r|/n \forall i \in [1..n] \tag{2.3}$$

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

Here, the focus is to predict 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 $prob_{=1}(v) = prob_{=4}(v) = 0$, and also $prob_{=2}(v) = prob_{=3}(v) = prob_{=5}(v) = 1/3$.

All these probability values can be used as video quality prediction targets, in the following referred to as $VQ_{prop}(v)$ which is defined as shown in Equation 2.4.

$$VQ_{prop}(v) \mapsto [prob_{=1}(v), prob_{=2}(v), prob_{=3}(v), prob_{=4}(v), prob_{=5}(v)] \tag{2.4}$$

The 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 $r = 1..5$, leading to the formulation of the video quality prediction task as a multi-output regression problem.

2.3 Review of Quality Models for Images and Videos

In the next part, several image or video quality models, that are targeting the VQ_{mos} quality prediction problem are briefly described. Wherever, depending on the input data, several model types can be differentiated, an overview of such model variants will be presented. The focus will be on video quality models, whereas some of the mentioned models originate from image quality. A naive approach to use an image quality metric for video quality estimation is to apply the model to each frame and calculate the overall video quality based on all per-frame quality values, e.g. using the mean value of all quality scores. Also, the opposite way is possible, e.g., in case that a video quality model predicts per frame quality scores, a video containing one single image could be created to estimate the image quality using such a video quality model.

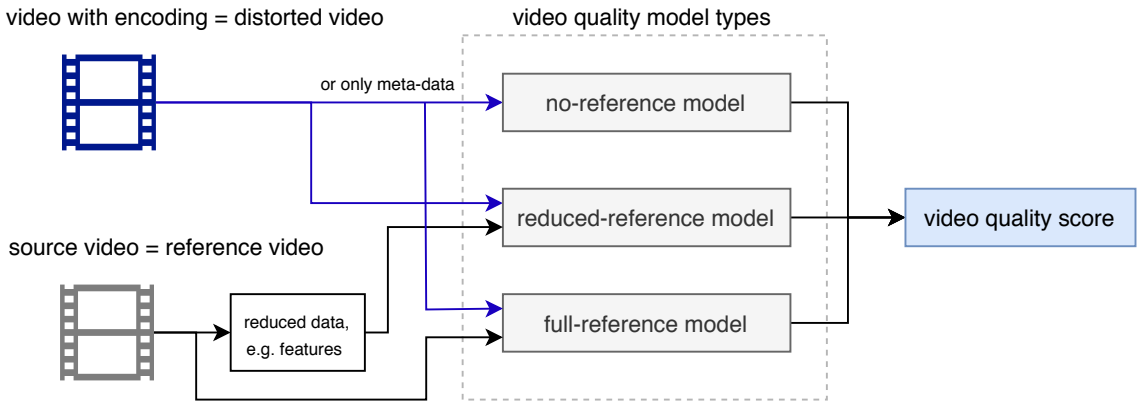


Figure 2.2: Overview of different types of video quality models, similar models can also be distinguished for images. Depending on the input (source video and distorted video, or only distorted video), several model variants can be developed and classified, e.g., full-, reduced- and no-reference.

In general, a number of video or image quality models exist and they can be typically categorized into three main types, namely full-, reduced- and no-reference models [Sha+14; BM19] depending on the input data that is available for quality estimation. In Figure 2.2 an overview of the different video quality model types is shown with their respective input data, in an analogous way image models can be differentiated. For example, a *full-reference (FR)* quality model needs access to the reference and distorted video, thus the overall quality estimation uses both signals.

On the other side, a *no-reference (NR)* model only requires to process the distorted video, this can start with an analysis of the given pixel information, or may be based on undecoded frame information using bitstream- or meta-data. In the 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. Besides no- and full-reference models, there are also *reduced-reference (RR)* models following a two step process. Here, in the first iteration, a reduced representation of the source video is created. Such a representation could be, for example, specific calculated per-frame features or other general characteristics of the source video. In the later analysis of the distorted videos, only the reduced source representation is accessible, that's why reduced-reference models can be seen as a trade-off between no- and full-reference models. Besides the three mentioned model types, there are also other models possible, e.g., a no-reference pixel-based model that has in addition access to meta-data would be referred to as hybrid no-reference model (mode 0, according to the used terminology in ITU-T P.1203 [ITU17] and P.1204 [ITU19a; Raa+20]), whereas in general all model types can be extended by bitstream-data, leading to different hybrid model types. The corresponding mode of such hybrid models depends on the level of access to the bitstream of the distorted video. Here, a typical mode 0 model would use only framerate, resolution, bitrate, and video codec as meta-data. Mode 1 would have, in addition to mode 0, access to frame-sizes and types, and finally, a mode 3 type model can access the full bitstream-data. Also a mode 2 type bitstream model is possible, where such a model only uses 2% of the mode 3 data and has less practical applications, especially considering today's encrypted streams. For this reason, it was also dropped in the ITU-T P.1204 series.

In general, two different aspects can be distinguished for *DASH*-type video quality estimation. First, how the per segment video quality is estimated, which is usually referred to as the short-term video quality. Second, how the overall audiovisual/video quality after a longer time including stalling, audio quality, and more is estimated, referred to as long-term video quality. For example, ITU-T P.1203 [ITU17] handles both cases in an integrated framework, where overall audiovisual quality can be estimated for up to 5 minutes of video duration.

Moreover, recently the ITU-T P.1204 [ITU19a] standard has been approved. Models from this standard series consider short-term video quality including H.264, H.265,

and VP9 encoded videos up to UHD-1/4K resolution. Raake et al. [Raa+20] 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.

Considering the variety of different *DASH* streaming parameters, video quality depends on several factors, starting from various used video codecs, differently optimized encoding settings, and corresponding bitrate-ladders, various recording settings and more parts in the typical end-to-end video streaming chain (compare Figure 1.4 in Section 1.3). The existing set of models are far from comprehensive as yet. For example, Barman and Martini [BM19] identified several open points that are not handled fully by such models, e.g. privacy, high model complexity, multiple influence factors on video quality perception, and others. 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, different example models will shortly be outlined.

2.3.1 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 or reduced-reference models do, because they cannot compensate for 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 been standardized by ITU-T SG12 or the Video Quality Experts Group (VQEG[1]) to date. In the following, some relevant examples of *NR*-type models will be presented.

2.3.1.1 Bitstream-based Models

As already introduced, ITU-T P.1203 [Rob+18a; Raa+17; ITU17], is a bitstream-based no-reference video quality model developed especially for adaptive streaming use

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

cases. The model is trained on Full-HD videos including audio 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. [Rao+19b] propose a method to extend P.1203 to modern codecs for the metadata-based Mode 0, namely to AV1, H.265, and VP9. Besides the 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. With a similar goal, Yamagishi et al. [Yam+21] propose a method to use *FR* models for the re-training of mode 0 model variants with promising results.

To cover more video codecs, higher resolutions, and framerates, the models from the newly standardized ITU-T P.1204 [ITU19a; Raa+20] 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 [ITU19b], with full access to the video bitstream. P.1204.3 uses several statistics that are extracted from the video bitstream [Rao+20b; ITU19a]. For example, statistics about motion vectors, quantization parameters, and frame sizes are extracted for the video codecs 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 [ITU17; Raa+17], whereas the machine learning part uses random forest regression with feature selection to predict the residual not captured by the first part of the model. Rao et al. [Rao+20b] use the AVT-VQDB-UHD-1 dataset [Rao+19a] to perform an additional analysis of the model performance, with an implementation of the model being made publicly available as Open Source.

2.3.1.2 Natural Scene Statistics based Models

Besides 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[2]. In scikit-video, only the feature extraction of **brisque** and **niqe** are included, the final model is usually a *SVM* or *SVR* [MSB13; MMB12] which uses the extracted features as input.

Both methods are independent of distortion-specific assumptions and focus on measuring differences in the naturalness of the given input image. This is realized using statistics of normalized luminance coefficients to measure the differences to the ideal coefficients of 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, for example, shown in [GR18] for images and in [GSR18] for 4K videos. Even for streaming quality of gaming videos or sessions, these models can be applied and show promising results [Bar+18a; GRR19; Bar+19]. However, to apply them for this type of 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 models. 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 **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 [GSR18].

Another natural scene-statistics-based model is BIQI [MB10]. 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 [SSB06] dataset.

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

2.3.1.3 DNN-based Models

Next to classical signal driven video quality models, models based on deep learning can also be used to estimate video or image quality or encoding optimization [KCR18; Kua+19; Liu+20]. Most DNN-based quality assessment models share similar approaches. For example, VeNICE [DWM17], the models of Bosse et al. [Bos+17; Bos+16b], Deviq/Deimeq [GR18; GSR18], or Wiedemann et al. [Wie+18] 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 [DWM17], the VGG16 [SZ14] network is used, similar to Bosse et al. [Bos+17; Bos+16b], whose DNN-based quality model also operates with the VGG network. The model Deviq/Deimeq [GR18; GSR18] uses Xception [Cho17] or Inception [Sze+16]. Usually, these pre-trained *DNN*s 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 *DNN*s are used as features and are combined or retrained to predict image quality. It was already analyzed which *DNN*s are more suitable for image quality evaluation [GR18]. However, especially for high-resolution videos or images, DNN-based processing is time-consuming, and also retraining is not a straightforward 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 frame can be assumed. This is shown for example in [Wie+18], indicating that quality scores for local patches can be used to estimate global image quality, however, the overall aggregation of local patches to global quality is not simple.

One no-reference model for video quality is Deviq [GSR18], which handles the mentioned high-resolution problem using hierarchical sub-images to reduce the overall number of patches. The Deviq model is explained in more detail in Section 3.4.3.

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. more task-optimized models like Mobilenetv2 [San+18] or VGG16 [SZ14] 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 [GR18].

Today, *DNN*s 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 the proposed models in this thesis the focus is 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. A common solution is that image quality models are developed in the first step and later are applied similarly to video quality prediction. However, it is mostly not fully covered how in such a case motion-related effects change video quality perception. On the other hand, subjective tests and models based on continuously rated quality scores have been proposed [Bam+18; Wei+14; HR99], e.g., 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 influenced of previously shown frames. Moreover, besides not being 'memory-less', 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 [Wei+14]. For no-reference video quality models, there are another possibilities to get ground truth data on a per-frame level. A feature based approach could be used, as it is the case for the P.1204.3 model [ITU19b; Rao+20b]. Or for example, per frame scores can be estimated using a suitable full-reference video quality metric, e.g., using VMAF as shown in [Bar+19; GSR18]. A drawback of this approach is that the scores are based on a different models, and thus the overall performance of the new model depends on the ability of the used full-reference model to measure quality variation over time.

2.3.1.4 Models for other Use Cases

Supplementary to 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 the case of 360° video, it was already shown that existing models like VMAF are able to perform quite well [Ord+19], if the equirectangular projection scheme is used, or that even meta-data and hybrid models can be applied [Fre+20a]. Similarly, for gaming sessions, VMAF has been reported to show good performance [Bar+18a]. 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 appropriate, 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 [Bar+19], Barman et al. use 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 an *SVR* approach. Moreover, subjective scores are 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 [Utk+20] 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.

However, it should be noted that especially gaming videos share similar properties, that are partially unique for gaming videos, e.g. computer-generated textures, different motion patterns that are specific to the game genre, or static head-up displays. Consequently, it is not clear if such models perform similarly well 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. [Rao+20a] evaluate the performance of the 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) [Zad+20b] and a dataset based on

Twitch are considered, showing promising results. It was further shown that the ITU-T P.1203.1 model can be applied to gaming videos [Zad+20b].

All Gaming-QoE models use similar or even the same underlying datasets, e.g. GamingVideoSET [Bar+18b] or KUGVD [Bar+19], where the used videos have a maximum resolution of Full-HD 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 the case of quality prediction for gaming sessions. Moreover, also models have been proposed to bridge quality prediction for traditional and gaming videos [Zad+20a], 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. In the second step, a random forest model uses the predictions of the neural network to estimate the overall quality.

2.3.2 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 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 feature 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 [Bam+17] and STRRED [SB12]. SpeedQA [Bam+17] is based on spatial efficient entropic differencing for quality

assessment and STRRED [SB12] uses spatial and temporal entropic differences. Another reduced-reference video quality model is ITU-T P.1204.4 [ITU19c] that is based on different types of edge statistics of the distorted and reference video to estimate video quality.

2.3.3 Full-reference Models

In contrast 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 metric is Peak-Signal-To-Noise-Ratio (PSNR), where a pure signal-based difference is estimated. PSNR is based on the mean squared error of the reference and distorted signal, where the final estimated q is logarithmically scaled. It is well known that PSNR does not well match human perception and video quality evaluation, both in general and especially in case of higher resolutions [Wan06; GSR18; Rao+19a].

Besides the classical PSNR, a measure that is also used as a quality metric in several applications is an extension of PSNR called the PSNR-HVS [Egi+06]. 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 [Egi+06]. However, using PSNR-HVS in the 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 [Hel+20] or for color CQM [YE13]. X-PSNR [Hel+20] is a low complexity extension of PSNR, that uses a block-wise weighting approach, and CQM [YE13] is a variant of PSNR where the overall score is a weighted sum of PSNR for luminance and chroma channels.

Most *FR* video quality models have their origin in image quality estimation, such as Structural Similarity Index Measure (SSIM) [Wan+04; WSB03] or Visual Information Fidelity (VIF) [SB06]. 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 the case of high-resolution videos, as reported in [Rao+19a; CL17].

Netflix's VMAF (Video Multimethod Assessment Fusion) [Lin+14; Net] 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 [Net18a; Net18b]. VMAF is based on two full-reference models, namely VIF [SB06] (4 scales) and ADM2/DLM [Li+11]. 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 a good prediction accuracy even for newly conducted video quality tests [GSR18; Rao+19a].

As features, VMAF extracts several image quality scores per frame, and in addition one motion feature. All per-frame values are later aggregated with an *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 Full-HD 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.

2.3.4 Hybrid Models

Along with pure bitstream- or pixel-based video quality models, combinations of models are possible, that are usually summarized as hybrid models [WM08; Bar+15]. 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 [ITU19d]),

which uses meta-data, that is 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, Kawano, and Hayashi [YKH09] proposed a model for IPTV, extending a meta-data model by content complexity, using the Spatial and Temporal Information (SI, TI) described in [Rec08].

2.4 Summary and Conclusion

In Table 2.1 a summary of most of all aforementioned image and video quality model types is given. All models can be used to estimate visual quality that is reflected in a continuous score for each video sequence, thus tackle the VQ_{mos} problem formulation for visual quality. Simple extensions could be used to handle the introduced quality problem in case of a discrete prediction value VQ_{class}, e.g., using rounding of the estimated values. None of the models can be used directly to predict VQ_{prop}, e.g., the DNN-based models or VMAF could be extended for this purpose. Moreover, most of them are not explicitly trained for higher resolution images or videos. Models with high accuracy considering human perception are also complex and require computation time.

High resolution images and videos lead to the need for quality prediction models that are not only accurate, but also fast, due to the increase in computing complexity and are capable of handling a large diverse set of contents. Moreover, it was also shown that machine learning models are widely used in the research field of *QoE* for visual quality prediction.

> For the identified visual quality prediction problem, that is finding a function f to approximate the subjective quality VQ, several formulations can be considered, for example as
>
> - VQ_{class}: classification task,
> - VQ_{mos}: regression problem or as
> - VQ_{prop}: distribution prediction problem.

Various machine learning models, that can be used to tackle the identified problem type, are applicable. The important models have been briefly described and linked to applications of *QoE* estimation.

Image and video quality prediction can be handled with different types of models, depending on the input data that can be used for the prediction. For example, no-reference, reduced- and full-reference models are possible. Each model type has different application scopes, and not every model is suitable for all real-world applications. Also, development of newer encoders resulting in newer encoder-specific distortions necessitate the development of newer video quality models. However, only a limited number of image and video quality models can handle larger resolutions and include precise human perception handling and prediction. As a result, models optimized specifically for high resolution image and video quality prediction are described and evaluated in detail in the following Chapters.

Table 2.1: Overview of image and video quality models, the "Accuracy" column refers to human perception, "Complexity" stands for time complexity.

Name/Acronym	Type	Code	Accuracy	Complexity	Support high resolutions	Source
PSNR	FR-img	open	low	low	low	
CQM	FR-img	open	low	low	low	[YE13]
PSNR-HVS	FR-img	open	mid	low	low	[Egi+06]
SSIM	FR-img	open	mid	low	low	[Wan+04]
VIF	FR-img	open	mid	low	low	[SB06]
ADM2/DLM	FR-img	open	mid	mid	low	[Li+11]
X-PSNR	FR	open	mid	low	mid	[Hel+20]
VMAF	FR	open	high	high	high	[Net]
STRRED	RR	open	low	low	mid	[SB12]
SpeedQA	RR	open	low	low	mid	[Bam+17]
P.1204.4	RR	closed	high	mid	high	[ITU19c]
P.1204.5	Hybrid-NR	closed	high	low	high	[ITU19d]
HybridIPTV	Hybrid-NR	closed	low	mid	low	[YKH09]
VRmeta	Hybrid-NR	open	high	low	high (VR)	[Fre+20a]
DEMI	NR-gaming	closed	high	high	mid	[Zad+20a]
NDNetGaming	NR-gaming	closed	high	high	mid	[Utk+20]
NR-GVSQI/E	NR-gaming	closed	high	high	mid	[Bar+19]
Nofu-gaming	NR-gaming	closed	high	high	mid	[GRR19]
deimeq	NR-img	closed	high	high	mid	[GR18]
VeNICE	NR-img	closed	high	mid	low	[DWM17]
brisque+niqe	NR-img	open	mid	mid	low	[MSB13]
BIQI	NR-img	open	mid	mid	low	[MB10]
deviq	NR	closed	high	high	high	[GSR18]
P.1203.1	NR	open	mid	high	mid	[Raa+17]
Mode 0 ext (1)	NR	open	mid	high	mid	[Rao+19b]
Mode 0 ext (2)	NR	open	mid	high	mid	[Yam+21]
P.1204.3	NR	open	high	mid	high	[Rao+20b]
DNN	NR/FR	closed	high	mid	mid	[Bos+16b]

Chapter 3

High Resolution Image Quality Evaluation

Image compression is not a new topic, however, developments and improvements are still possible. With the increase of transmitted data, it is even more important to deliver high-quality images with high compression. For example, Whatsapp uses image compression for transmission, leading to an additional distortion and resizing of any image shared [Sha17]. Besides the communication aspect, there is also an increase in photos uploaded to sharing platforms, e.g., Instagram, because new technologies enable easy creation and uploading of photos.

Quality prediction for high-resolution images and analysis for compressed images is still an important topic due to the aforementioned reasons. Modern approaches for evaluation and prediction models will be discussed and presented in the this chapter. Alongside traditional lab-based evaluations, further crowdsourcing or online-testing approaches will be considered with adaptions to higher resolution images.

The following questions will be answered in the subsequent Chapter:

- ▷ Can video quality models be used to evaluate image quality? (**Research Question** 5)
- ▷ How can crowdsourcing tests be applied to high-resolution images?
- ▷ Is it possible to compress images using video compression technology while maintaining similar or better visual quality? (**Research Question** 5)
- ▷ Is it possible to use deep neural networks for image and video quality prediction? (**Research Question** 1)

The chapter is mostly based on the following publications:

[GR19] **Steve Göring** and Alexander Raake. "Evaluation of Intra-coding based image compression". In: *8th European Workshop on Visual Information Processing (EUVIP)*. IEEE. 2019, pp. 1–6

[GR18] **Steve Göring** and Alexander Raake. "deimeq – A Deep Neural Network Based Hybrid No-reference Image Quality Model". In: *7th European Workshop on Visual Information Processing (EUVIP)*. IEEE. Nov. 2018, pp. 1–6. DOI: 10.1109/EUVIP.2018.8611703

[GSR18] **Steve Göring**, Janto Skowronek, and Alexander Raake. "DeViQ – A deep no reference video quality model". In: *Electronic Imaging, Human Vision Electronic Imaging* 2018.14 (2018), pp. 1–6

3.1 Video Compression for High Resolution Images

Considering that newer image codecs and methods have been developed for higher resolution images, e.g., AVIF [Ope19] or HEIF [Lai+16; Nok20], there is a need to assess the effect of these on the quality of such high-resolution images. A limiting factor are datasets, because most published datasets either use lower resolutions or include only JPEG images, for example, the Tampere Image Dataset 2013 [Pon+15]. As it is shown in [GR19] video compression methods applied to image compression can outperform classical state-of-the-art image compression, e.g., compared to JPEG. More details for this evaluation are presented later in this chapter.

Recent developments, such as, the JPEG-AI competition[1], focus on image compression methods that are learning-based and suitable for higher resolution images. Those learning-based methods can be implemented using *DNN*s [Che+20; Zou+20; Lin+20] or use hybrid approaches that rely on traditional methods combined with neural networks for image enhancement [Lee+20]. An example for a hybrid variant is proposed by Lee et al. [Lee+20]. Lee et al. [Lee+20] use VVC [ITU20], a recently published video encoder, to compress images and later perform image enhancement using a neural network.

[1] https://jpegai.github.io/

On the other side, newer and promising image compression methods like JPEG-2000 or JPEG-XR are not widely used in practice. In addition, current extensions of camera devices to record videos enable the usage of video compression methods for still image compression, especially because hardware acceleration can be used. Here, in the case of recording several photos in a series, video compression methods can store such highly redundant data more efficiently in contrast to JPEG or other still image-based methods. The reason for this is that such a series can be assumed to be a video, and motion estimation of video codecs will enable a larger compression efficiency. In general, lossy image compression algorithms enable the possibility of storage reduction with minimal quality changes. The current most popular web image formats are JPEG, GIF, and PNG [Dia18]. Newer formats are WebP [Goo20], BPG [Bel18], HEIF [Lai+16; Nok20] or AVIF [Ope19]. All four new formats have in common that they are based on video compression algorithms, e.g., WebP is based on VP8, BPG is based on a modified HEVC variant, HEIF uses HEVC, and AVIF is based on AV1. The trend to use video coding approaches for images leads to the question of whether they can outperform traditional methods in compression efficiency and quality.

3.2 Objective Evaluation for Image Compression using Video Encoders

As mentioned before, modern image compression methods also use the development of video encoders to improve quality while targeting a higher compression efficiency. However, only a few published studies are comparing newer developed methods. For example, in [Lai+16], it is shown that HEVC/H.265 is able to save bitrate while keeping the same quality in comparison with JPEG, the evaluation was performed using 14 high-resolution images (height/width of maximum 4064 pixels). Moreover, other studies confirm that HEVC's intra-frame coding is a well suitable still image compression approach [NM14]. Besides HEVC, VP9, and VP8, AV1 is another promising video codec, however, there are only a few studies available comparing still image compression of AV1 or the AVIF format [BM20]. In [Egg+15], the still image compression performance of the Daala video codec is analyzed. The evaluation

of Daala's compression ability is performed using 8 images up to Full-HD resolution. However, the Daala codec development is mostly subsumed in AV1. Barman and Martini [BM20] compares JPEG, WebP, JPEG-2000, HEVC and AVIF using objective quality metrics such as VMAF, SSIM, VIF and PSNR. The evaluation is performed using three different datasets, consisting of images with a resolution of 2040×1346 and 1920×1080 thus Full-HD and 2K resolutions. Based on the results it can be concluded that AVIF outperforms other methods considering the quality and bitrate savings. However, it should be mentioned that there are no high-resolution images (higher than 2K) included in the evaluation. For this reason, another evaluation is required.

3.2.1 Approach for Image Compression with Video Encoders

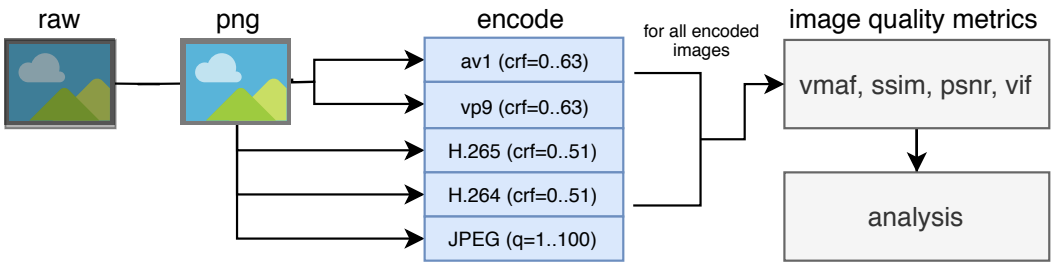


Figure 3.1: General overview of the image compression and processing pipeline.

The general image processing and compression framework is shown Figure 3.1. The overall process starts with an uncompressed 'raw' camera image. At first, this image is converted to a lossless compressed PNG image. As next, the PNG image is converted to a video consisting of one frame using the lossless video codec *ffvhuff* with chroma sub-sampling of 4:2:0 and 8 bits. This specific sub-sampling and bit depth has been selected because WebP also used the same settings. For all processing steps, the following software components are used: a static build of ffmpeg[2] with version 4.1.3, the convert tool that is part of ImageMagick[3] version 6.9.10-14 and darktable[4] version 2.4.2. The generated lossless video consisting of one frame is later encoded using VP9, AV1, H.265, and H.264. For H.264 and H.265, the encoding preset

[2] https://ffmpeg.org/
[3] https://imagemagick.org/index.php
[4] https://www.darktable.org/

is veryslow, similarly, the "cpu-count=1" parameter of AV1 and VP9 are configured, which ensures that similar encoding effort is spent in encoding.

AV1/VP9 and H.264/H.265 have different quality influencing parameters, using pre-tests similar settings considering quality have been selected. For each video codec, a one-pass constant rate/quality (crf) encoding of the input image with various crf values is implemented, which results in several distorted versions.

Similarly, JPEG is handled, where the images are encoded at all possible quality settings of JPEG (ranging from 1 to 100). After encoding a given image, the VMAF[5] tool is used to estimate several objective quality metrics, i.e., VMAF, VIF, SSIM, and PSNR. To use VMAF for the JPEG case the distorted JPEG image is converted to a video consisting of one frame without quality change during conversion.

To analyze all different codecs in a unified way, a similar scaled quality level setting is required. Based on crf encoding (ranging from 0 to n_{codec}) such a quality level setting can be defined as shown in Equation 3.1

$$\text{quality-level} = ql = 1 - crf/n_{codec} \tag{3.1}$$

The n_{codec} value is normalized according to the used video codec, where $n_{\text{AV1}} = n_{\text{VP9}} = 63$ and $n_{\text{H.264}} = n_{\text{H.265}} = 51$. In the case of JPEG compression, as quality level, the normalized value shown in Equation 3.2 is used.

$$ql = (\text{quality-setting} - 1)/99 \tag{3.2}$$

The measure is based on the used quality-setting parameter for JPEG, which ranges from 1 (lowest) to 100 (highest quality). The normalization and unification ensure a codec independent quality-level ql ranging from 0 to 1 where 1 means least distortion introduced and 0 highest distortions.

For a given PNG image the presented pipeline creates $2 \cdot 64 + 2 \cdot 52 + 100 = 336$ different image representations and several corresponding image quality metrics.

To perform a large-scale evaluation with a focus on high-resolution images, a suitable dataset is required. Usual image quality databases as for example the Tampere Image

[5] https://github.com/Netflix/vmaf

Database 2013 [Pon+15] or LIVE-2 Dataset [She+16] are not applicable, because such databases consist of only a small number of images that are of relatively low-resolution. Also, on the other side, there are larger image databases available, e.g. KonIQ-10K [Hos+20] or LIVE In the Wild Image Quality Challenge Database [GB16], however, they are based on lossy compressed source images, that would introduce compression already in the beginning of the process.

For these reasons, a dataset is created, where the compression pipeline is applied with various settings. As next, a brief description of the dataset is outlined. The dataset consists of 1133 source images. In total, using the provided pipeline, $1133 \cdot 336 = 380,688 \approx 380k$ different versions of quality degraded images are created. Further, for each of the images, various quality metrics are calculated.

3.2.2 Overview of the used Dataset

Figure 3.2: Ten randomly selected example images from the collected raw image database.

The dataset (AVT-Image-Database) consists of 1133 unique images from the image sharing platform Wesaturate[6], all downloaded images are CC-0 licensed[7]. Wesaturate's idea was that users can share their raw images with the community, e.g., for raw image processing. As a first step after downloading all images, all the raw images are converted to lossless PNG format because they were stored using several proprietary camera-specific image formats. All following steps were done using the PNG version of each image.

[6] see https://web.archive.org/web/20170603213404/https://wesaturate.com/; the page is not existing anymore

[7] code and full dataset: https://github.com/Telecommunication-Telemedia-Assessment/image_compression

In Figure 3.2 ten example images are shown to present an overview of the raw image dataset. For all images, the mean height/width ranges from approximately 3980 to 4375 pixels, maximum height/width is 10368 pixels. The dataset also consists of about 20 low resolution (height/width=526 pixel) images, that were excluded. In addition, the descriptive tags that are used as annotation for the images are analyzed, on average each image has about 3.76 tags assigned.

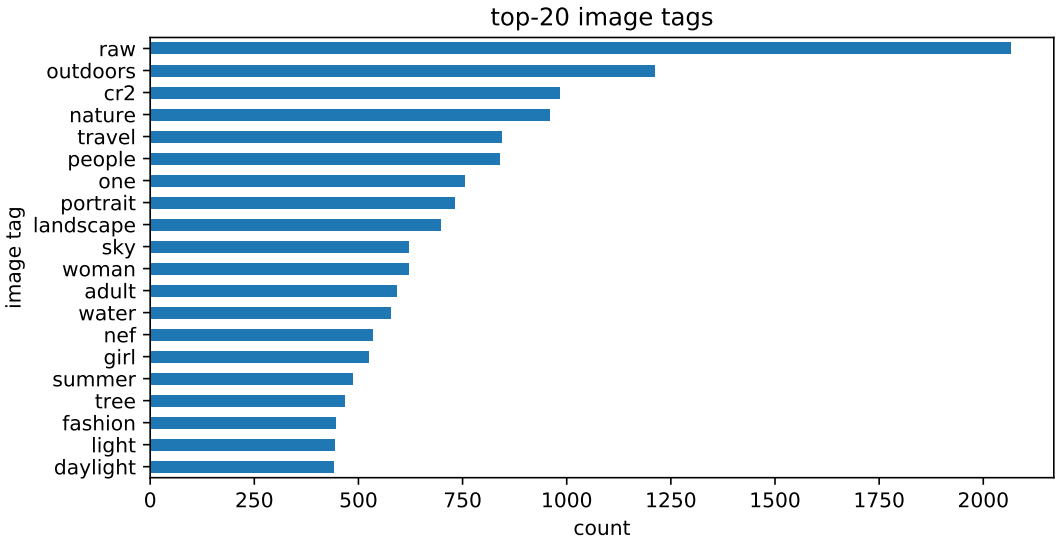
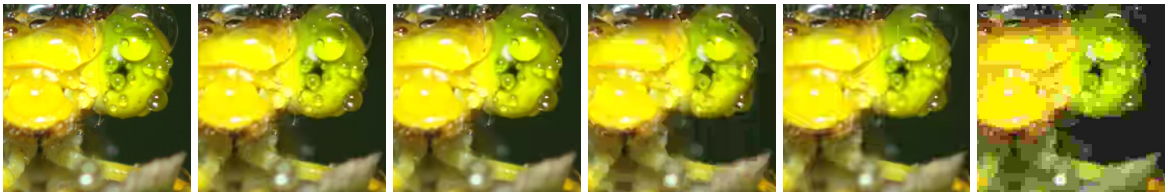


Figure 3.3: Top-20 used image tags, one image can have several image tags, on average approximately 3.76 tags per image.

Figure 3.3 depicts the top-20 used image tags of the dataset. Most used image tags are `raw`, `cr2`, `outdoor` and `nature`, obviously all images have the `raw` tag. There are other tags that are often used, e.g., light, day light, summer, water, however some of them probably belong to similar photo subjects. Moreover, Figure 3.3 demonstrates that the database has a broad range of photo subjects covered.

3.2.3 Visual Quality Comparison

As a first step, a visual exploration may be helpful to get a general impression of the introduced compression approach. For this 360p center crops were extracted for one image with the lowest quality level setting reflecting the highest visual loss. In Figure 3.4 for each codec the extracted crop is shown. It is visible that, e.g., VP9 and AV1 are quite similar, also H.264 and H.265 are similar. However, all video codec



(a) Source (83 MB) **(b)** VP9 (15 KB) PSNR=45.36 **(c)** AV1 (11 KB) PSNR=45.87 **(d)** H.264 (19 KB) PSNR=39.74 **(e)** H.265 (17 KB) PSNR=42.72 **(f)** JPEG (110 KB) PSNR=32.26

Figure 3.4: 360p center crop of an example image with the highest compression (quality-level=0) for all considered methods, in brackets file-size of the full image and PSNR values to reference image without a crop.

compressed images are visually better than JPEG, where clear block artifacts are visible. Besides block artifacts, also some color loss is recognizable. SSIM and PSNR values lead to the same conclusion. In addition to the images, file sizes are provided to enable a first check on compression efficiency. It is notable that the largest file is the JPEG file, where AV1 has the smallest, followed by VP9. Still, H.264 and H.265 show a smaller file size than JPEG, about a sixth of it, with clearly better visual quality.

Based on this example it is not in general clear if the observations are valid for other images, for this reason, a larger evaluation is required.

3.2.4 Compression Level compared with Quality

The focus of the following analysis is what quality can be reached for a specific quality-level setting during encoding of the given images. In Figure 3.5 results for VMAF, VIF, SSIM, and PSNR for each considered codec are shown. Mean quality values for each quality level across all images of the database were calculated. First, it can be observed that there is no linear behavior for all analyzed image compression approaches for quality and quality-level settings. Consequently, users should consider this interconnection in their choices for compression. In general, it is clearly visible that for a given quality level JPEG is outperformed mostly by all video codecs. For example, only in the VMAF plot, H.265 and H.264 show worse quality for low quality-level settings in comparison with JPEG. However, on the other side, for the higher quality-level range JPEG is again outperformed by the intra-frame compression of all analyzed video codecs. The image quality metric VIF shows

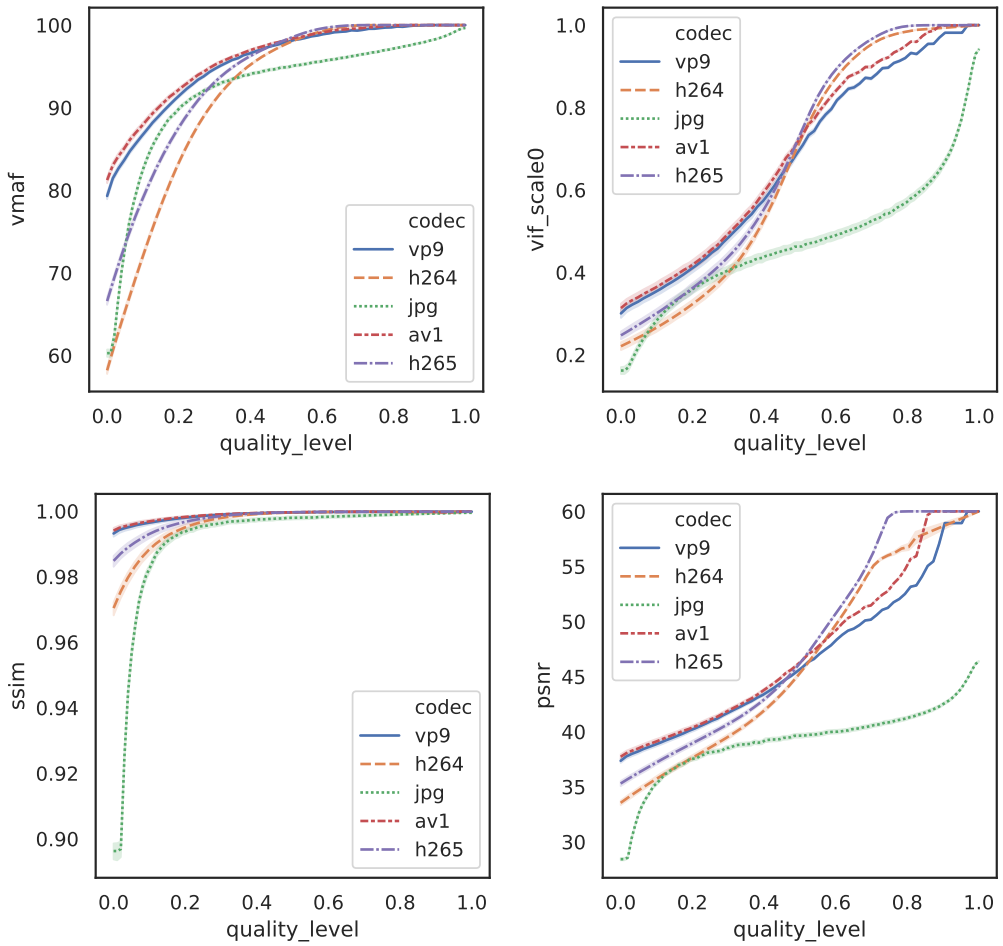


Figure 3.5: Comparison of estimated image quality with quality level settings, for each quality-level setting mean values of the corresponding quality metric are shown, also 95%-confidence intervals were calculated.

a similar behavior as VMAF, even though H.264 and H.265 performance is only worse than JPEG in a quite small quality-level range. For the SSIM and PSNR metric, all video approaches outperform JPEG. In general, intra-frame based compression shows better quality for similar quality-level settings, especially for the high-quality range where they reach higher quality values than JPEG.

3.2.5 File Size compared with Quality

Usually, the quality-level selection also affects the resulting file size of the compressed image. Hence, the final file size of the compressed images is investigated. For this a compression-ratio *cr* is calculated, as shown in Equation 3.3.

$$\text{compression-ratio} = cr = filesize(I)/filesize(R) \tag{3.3}$$

For this calculation, I is the distorted image, and R is the highest quality-level equivalent version of the image I. Thus R is always the corresponding lowest compressed version of the image. For example, in case of video codecs crf=0 settings, in case of JPEG compression quality-setting= 100. In general, the *cr* value is [0,1]-normalized, and a low *cr* reflects a small file size, thus a high compression independent of perceived quality.

Figure 3.6 shows the comparison of compression-ratio with image quality for all codecs and metrics. The mean values of quality scores for all images for a given and rounded compression ratio are calculated. First, SSIM shows no clear change for all codecs, the differences are minimal. On the other side in the case of PSNR, it is visible that JPEG is always outperformed by the other methods. Moreover, VIF and VMAF show that for higher compression-ratio values, meaning for lower overall compression, JPEG's performance is worse than the other metrics.

It can be concluded that for high quality, where users accept higher file sizes, video based compression methods are able to outperform JPEG. In addition, AV1 and VP9 are mostly always equal or better than JPEG, thus it follows that images can get a higher quality for less storage. Further, the VIF and VMAF plots show, that H.264 is worse than JPEG, especially in the low compression-ratio part, so it should not be used for image storage if a low file size and high quality is the requirement.

3.3 Quality Evaluation for High Resolution Images

Most of the image compression benchmarks or comparisons are based on PSNR [AMK15] or other objective metrics, while it is already shown that there is only a medium or

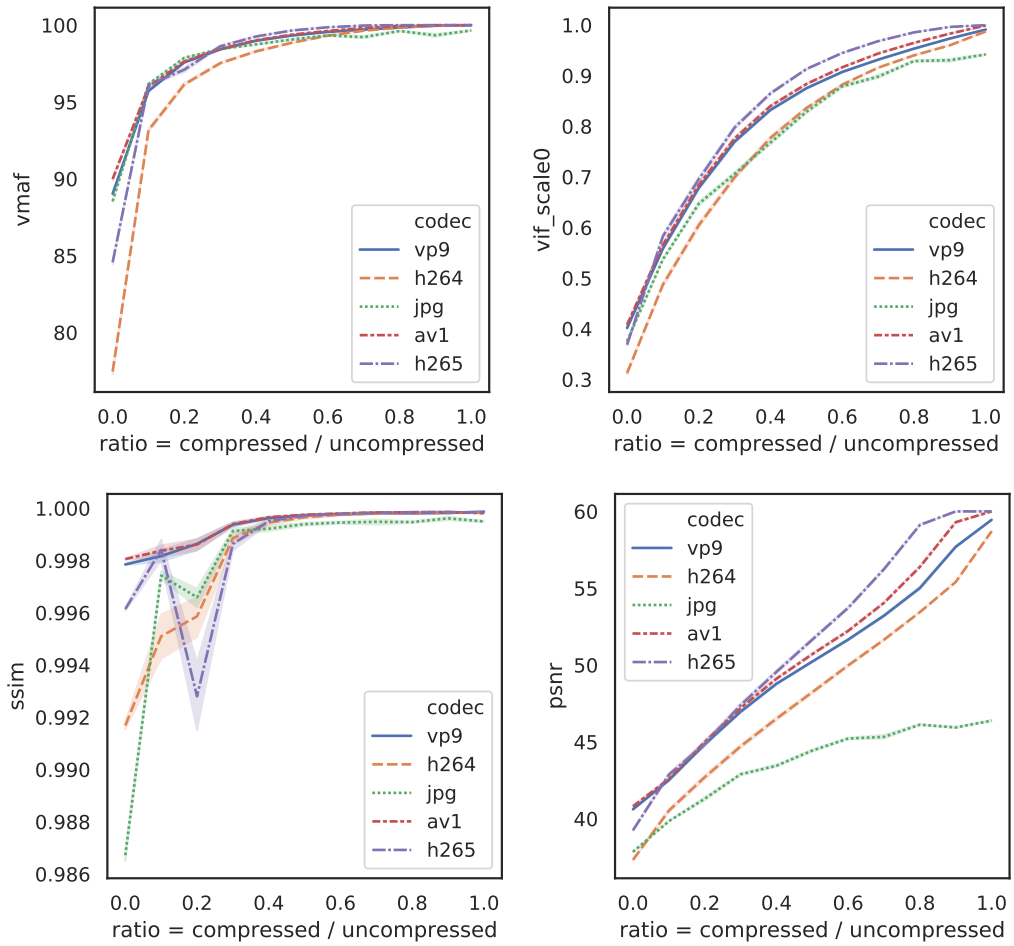


Figure 3.6: Comparison of estimated image quality with gained compression ratio, where the compression ratio is rounded to the first decimal, mean values of quality metrics, and 95% confidence intervals are calculated.

low correlation with subjective scores [Pon+15; SSB06]. Thus, it can be stated that a detailed analysis of which objective metric best reflects human perception in the context of high-resolution images is required.

For example, in the Tampere Image Database 2013 (TID2013) [Pon+15] PSNR has the lowest pearson correlation to subjective scores when only JPEG compression artifacts are considered. The TID2013 consists of medium resolution images and includes different distortions, e.g., noise. Furthermore, most of the recently published databases focus on medium resolution images, e.g., the KonIQ-10k Dataset [Hos+20], KADID-10k [LHS19] or the LIVE In the Wild Image Quality Challenge Database [GB15]. Such

datasets target user-generated content, include a larger number of images and the quality ratings are gathered using crowdsourcing studies. Most of these datasets can be used to train deep neural networks for image quality prediction, as is also shown in [GR18; LHS20; Hos+20]. On the other side, especially for videos, there are datasets available focusing on higher resolutions. In addition, also video quality models for higher resolutions show high correlation with subjective scores, e.g., Netflix's VMAF [Net; Lin+14] for UHD-1/4K video contents [GSR18; Rao+19a], or the recently standardized P.1204 series [Raa+20; Rao+20b]. The applicability of VMAF for quality assessment of still images will be evaluated in the following Section.

3.3.1 Crowdsourcing-based Quality Tests

As mentioned before, image quality or general *QoE* tests can also be conducted using crowdsourcing or online tests [Rob+20; Hos+17b; Hos+17a; Hoß+11; Nad+20]. However, a general drawback of crowd testing is that there is less control regarding environmental factors, the used setup to perform the test, general distractions, and more [Hos+13]. For example, it can be assumed that usual crowdsourcing participants do not have a high-resolution display with a powerful PC to play uncompressed videos or to show high-resolution images. On the other side, crowdsourcing tests include more variation in terms of the participants and it could be assumed that their used environment and setup is more realistic. Further, crowdsourcing tests require more effort in designing, the inclusion of hidden conditions as checks, or reduced overall duration [Hos+13]. Here, a linking of standardized methods and crowdsourcing approaches can be used to evaluate the reliability of such test paradigms. For example in [Nad+20], such an evaluation for speech quality assessment is performed. Naderi et al. [Nad+20] show that standardized methods and crowdsourcing yield comparable results.

The crowdsourcing paradigm for quality assessment of higher resolution images is still challenging. Most of the image quality datasets focus on lower or medium resolution images. Moreover, most quality evaluation studies use PSNR or SSIM or similar quality metrics that are not designed for higher resolution images or do not cover perceptual aspects. While in addition, most of the aforementioned studies do not include more recently developed image compression methods.

For this reason, in the following a crowdsourcing approach with modification to traditional lab tests targeting H.265 encoded images is proposed. To further evaluate the validity of the introduced crowd test, a standardized lab test for image quality evaluation was conducted.

3.3.2 Dataset and Processing Pipeline

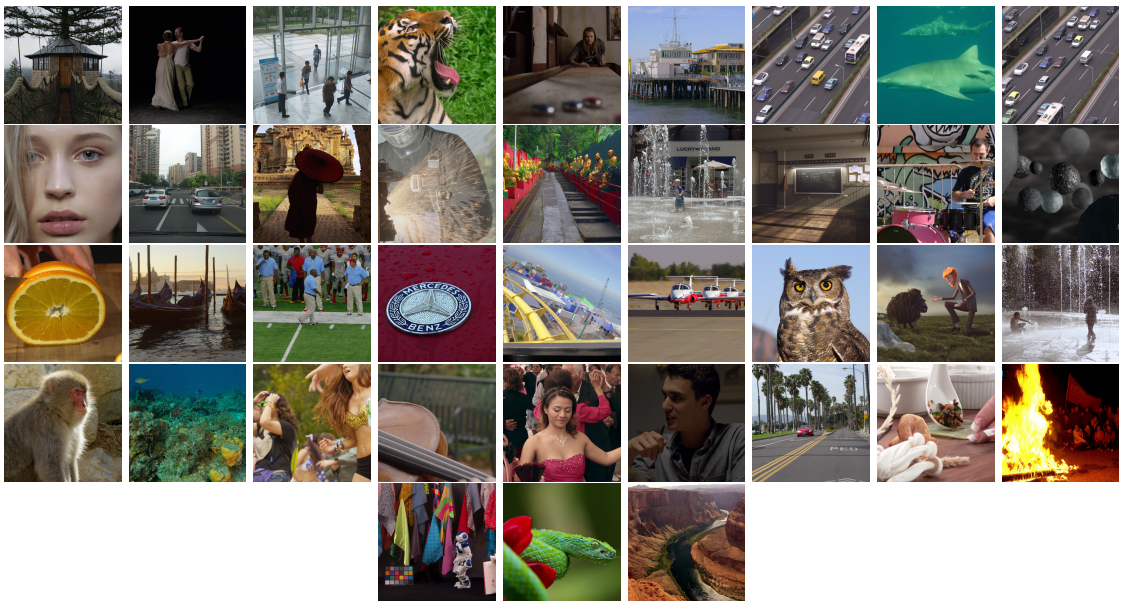


Figure 3.7: Overview of the used uncompressed source 4K frames, the extracted video frames are center cropped.

Typical high-resolution images have a larger resolution than Full-HD. As a starting point to analyze the quality of such images, UHD-1/4K video frames can be used and are widely accessible. In addition, this also enables a link of video and image compression and the relationship with regard to quality.

In total 39 different single UHD-1/4K frames have been extracted from several uncompressed UHD-1/4K videos. The source videos were available in a 4:2:2 or 4:2:0 chroma sub-sampled 10 bits lossless video format. Subsequently, all frames are center cropped to ensure that they have the same width and height of 2160 pixels. In Figure 3.7 all used source images are shown, the selection is based on a wide range of different video/image genres such as animated content, short movies, or documentaries. The rationale here is to cover a wide range of realistic images.

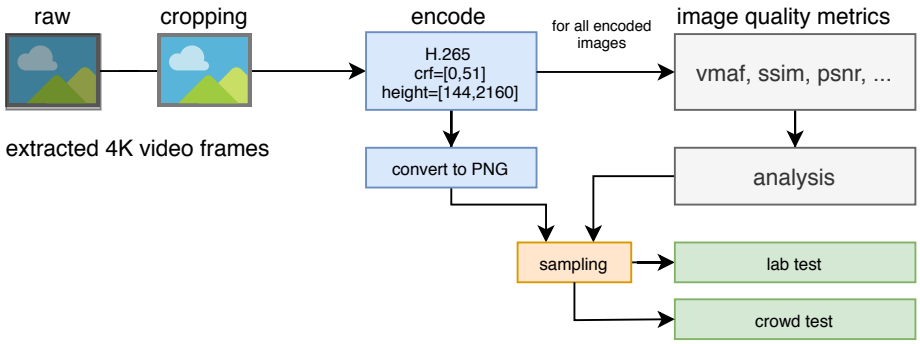


Figure 3.8: Image processing pipeline, starting from extracted raw 4K video frames, to cropping, encoding and designing subjective tests based on sampling.

Further, all extracted single frames were encoded with H.265 using FFmpeg 4.1 with several resolutions into 246,126 individual compressed images. H.265 was used because it was already reported that it outperforms JPEG [GR19], see further Section 3.2. The general processing pipeline is shown in Figure 3.8. Each image originated from a 4K video, is center cropped (so only the square 4K center part is used), and will be encoded with several settings. The target resolutions vary with a $height/width$ in the range of $[144,\ldots,2160]$ with a step size of 16 pixels. The specific small step size is selected to further analyze the impact of upscaling algorithms on image quality. As encoding, a crf based 1-pass scheme is used, here the value for crf is varied within the range of $[0,\ldots,51]$ with a step size of 1.

Afterward, for all encoded images, several traditional objective image quality metrics were calculated. For all metric calculations, the VMAF tool is used, thus also a VMAF score is estimated. Even though VMAF is designed for video quality analysis, it is suitable for images [BM20] and for higher resolutions, e.g., for 4K video [Net18a; Net18b; GSR18; Rao+19a]. In the case of images, it can be assumed that it is a still image video and the motion estimation feature can be neglected because it also has a generally lower impact on the estimated image quality scores that underly the VMAF calculation. This can be concluded by the low prediction performance of VMAF in the case of framerate variations for videos [Rao+19a].

As next, some initial analysis of the extracted objective metrics is performed. This analysis forms the base of data sampling to design the lab and crowdsourcing tests.

3.3.3 Analysis of Objective Scores

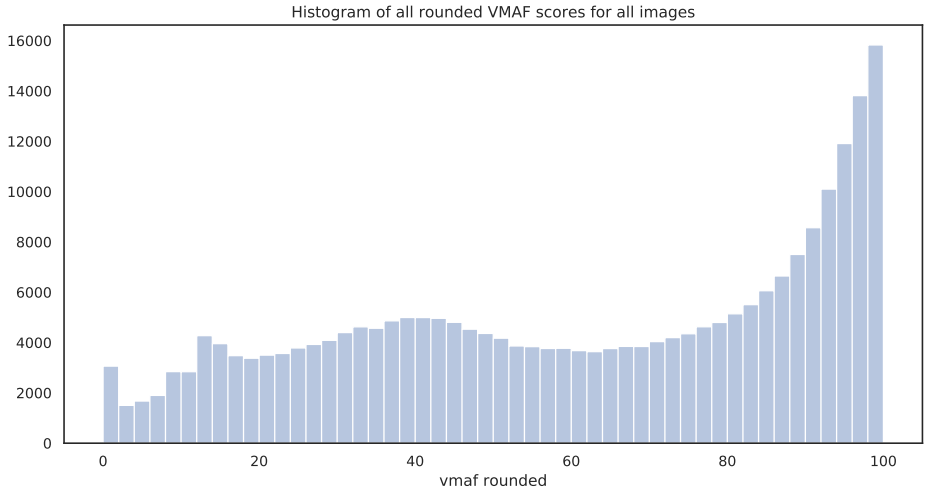


Figure 3.9: Histogram of rounded VMAF scores for all 246126 encoded images.

In Figure 3.9 the distribution of all rounded VMAF scores for all encoded images is shown. Especially in the high-quality range, starting from a VMAF score of 85, it is visible that more often a similar quality score is reached. This leads to the conclusion that high-quality scores can be reached with several encoding settings.

Moreover, a specific view of the encoding parameters is required.

$$bpp(image) = \frac{filesize(image) \cdot 8}{height(image) \cdot width(image)} \tag{3.4}$$

To enable a better comparability, the bits-per-pixel (bpp) measure is used, where $bpp(image)$ is defined for an image according to Equation 3.4. Here the $filesize(image)$ corresponds to the file-size in bytes of the constant mkv container including the H.265 encoded image as single encoded frame and $height(image) = width(image)$ due to the used center cropping approach.

In Table 3.1 log_2 values of the parameters and the raw parameters ($height$, crf, $filesize$, and bpp) are compared according to their correlation (linear pearson correlation coefficient, spearman and kendall) with the estimated VMAF quality score. Visible is that $log_2(filesize)$, $log_2(height)$ and crf have the highest correlation with VMAF scores in this context, they define the performed lossy compression parameters (here $filesize$ is the result of the compression parameters). Based on the table,

Table 3.1: Correlation values of VMAF scores and encoding parameters, sorted by pearson correlation; all values are rounded to 2 decimal places and sorted by absolute pearson values.

parameter~VMAF	pearson	kendall	spearman	|pearson|
$log_2(filesize)$	0.85	0.73	0.90	0.85
crf	-0.73	-0.58	-0.75	0.73
$log_2(crf)$	-0.61	-0.58	-0.75	0.61
$log_2(height)$	0.58	0.39	0.54	0.58
$height$	0.53	0.39	0.54	0.53
$filesize$	0.52	0.73	0.90	0.52
$log_2(bpp)$	0.40	0.31	0.45	0.40
bpp	0.29	0.31	0.45	0.29

$log_2(filesize)$ has the best correlation, however, it is interesting that there is no need to know about the used image resolution, showing that the estimated quality is mostly based on the filesize (=compression performance) only.

3.3.4 Bitstream-based Image Quality Models

Based on the previously identified no-reference features of the given images, that are encoded with H.265, the question arises whether such features can be used to predict image quality. The features and model is similar to bitstream-based video quality models, e.g., the P.1203 [Rob+18a; Raa+17; ITU17; Rao+19b; ITU17] and P.1204 series [ITU19a; ITU19b; Raa+20; Rao+20a] or meta-data based models for 360° videos [Fre+20a].

Here, it is important to mention that the crf value is usually unknown to the final decoding device, for this reason, this feature type is neglected and only meta-data will be used. In the following, two simple no-reference image quality models (*IMG-h265-para* and *IMG-h265-rf*) are trained to predict VMAF scores and evaluated briefly. For training and validation of the models, the introduced dataset was divided into 50%-50% partitions with no image overlap. Here, the first 19 images are used for training and the remaining 20 for validation.

As first, *IMG-h265-para* is a parametric model, that is based on Equation 3.5.

$$
\begin{aligned}
Q = clip(& \\
& a \cdot log_2(filesize) \cdot log_2(height) + \\
& b \cdot log_2(filesize) + c \cdot log_2(height) + d, \\
& 0, 100)
\end{aligned} \tag{3.5}
$$

In the case of *IMG-h265-para*, quality is estimated using several functions. The function $clip$ ensures that the values are within the range $[0, 100]$. The final coefficients that have been estimated during training are listed in Table 3.2.

Table 3.2: Coefficients for *IMG-h265-para* model.

coefficient	a	b	c	d
value	-1.8162842	27.88478854	31.95868242	-399.45484944

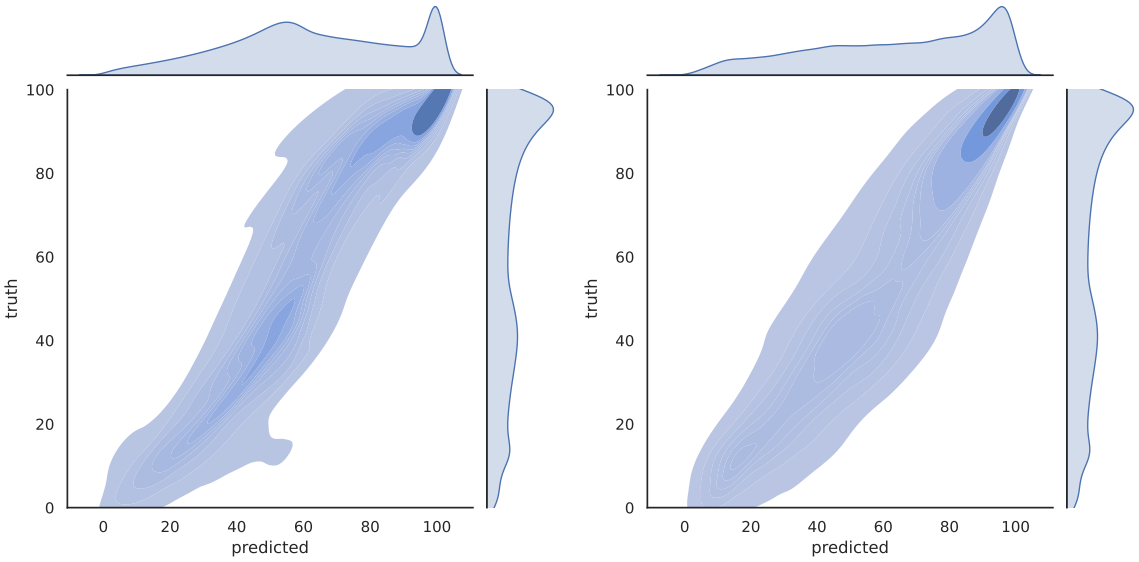
Secondly, similar to the parametric model, a simple random forest model, in the following referred to as *IMG-h265-rf*, was trained. Here, the input feature set is extended by $log_2(filesize) \cdot log_2(height)$ to have a synchronized structure compared to the parametric model. In total 100 trees have been used for training, the remaining parameters are default values for scikit-learn [Ped+11].

Table 3.3: Estimated performance metrics of two image quality models; values are rounded to 3 decimal places.

model	pearson	kendall	spearman	rmse
IMG-h265-para	0.903	0.750	0.916	13.250
IMG-h265-rf	0.910	0.751	0.920	13.188

In Table 3.3 the performance values for the validation dataset for both models are summarized. Both models show similar performance considering pearson correlation and rmse, the values for kendall and spearman are slightly different, however, they are still similar. In addition to the performance values, in Figure 3.10a and 3.10b kernel density plots for both models are shown.

Based on the analysis it can be concluded that both models perform well with a high pearson correlation. Though it should be mentioned that here only square images are



(a) *IMG-h265-para.* **(b)** *IMG-h265-rf.*

Figure 3.10: Kernel density estimations for model predictions, VMAF is used as ground truth.

considered, and all images are crf encoded with H.265, a more diverse set of encoding parameters (e.g. quality differences in 1-pass or 2-pass encoding [GRR20], or fixed bitrate encoding, presets) can lead to different results. Moreover, it further shows, that both model paradigms (parametric function fitting and random forest/machine learning algorithms) are capable of predicting quality scores with a good overall performance. The introduced models are only a proof-of-concept, because both models assume that VMAF is a good quality indicator even for image quality in case of high resolution images.

3.3.5 Data Sampling

To verify the suitability of VMAF and the compression performance of H.265, subjective tests can be used. However, some sampling of the encoded images is required, because it is otherwise not possible to perform a lab test, as every participant would need to rate all images to get *MOS* for all images. For this reason, a first selection is performed. This selection uses one representative image for each rounded VMAF score $\in [0, 100]$. A criterion to sample representative images for each corresponding rounded VMAF score is needed, because as it is visible in Figure 3.9 for each rounded

VMAF score several parameter combinations are suitable. The first sub-sampling uses for each source image and for each rounded VMAF score the following approach. First, a selection was performed on images that have a lower *height* than the mean *height* of all images in the current rounded VMAF score group. As next, only crf values lower than the mean crf and larger than the median crf of the remaining images are considered. Afterward, the representative image was selected by the maximum remaining *height*. This ensures a deterministic sampling, and based on the VMAF scores all images were similar in each of the groups, thus even different samplings would result in similar images.

Using the described approach of sampling it is possible to reduce the number of encoded images to approximately 100 stimuli per source image, in the remaining referred to as ICF_{100}. Here, it should be mentioned that some source images do not cover the full range of possible VMAF scores using the described encoding approach, e.g., some images show no changes in lower ranges due to the high spatial complexity of the source images. However, the mentioned sampling still creates for all source images in total approximately 3900 different distorted images. Because 3900 images are still not feasible for a test, a second sampling step was required to select suitable images for a traditionally conducted subjective image quality test, in the following referred to as ICF_{test}.

Here, in a first step, for each image the rounded VMAF scores are transformed linearly to $[1, 5]$-scaled MOS. Afterward, each MOS is rounded to the next integer. For each source image, a selection is performed in the following way. For each rounded MOS two encoded images are randomly selected for the test. It should be mentioned that some source images have only one encoded image for a specific rounded MOS value, thus in these cases, only one image can be used in the resulting test. The second sampling step resulted in a total number of approximately 8 to 10 stimuli for each source image that is used in the lab test. As a result, the overall test consists of 371 stimuli shown to the participants.

3.3.6 Lab Test for Image Quality

Using the finally sampled images of the dataset ICF_{test}, a lab test was conducted. To enable high reliability of results and further reproducibility, the subjective test

was implemented in a standardized lab environment as recommended in ITU-T P.910 [Rec08] and ITU-R BT.500-13 [ITU14b]. The image stimuli were presented using a 4K screen (55 inches 4K LG OLED) with a viewing distance of approximately 1.5 to 1.6 times the height of the screen, as recommend in ITU-T Rec. P.913 [ITU16b]. Before a participant rated the stimuli, a vision test (Snellen chart) was performed. Afterward, a short training phase followed before the rating of the stimuli started using the *ACR* scheme. In the training phase, possible image contents with typical distortions and the rating software were introduced to the participant. As rating software AVRateNG [AVR] was used. AVRateNG is a configurable client-server-based software for the conduction of subjective tests, the ratings are collected on the server-side, while a browser is used for the user input. In this case, both the server and client were running on the same computer. Some small modifications of the software were required to enable the applicability to images. AVRateNG is supposed to be a framework for such subjective tests and as default, it supports video ratings. One of the modifications was for example that the image was shown using a command-line video player (mpv[8]). To use this player changes in the configuration of AVRateNG are needed. Each image was presented using this software for 3 seconds and then rated by the participant according to the shown quality using the typical 5-point *ACR* scale. The overall subjective quality test lasted around 30 minutes. In total 21 participants took part in the study, mainly consisting of students and employees of the university.

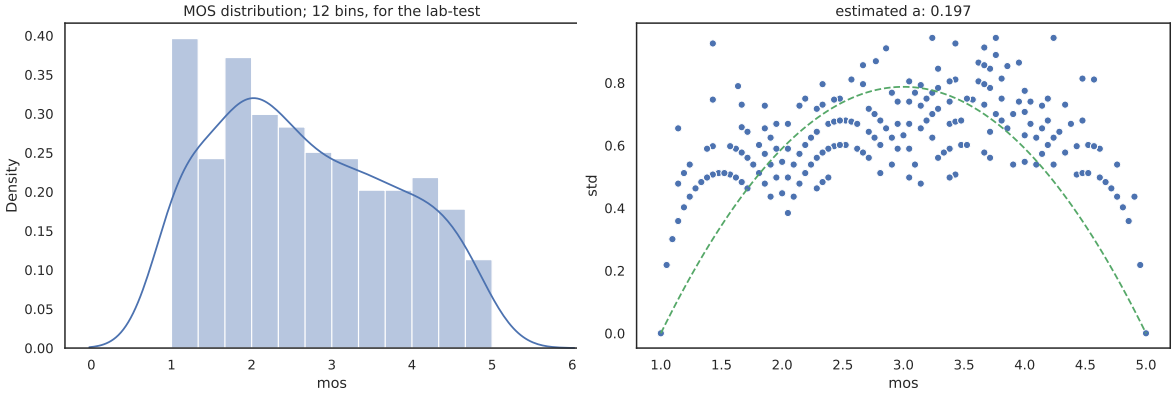
After conducting the subjective image quality test, for each shown stimulus ratings (*ACR*) in the range of $[1,5]$ are collected for each participant individually. In the following, as first all of the collected ratings are analyzed and later a comparison with objective image quality metrics is carried out.

3.3.6.1 Analysis of the Ratings

To investigate the reliability of the laboratory test results, a simple outlier detection was performed. The used detection algorithm is widely used in state-of-the-art, e.g. in [ITU17; ITU19a; Raa+20]. This outlier detection method uses a pearson correlation threshold to identify outliers. The used threshold was 0.8 (in other tests a value of

[8] https://mpv.io/

0.75 was used [Rob+18a; Rao+19a]). The procedure is that based on all ratings the pearson correlation of each individual rater is calculated and in case this correlation value is below the defined threshold this rater is classified as an outlier. For the lab test, no outliers have been identified, which is not uncommon in such a controlled lab test, because environmental and other influences are quite low.



(a) Distribution of MOS for the lab test.

(b) SOS evaluation plot for the lab test, including a quadratic fitted curve according to the SOS hypothesis.

Figure 3.11: Evaluation of the lab test.

For the overall score distribution, presented in Figure 3.11a, a tendency towards uniform or lower rated stimuli is recognizable. Here, it is required to consider the uniform sampling of VMAF scores in the filtering process as described in Section 3.3.5, it seems that lower VMAF scores are rated more critically by the participants. Another method to check the reliability of subjective tests is the SOS-analysis, proposed by Hoßfeld, Schatz, and Egger [HSE11]. The general idea is to perform a quadratic curve-fitting on *MOS* and standard derivation values of the ratings. Afterward the scale factor of the quadratic function a refers to the reliability of the test results. For the conducted laboratory test, the estimated a parameter is $a \approx 0.197$, furthermore the corresponding SOS-plot is shown in Figure 3.11b. The calculated value a is typical and valid for an image quality test, e.g., comparing to other reported values such as for the IRCCyN/IVC image test [HSE11] where the value is $a \approx 0.17$. For this reason, it can be concluded that the conducted subjective test has reliable results according to the SOS hypothesis.

3.3.6.2 Correlations with Objective Image Quality Metrics

Based on the conducted subjective test, it is possible to evaluate the performance of objective image/video quality metrics, i.e., VMAF [Lin+14; Net], ADM2 [Li+11], VIF [SB06](several scales), PSNR, SSIM [Wan+04] and MS-SSIM[WSB03]. For all images, the aforementioned quality metrics are calculated using the publicly available VMAF tool [Net].

Table 3.4: Correlation values of several objective quality metrics to the subjective scores; values are rounded to 3 decimal places.

metric	pearson	kendall	spearman
vmaf	0.919	0.757	0.925
adm2	0.868	0.722	0.901
vif scale2	0.861	0.740	0.911
vif scale3	0.852	0.786	0.941
vif scale1	0.846	0.674	0.859
ms ssim	0.701	0.658	0.851
psnr	0.698	0.524	0.719
ssim	0.658	0.802	0.948
vif scale0	0.619	0.472	0.643

In Table 3.4 for all considered objective metrics, correlation values are presented, namely the pearson correlation coefficient (pearson), the kendall rank correlation coefficient (kendall), and spearman's rank correlation coefficient (spearman). The best performing metric in this comparison is VMAF, directly followed by ADM2. However, ADM2 is used by VMAF as one of the underlying metrics. It was already analyzed in [Rao+19a] that ADM2 seems to have the strongest impact on the overall VMAF prediction for videos, thus a similar conclusion holds for image quality. In general, a high relationship between VMAF and the collected subjective scores is visible considering all three correlations.

Overall, the shown results are good, considering for example the pearson correlation for the same quality test that is conducted in several labs. As it is shown by Pinson and Wolf [PW03], the pearson values are ranging from 0.902 to 0.935 for such inter-lab correlations. Based on this it can be argued that the VMAF prediction is within the expected error range. Thus it can be concluded that VMAF can be used for image quality prediction.

3.3.7 Crowdsourcing-based Test for Image Quality

Traditional lab tests are a well-established tool to analyze the quality perception of participants. However, within the last years, crowdsourcing-based tests have increased in popularity [Hoß+11; Hos+13; Hos+17b; Hos+20; GB15]. Especially due to the fact that people with wider demographic backgrounds and more realistic viewing conditions can be recruited faster and at lower test costs, ensuring the overall sample of participants to be more realistic. For this reason, the sampled images ICF_{test} are additionally used in a crowdsourcing test.

3.3.7.1 Challenges

In general crowdsourcing-based tests introduce different aspects to the test design, conduction, and final analysis of the results [Hos+13]. Such differences originate from the diversity of possible crowd-users taking part in such a study, e.g., different end-devices, less constant environmental conditions, lighting conditions, distractions during the test participation, and even more [Hos+13]. Especially because of the variety of end-devices, that are used to show the stimuli, it is not always possible to assume that participants own a 4K screen or are even using it for such a crowd test. Usual crowdsourcing participants have more common or even older hardware, that is not required to be up to the latest technology trends. However, the focus of the introduced sampled images and processing pipeline is still high-resolution image quality assessment, which would require a 4K capable screen. Clearly, some crowd platforms allow to filter users based on equipment, however, this would also influence the test results. To tackle this problem and further not to exclude the majority of possible crowd test participants, the sampled images of the dataset ICF_{test} are pre-processed. The main idea is to convert each 4K square stimulus into 4 patches with a square size of 1080p each. In addition to solving the presentation dilemma, such an approach will also enable the possibility to analyze the connection between patch-based and overall image quality considering patches with higher resolution, in contrast to [Wie+18], where only lower resolution images are used.

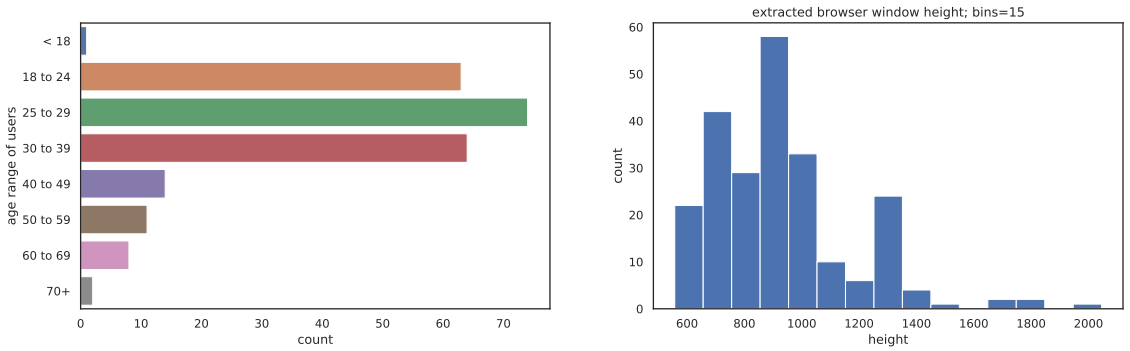
As test software, similar to the lab test, a modified AVRateNG [AVR] version was used. This version extended AVRateNG by some pre-checks during the test, e.g.,

whether the used display resolution is suitable (above 500p height), included a minimalistic demographic form (not covering any privacy or person-related data), and uses HTML 5 based pre-caching of images to remove influences of loading time for the images. An extended variant of this modification is published as AVRate Voyager [Gör+21b], which is publicly available[9]. In addition to the ratings for each stimulus, further demographic information, the used browser, and browser size are stored for later analysis of the crowd users.

Moreover, the original sampled 371 images of the ICF_{test} set resulted in 1484 Full-HD sized patches. The lab test was designed to last around 30 minutes for the complete rating of all images, whereas rating of 1484 patches is neither suitable for a lab test nor for a crowdsourcing-based test. Here, another modification to traditionally conducted full-factorial lab tests is required. In the crowdsourcing test each participant rates 150 uniform random sampled Full-HD patches, referred to as part-factorial design. Pre-tests showed that approximately 10-15 minutes are required to perform the designed crowd test, which is necessary as the overall duration has an influence [Hos+13] on the rating quality. Moreover, an explicit training phase was removed, to shrink the overall time for the test even more. This modification also results in the need for more participants in the crowdsourcing test, so that each shown patch is rated by around 20 participants in the ideal case. To rate all included images of the lab test it is thus required to have approximately 200 participants taking part in the crowd test, following the described part-factorial design.

In total 238 subjects took part in the study to rate image quality, they were recruited within the university, to also ensure comparability with the conducted lab test.

In the following, only participants who finally rated images are considered to be valid participants, all other participants were already removed (e.g. participants who just filled out the first form and never rated an image). First, the participants themselves are investigated in more detail, this is required for the design of future crowdsourcing-based tests.



(a) Age range of all participants of the crowd test; **Q1**.

(b) Distribution of used browser window heights for the crowd test.

Figure 3.12: Evaluation of the users of the crowd test.

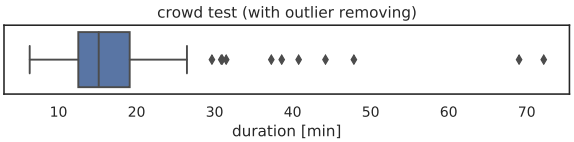


Figure 3.13: Duration required for the crowd test; most participants needed ≈15 minutes for the test.

3.3.7.2 Analysis of the Crowd Users

The participants have been asked to fill out a demographic form at the beginning of the test, the rationale of this questionnaire was to pre-cache the images during the time it takes to answer the questions. In total the following four questions (**Q1**, **Q2**, **Q3**, and **Q4**) have been asked.

- **Q1**: "What is your age?" (8 answer categories possible),
- **Q2**: "How good is your vision?" (6 answers possible),
- **Q3**: "Which option best describes your environment?" (3 possible answers),
- **Q4**: "What type of device are you now using?" (4 answers, 'Phone', 'Tablet', 'Laptop', 'Desktop')

In Figure 3.12a the results for the age question (**Q1**) are shown, it is visible that the recruited participants form a "younger" crowd, whereas even some older participants took part in the study. The next question (**Q2**) was a self-report about vision. The majority of the crowd users either selected excellent or good vision, whereas

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

some selected worse options, this question even created some confusion for some participants, where individual email exchange was performed for clarification. **Q3** refers to the environment of the participants, here also a self-report was used, as other approaches were considered too intrusive regarding test subjects' privacy for this test. Most participants were either in a quiet room or stated to be just minimally influenced by noise. The last question (**Q4**) refers to the user's device, here it is important to know that in the invitation email it was strongly recommended to use a PC or Notebook. In addition, the rating software used a check of the browser window size to ensure a minimum height and width, this check enforced that it is not possible to run the test on a smartphone or tablet respectively. This decision was made to include only participants with larger screens, because in a pre-test it was observed that some participants may use devices with very low display resolution. There were a few participants who took part in the test before the implementation of this check.

In addition to the questionnaire, AVrateNG also collected some generic information about the crowd participant. Here, only the window size and the used browser agent have been stored. In Figure 3.12b the used window heights are shown. There are some participants with a 4K screen within the crowd. Most of the crowd users used a window height of approximately 720-1000 pixels, which leads to an HD or Full-HD native display resolution. So the general assumption in the preparation of the crowd test, to only handle 1080p patches, is mostly confirmed. In addition to the gathered answers in the questionnaire, the overall duration for the crowd test can be estimated, shown in Figure 3.13. Most of the participants needed about 15 minutes to conduct the test which was the time that was initially planned for the crowd test.

3.3.7.3 Ratings and Score Distributions

The crowd test provides the participants with 150 out of 1484 to-be-rated image patches, that are randomly selected. Figure 3.14 shows how often image patches are rated. On average each image patch is rated by ≈ 17 participants. In total 1439 patches were rated at least 10 times. 45 image patches were rated less than 10 times.

Furthermore, in Figure 3.15a the distribution of *MOS* for all patches is shown. The rating distribution is similar to the laboratory test (see Figure 3.11a). However, there

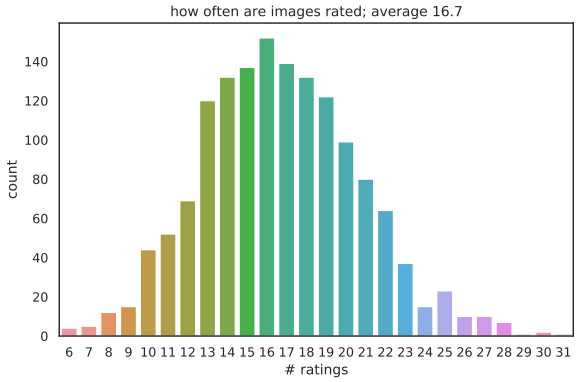
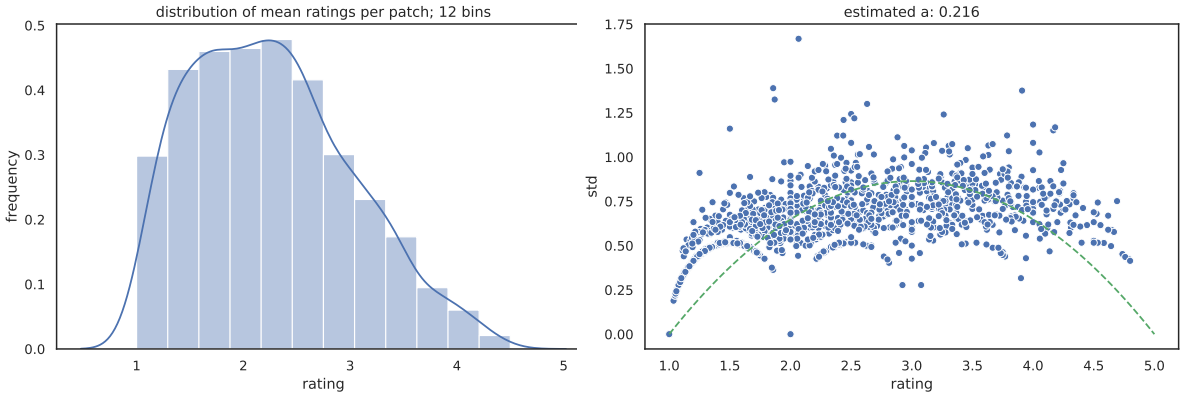


Figure 3.14: Histogram about the number of ratings for each image.

are fewer cases where a high-quality rating was selected by the participants. The reason for this is that some patches are difficult to rate due to compression artifacts, or because the patches are hard to recognize (e.g. a black patch). Also, this could be a result of the patching approach, because it decomposes the "global" picture and participants are less able to understand the image itself.



(a) Rating distribution for all patches.

(b) SOS evaluation plot for the crowd test, including a quadratic fitted curve according to the SOS hypothesis.

Figure 3.15: Evaluation of the crowd test.

Similar to the lab test, a SOS analysis [HSE11] was carried out, compare Figure 3.15b, where the mean and standard deviation values are shown for all patches. An a value of $\approx$ 0.22 was estimated, this value is similar to web surfing or video streaming tests [HSE11]. Furthermore, a shift to lower ratings is visible, similar to the distribution plots 3.15a. This can be explained by, for example, the more critical view of the individual participants.

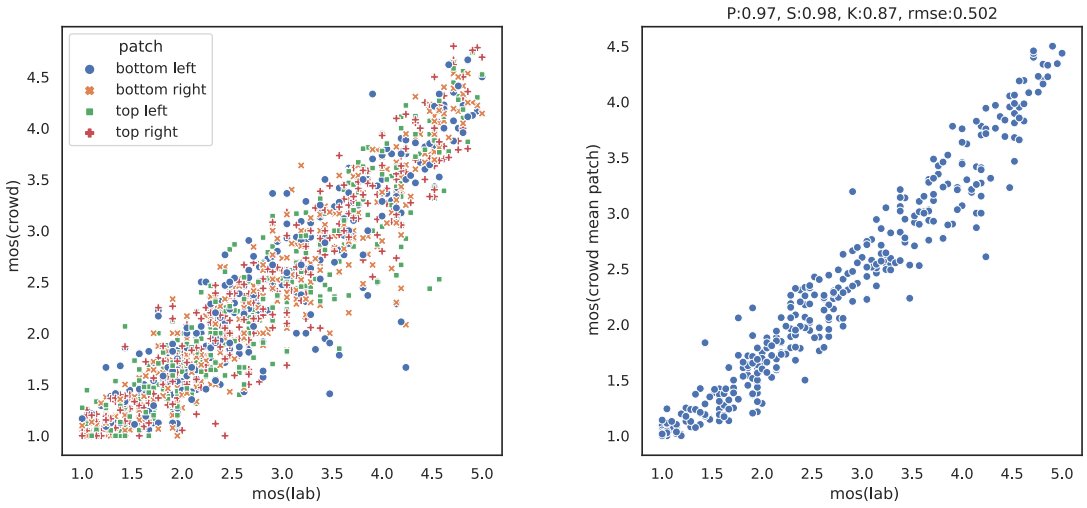
3.3.7.4 Correlations with Lab Test

It is further important to consider that each original image is split into 4 patches, which implies that for one image 4 individual ratings are collected within the crowd test. To compare the conducted crowd and lab test, first, each patch rating is compared to the lab test ratings, and later a mean rating of all patches.

In Table 3.5 correlation values, pearson, kendall and spearman, along with *rmse* values for each patch compared to the lab tests results are presented. First of all, the individual patch mean rating correlates high with the lab test ratings, compare also Figure 3.16a. However, the performance of all individual patches is nearly identical, thus it can be concluded that individual patches can be used individually for image quality evaluation. This is similar to results of Göring, Krämmer, and Raake [GKR19], where e.g. center cropped video frames showed similar results for the overall quality estimation in the case of videos, the approach is further described in Section 5.4. As next, the mean rating of all patches per image is considered to form the overall mean quality rating per full image. In Figure 3.16b the corresponding scatter plot is shown. The combination of all patch ratings leads to an overall better correlation (pearson of 0.97) than individual patches and an overall lower error (*rmse* of 0.502). However, in general a tendency for lower ratings for image patches can be observed, because the overall rating range in the case of the crowd test is $[1.0, 4.5]$ in contrast to $[1.0, 5.0]$ for the lab test. Here, it should be noted that in usual model development a linear fitting would be performed, to normalize the two tests. Such a linear fitting is already captured within the pearson correlation values. Moreover, the high correlation of mean patch ratings compared to the lab test also indicate that participants seem to not focus on individual image aspects for their quality rating.

Table 3.5: Correlation values of individual image patches in comparison with lab test ratings, furthermore *rmse* is calculated.

patch	pearson	kendall	spearman	rmse
mean	0.970	0.870	0.980	0.502
top right	0.954	0.954	0.823	0.527
bottom right	0.946	0.950	0.811	0.551
top left	0.941	0.948	0.805	0.565
bottom left	0.933	0.934	0.794	0.567



(a) Per patch ratings – mos(crowd)– compared to lab mean ratings –mos(lab).

(b) Mean patch rating in comparison with lab mean ratings –mos(lab).

Figure 3.16: Comparison of lab and crowd test.

3.4 Pixel-based Image Quality Prediction

In the last section, bitstream-based image quality models have been introduced. However, the models do not use any pixel data for quality evaluation. Typically, image quality is evaluated using objective metrics that rely on the pixel data for quality prediction, usually in a full-reference manner. Such models are mostly based on hand-crafted features (signal-based, computer-vision-based,...) [MMB12; Lap+16; Li+11; SB06; Liu+16; MB10] or deep neural networks [Kan+14; DMW16; Bos+16b; Bos+16a; LC10]. *DNN*s are already successfully used to address several image-related problems, for example classification [Sze+15], segmentation [LSD15], face-detection [PVZ15], and more [Lin+15]. For quality prediction, most existing models use a patching approach to avoid large input image sizes to *DNN*s (e.g. [Kan+14; DMW16]), because such large sizes result in large processing time or high memory consumption. A patching approach leads to the problem that connected image regions and distortions are lost and that such links are not considered by the model. Even though the subjective evaluation in the previous part showed that for humans this is not a problem, for a model the global picture may be required. Furthermore, for training a new *DNN* from scratch, a huge human-annotated database is required. Comparing to other image problems such as image classification (e.g. Ima-

geNet competition: 150,000 images [Rus+15]), available image quality databases (e.g. TID2013 [Pon+15]) are relatively small. However, a full re-training is not required in every use case and can instead be based on other approaches, for example using transfer-learning [Cho]. A pre-trained *DNN* could be used as the basis for re-training to a different problem space. In turn, it is hard to include in such a re-training process other per-determined, e.g., quality-related feature values without changing the complete *DNN*.

In the following Section, a no-reference pixel based image quality model called **deimeq** (**de**ep **im**age no-reference hybrid model to **e**stimate **q**uality) is introduced. The idea is to use a pre-trained *DNN* as automatic feature extractor and extend the features by classical image quality features. Instead of a pure patching approach, a hierarchical sub-image approach is used.

3.4.1 Deimeq Model Architecture

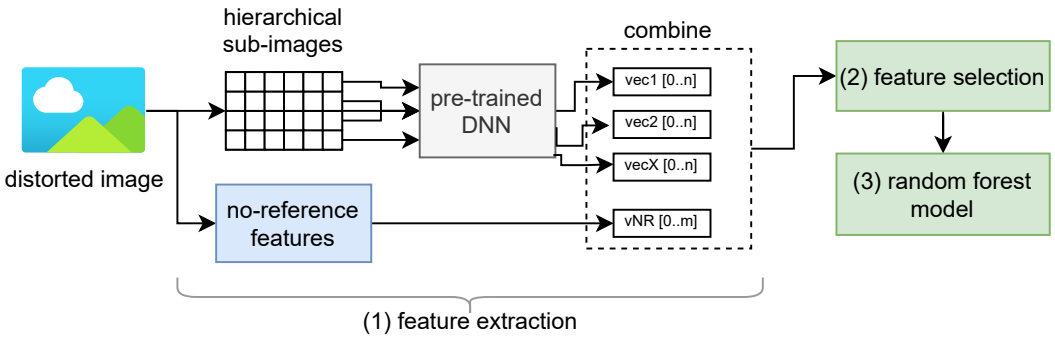


Figure 3.17: General model structure of **deimeq**; a pre-trained DNN is used together with no-reference features to train a final model component with feature selection. For each input image, hierarchically created sub-images are used.

In Figure 3.17 the general model structure of **deimeq** is shown. The pipeline consists of three main steps, (1) feature extraction with summarization and extension with state-of-the-art no-reference model features, (2) feature selection, and (3) training of the final machine learning model. The implementation uses the Keras framework [Cho+15] for *DNN*s and scikit-learn [Ped+11] for the machine learning part.

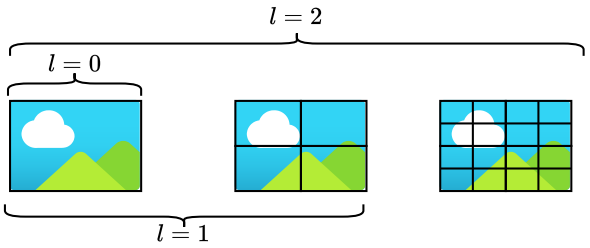


Figure 3.18: Hierarchical sub-images; each sub-image will be rescaled to the matching DNN input resolution.

3.4.1.1 DNN Feature Extraction and Reduction

Each input image is hierarchically divided into several sub-images as shown in Figure 3.18 and fed into a pre-trained *DNN*. All generated images are then re-scaled to the input size of the *DNN* and processed. With this hierarchical approach, the smallest patch size to be chosen for a given model implementation depends on the input-image resolution of the underlying pre-trained *DNN* model. For example, let an input image be of resolution $w_i \cdot h_i$ (with width w_i and height h_i of the input image) and the expected input resolution of the pre-trained *DNN* model be of $w_D \cdot h_D$ (with w_D the image width and h_D the image height expected by the *DNN*). Then, to preserve optimal image resolution under the constraint of the *DNN* input, the hierarchical patching should contain at least l levels, with $l = \max(\log_2(w_i/w_D), \log_2(h_i/h_D))$. Then, the smallest patches will just not be down-scaled before input, fitting the requirement of maintained maximal resolution stated above. Besides preservation of input image resolution at the smallest patch sizes, this approach ensures a connection between the distorted patches. Further, for the pre-trained *DNN*s some modifications are required. For example, for the Xception network [Cho16], the modification is that the last layer, which is usualyl the final classification and a fully connected layer, is excluded and replaced by an average pooling layer. With the modifications, such a classification network would generate, in the case of the Xception network, 2048 values for each sub-image. Using all these values would lead to a huge dimensionality, which is why a simple summation of each of the generated prediction values of the *DNN* is applied, assuming that the features would be similar in each of the sub-images. The generated feature vector f is sparse, which means it includes many zeros. Because of that, a second vector $f_{!0}$ is created, containing only the values that

are not zero. As next, for the generated feature vector f and the non-zero version $f_{!0}$, the following statistical values are calculated: mean, sum, standard deviation, skewness, kurtosis, harmonic mean (only for $f_{!0}$), geometric mean (only for $f_{!0}$), interquartile range and entropy. These values are a statistical description of the feature vector and are extended by one value that is the fraction of zeros in the feature vectors $1 - |f_{!0}|/|f|$ as an indicator of how sparse the feature vector is. In case of the Xception [Cho16] *DNN* this results in 2065 feature values for each image.

The total sum of the generated features is quite high in comparison with other state-of-the-art no-reference metrics. For this reason, to reduce the overall calculation and dimensionality, an automatic feature selection step is included in the machine learning pipeline.

3.4.1.2 Extension of Features

The *DNN* features can be extended by state-of-the-art no-reference values from other models if desired. Examples for such no-reference features are the brisque [MMB12] and niqe [MSB13] features. Both features are luminance-based and when combined perform quite well, also in comparison to other state-of-the-art models such as VeNICE [DWM17].

Furthermore, re-trained model variants of the features brisque and niqe are used as comparison baseline models. These re-trained models are based on the same feature-selection and random forest pipeline that is used for **deimeq**. With this re-training, it is ensured that the baseline model performance is the best possible. In this step, also other no-reference quality features or image aesthetic features could be introduced.

3.4.1.3 Random Forest Model and Feature Selection

The last step consists of a feature selection and training of a random forest model. The feature selection step uses an ExtraTreesRegressor using $0.001 \cdot mean$ as the threshold for feature importance selection. For all generated models 100 decision trees are used for the random forest model with mean squared error (MSE) as a split

criterion. All other parameters are default values provided by the used scikit-learn framework [Ped+11].

It should be noted noted, that the general idea is not restricted to these algorithmic choices or settings. Other combinations may be suitable and will probably perform with similar results. **deimeq** is a meta concept consisting of the idea to use a pre-trained *DNN* with hierarchical sub-images and additional features to predict image quality using machine learning algorithms.

3.4.2 Evaluation of deimeq

To evaluate the proposed method two databases are used in a cross-dataset evaluation approach. Using a cross-database evaluation will ensure that the model is not over-fitting to a specific database, and additionally, it shows how the model performs on completely unknown data. It should be mentioned, that both datasets are lower resolution datasets, and a sub-image level of $l = 2$ is used. Furthermore, for the **deimeq** model, different pre-trained *DNN*s are analyzed in comparison to re-trained no-reference baseline models namely **brisque** and **brisque+niqe**. It is also checked whether extending the model with these no-reference features will lead to higher prediction accuracy. As metrics for evaluation of model performance several correlation values (pearson, kendall, spearman) and the root mean squared error (RMSE) are used.

3.4.2.1 Datasets for Training and Validation

Before the evaluation both image quality datasets are briefly described, namely the Live-2-database [SSB06] and TID2013 [Pon+15]. More image quality databases are available, e.g. CSIQ [LC10]. However, they are with even lower resolutions, lesser images, or include only gray images. Live-2 and TID2013 have been selected, because they include similar distortion types (not only compression artifacts) and have approximately the same number of source images.

In Table 3.6 all key properties of both databases are summarized. The Live-2 dataset consists of 29 source images, in contrast to the 25 images of TID2013. TID2013

Table 3.6: Image Quality Assessment Datasets; quality scores are transformed to [0,100] scale.

	Live-2	TID2013
# source images	29	25
# distortion types	5	24
# total distorted images	779	3000
image resolution	768x512	512x384
quality score min	0	3.4
quality score avg	51.5	62.1
quality score max	100	100

includes approximately 5 times more distortion types, therefore the total number of distorted images is approximately three times higher. Both datasets share some similar distortions. The image resolution for both datasets is relatively small for today's image sizes, considering that current cheap smart phones already create 8 MP images. However, for proving the effectiveness of the image quality modeling approach, the considered databases are practically useful.

For both datasets, the published quality scores are transformed to the same scale for unification. To this aim, the quality scores are normalized ($[0, 100]$-DMOS in case of Live-2; $[0 - 10]$-MOS in case of TID2013) to a $[0, 100]$-score using a linear mapping approach, where 0 is the lowest and 100 is the best quality score.

3.4.2.2 Performance of deimeq Model Variants

Different pre-trained *DNN*s are analyzed, namely Xception [Cho16], VGG16 [SZ14], VGG19 [SZ14], ResNet50 [He+15], InceptionV3 [Sze+15], InceptionResNetV2 [SIV16] and MobileNet [How+17]. All used *DNN*s are classification networks trained for the ImageNet competition [Rus+15]. For feature extraction, the last layer is excluded and replaced by an average polling strategy to access the prev-last layer values. In all experiments, the models are trained on the Live-2 database and validated with the TID2013 images.

Considering Table 3.7, only **deimeq** variants with Xception, InceptionV3 or ResNet50 *DNN*s are able to outperform the baseline **brisque/niqe** model variant. All other *DNN*s are not suitable in the setup, concluding that they are not reflecting quality-related features in their layers.

Table 3.7: Performance of **deimeq** model variants and *brisque*, *P*=pearson, *K*=kendall and *S*=spearman correlations and RMSE values; **B**=brisque/**N**iqe as additional features; sorted by correlations; trained on Live-2 and validated with TID2013

model	used dnn	+feat.	*P*	*K*	*S*	RMSE
deimeq+	xception	**B**	0.53	0.32	0.47	17.33
deimeq	xception	**B**+**N**	0.53	0.32	0.46	17.04
deimeq	inceptionV3	**N**	0.53	0.28	0.40	15.99
deimeq	inceptionV3	**B**	0.52	0.33	0.47	17.69
deimeq	inceptionV3	**B**+**N**	0.52	0.31	0.45	17.35
deimeq	resnet50	**N**	0.52	0.30	0.43	16.77
deimeq*	xception		0.51	0.27	0.40	19.43
deimeq	inceptionV3		0.51	0.27	0.40	16.61
deimeq	resnet50	**B**	0.50	0.32	0.47	17.15
deimeq	xception	**N**	0.50	0.27	0.38	17.82
deimeq	resnet50	**B**+**N**	0.49	0.32	0.47	17.33
brisque			0.48	0.31	0.44	18.92
brisque		**N**	0.48	0.30	0.44	18.48
deimeq	vgg19	**B**+**N**	0.48	0.30	0.43	17.66
deimeq	vgg16	**B**	0.48	0.29	0.42	18.47
deimeq	vgg16	**B**+**N**	0.48	0.29	0.42	18.26
deimeq	incept-res	**B**+**N**	0.48	0.27	0.40	18.26
deimeq	vgg19	**B**	0.47	0.30	0.43	18.03
deimeq	mobilenet	**B**	0.47	0.30	0.43	17.54
deimeq	mobilenet	**B**+**N**	0.47	0.28	0.41	17.77
deimeq	incept-res	**B**	0.46	0.26	0.38	18.50
deimeq	resnet50		0.44	0.27	0.40	21.12
deimeq	mobilenet	**N**	0.41	0.20	0.30	18.18
deimeq	incept-res	**N**	0.41	0.19	0.28	20.23
deimeq	vgg19	**N**	0.38	0.17	0.25	19.60
deimeq	vgg16	**N**	0.36	0.17	0.24	21.02
deimeq	vgg19		0.36	0.14	0.21	26.17
deimeq	vgg16		0.29	0.13	0.19	26.40
deimeq	mobilenet		0.27	0.11	0.16	25.34
deimeq	incept-res		0.25	0.11	0.17	24.89

The best performing model is **deimeq+**, using the Xception network in combination with **brisque** features. The performance of **deimeq** using **brisque** and **niqe** features is approximately the same as for **deimeq+**. An approximately 10% higher Pearson-correlation can be observed with the **deimeq+** variant. A similar performance boost of the other correlations and the RMSE can be seen. In contrast, using only the *DNN* provided features without extension of no-reference features marked as **deimeq***, it results in approximately 6% higher correlation than the individual baseline models. There are also other models listed with similar performance. For

example, **deimeq** with InceptionV3 shows similar performance regarding correlation and error. Furthermore, the performance of the VGG16, VGG19, InceptionResNetV2, and MobileNet is worse than the baseline models. Here, it needs to be considered that the used *DNN*s were originally designed for image classification tasks or were optimized for speed. Moreover, the cross-dataset comparison is a hard task, because the DNNs are evaluated using complete unknown data and the TID2013 dataset includes more distortions. The comparison with the baseline models shows that DNNs are able to outperform traditional developed no-reference quality models.

3.4.3 Linking Image and Video Quality Prediction

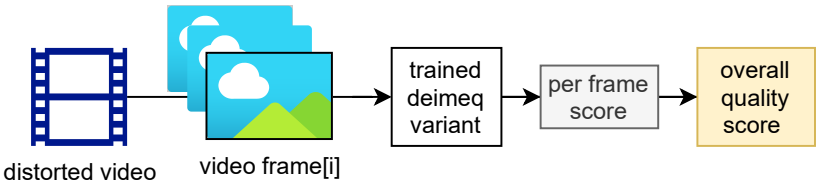


Figure 3.19: Meta-Model structure of **deviq**, using a specifically trained **deimeq** model for per-frame prediction.

The **deimeq** model is a demonstration of how *DNN*s trained for image classification can be ported to the task of image quality prediction. Similarly, pre-trained *DNN*s can be used for video quality prediction. The video quality model **deviq** [GSR18] uses a slightly modified pipeline as **deimeq** (see Figure 3.17 and compare with Figure 3.19). Here, a specifically trained **deimeq** model is applied to each video frame, and afterward based on all individual predicted frame scores an overall video quality score is estimated.

Because **deviq** targets video quality prediction for UHD-1/4K several adaptions are required. For example, the hierarchical sub-images are extended to a level of $l = 4$, which results in a total number of 85 sub-images per frame. Moreover, because there are no per-frame subjective scores accessible for videos, the data used for training the **deimeq variant** is based on per-frame VMAF scores. Such an approach is also used in other deep neural network based video quality models, e.g., NR-GVSQI [Bar+19] or DEMI [Zad+20a]. It should also be mentioned that processing such many sub-images on a per-frame based in the case of a UHD-1/4K video results in a huge

requirement for processing. The **deviq** model was trained on a synthetic dataset with average VMAF as ground truth data. And it was evaluated using a subjective video quality test, which was a pre-test to test_1 of the AVT-VQDB-UHD-1 [Rao+19a], with similar encoding settings and videos. The datasets for training and validation do not share common videos, however, they include the same encoding conditions, such as video codec, bitrates, and resolutions.

Table 3.8: Analysis of state of the art models and **deviq** compared to MOS for the validation dataset.

method	pearson	kendall	spearman
vmaf	0.92	0.72	0.89
deviq	0.84	0.61	0.81
brisque+nique	0.75	0.53	0.73
vifp	0.70	0.52	0.67
msssim	0.69	0.46	0.61
ssim	0.65	0.45	0.60
psnr	0.34	0.60	0.76

In Table 3.8 the evaluation of the **deviq** model compared to other state-of-the-art models is shown. Important to mention here is that **deviq** has been published before **deimeq** (using a slightly different feature aggregation approach) and even before there was a UHD-1/4K support in VMAF implemented. First of all, it can be seen that VMAF has a good prediction performance for UHD-1/4K video quality. Moreover, because **deviq** is trained using VMAF, it is clear that **deviq** cannot outperform VMAF. However, the main difference between **deviq** and VMAF is that **deviq** is a no-reference model while VMAF is full-reference. Based on the other re-trained no-reference models (**brisque+niqe**) it can be seen that **deviq** outperforms them. However, the main disadvantage of **deviq** is processing power and the lack of motion aspect-related features considering video quality. Moreover, the models **deimeq** and **deviq** demonstrate the connection of image and video quality prediction.

3.5 Summary and Conclusion

Image compression and quality evaluation are still highly relevant topics, due to the increase of uploaded photos and technology improvements. Moreover, it was shown that both topics are highly related to video compression and quality. As the first

outcome of this chapter, it can be concluded that video encoders can be applied to high-resolution images and that they outperform established formats such as JPEG in comparison with objective metrics. To validate this, a synthetically generated dataset based on high-resolution images was used and compressed with several video encoders. For example, AV1 is one of the evaluated video encoders that shows promising results for image compression. AV1 outperforms JPEG and other methods considering the quality and compression efficiency.

Furthermore, subjective methods can be used to evaluate the quality of high-resolution images even in crowdsourcing-based test. For this reason, a high-resolution image dataset has been created using H.265 for image compression. To carry out such crowdsourcing tests modifications of the test design are required. For example, a conversion of images to several patches or random selection of stimuli to the participants in a part-factorial approach is needed. Using the mentioned modifications, it is possible to even get similar results for the crowd test compared to a conducted lab test. It should be mentioned that even the per-patch quality evaluation is already similar to a lab test. The analysis of passively and further collected data of crowd test participants shows that the usual "university" crowd user has a Full-HD or HD display. Moreover, it was also shown that VMAF can be used as an image quality metric because it showed high correlations to the *MOS* of the lab and crowd tests.

Considering that no-reference image or video quality prediction is still a challenging task, a no-reference deep learning based image quality model (**deimeq**) has been proposed. **deimeq** is evaluated using a cross-dataset evaluation, showing good results and can be extended by other traditional state-of-the-art image features. It should be mentioned that due to the requirements of processing this evaluation was performed using medium resolution images, however similar models can be developed for higher resolution images as well. To bridge the gap between image and video quality models, the **deviq** model was briefly introduced, showing that deep learning based video quality models can outperform other no-reference video quality models. However, they typically require more processing power and in the case of **deviq** motion-related aspects are not covered.

Chapter 4

Models for Video Quality Prediction

Because screen sizes are increasing more and more [Sam18], and TV screens with such resolutions are becoming affordable, more videos with UHD-1/4K resolution are streamed on YouTube, Netflix [Net18a], Amazon Prime Video, and similar platforms. For this reason, in the following Chapter video quality prediction models will be introduced that are specifically designed for higher resolutions, e.g., up to UHD-1/4K. Each of the introduced models is evaluated in large-scale experiments and for different prediction targets. However, the described architecture, consisting of features, speed up of calculations, and machine learning pipeline can also be used for resolutions beyond UHD-1/4K in the future.

The following questions will be answered in the subsequent Chapter:

- Can a common feature set and architecture be used to estimate video quality for several application scopes? (**Research Question** 1)
- Is it possible to develop no-reference pixel-based video quality models that have 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? (**Research Question** 2)
- Are the proposed models able to predict more than only mean opinion scores? (**Research Question** 4)

The chapter is mostly based on the following publications:

[GRR19] **Steve Göring**, Rakesh Rao Ramachandra Rao, and Alexander Raake. "nofu - A Lightweight No-Reference Pixel Based Video Quality Model for Gaming Content". In: *Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*. Berlin, Germany, June 2019

[Gör+21a] **Steve Göring**, Rakesh Rao Ramachandra Rao, Bernhard Feiten, and Alexander Raake. "Modular Framework and Instances of Pixel-based Video Quality Models for UHD-1/4K". in: *IEEE Access* 9 (2021), pp. 31842–31864. DOI: 10.1109/ACCESS.2021.3059932. URL: https://ieeexplore.ieee.org/document/9355144

4.1 General Video Quality Model Architecture

The prediction of visual quality is an important tool for several applications. For example, not in all cases, subjective tests can be applied directly to evaluate video quality. Thus, there is a need to use automated video quality prediction as a replacement for subjective tests. However, such models can have different input formats, and for this reason, several model types have been established, ranging from no-reference over reduced-reference to full-reference video quality models and variants.

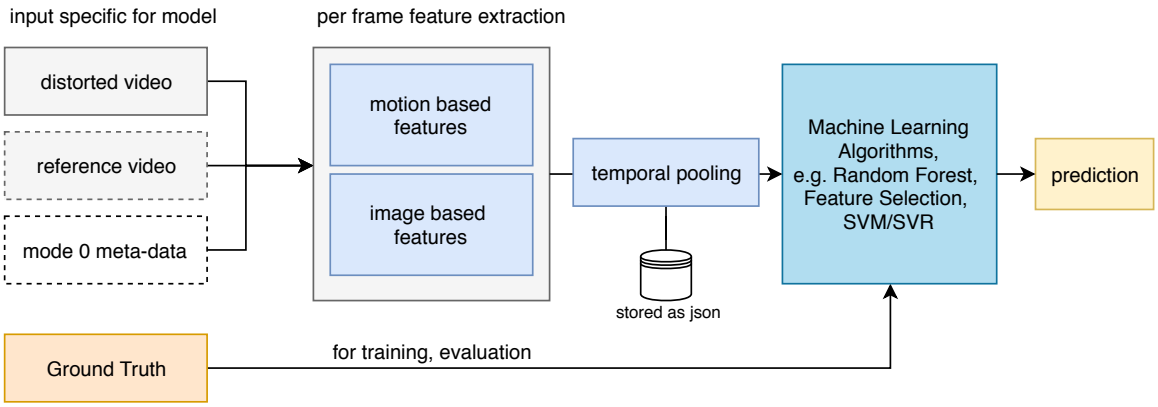


Figure 4.1: General Video Quality Model structure consists of feature extraction, temporal pooling, and machine-learning-based model training or prediction.

To tackle the problem of video quality estimation with different types of available input data, several pixel-based video quality models have been developed with a

specific focus on high-resolution videos. All models behave similarly, moreover, they share specific features and conceptual parts in a common framework. In Figure 4.1, the general structure of the proposed video quality models is illustrated. Usually, the distorted video and reference video have the same input resolutions, pixel format, and framerates, otherwise, before applying the 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*[1] and the specific model instances are part of *pixelmodels*[2]. 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, advanced temporal pooling is performed, this method includes several statistical pooling approaches, this approach is also applied to solve different video quality research problems [Gör+19; GRR19].

As a last general step, all pooled features are used to train a machine-learning algorithm. For the models that are described here, a *RF* (120 trees for a no-reference and 240 for a full-reference model) with a previously applied feature selection step using the ExtraTreesRegressor algorithm is used. The number of trees for all models has been evaluated using 10-fold-cross validation in several additional training runs, and the selected parameters showed the most stable behavior. The implementation is based on Python 3 and uses scikit-video[3] for video processing and scikit-learn [Ped+11] for all machine learning parts. However, it should be mentioned that the introduced models are not restricted to the used machine learning algorithms. Various algorithms have been analyzed, 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, the prediction accuracy of all models can be analyzed. To this aim, several commonly established evaluation performance metrics are used, e.g., for the MOS prediction scenario Pearson Correla-

[1] https://github.com/Telecommunication-Telemedia-Assessment/quat
[2] https://github.com/Telecommunication-Telemedia-Assessment/pixelmodels
[3] http://www.scikit-video.org/stable/

tion 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, the individual parts of the general model structure are described in more detail. This covers the pixel-based features, followed by a description of details regarding speed up of calculations, the used temporal pooling, and finally concludes with different instances of the general model pipeline and their respective use cases.

4.1.1 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 handling such effects. In addition, also masking effects can have a strong influence on perceived video quality [RZM09]. To describe the effects that are the reasons for the final quality rating of a user, the features are grouped into two general sets, namely motion-based (**mov**) and image-based no-reference features (**img**). Further, several other features are included, e.g. image full-reference features (**img-fr**). To enable the described models to use bitstream or meta-data, bitstream specific features (**bs**) are estimated as well. Table 4.1 summarizes all features of the shown model pipeline, moreover references to the source of the given features are provided additionally. Features marked with **own** are features that have been developed by the author. It is noted that each feature produces either per-sequence values (e.g. in the case of bitstream features) or per-frame values. Further, **brisque** has been added as additional features in the table, it will only be used for one specific model.

Some of the **own** implemented features can also be used for different video quality-related research directions, for example for gaming video quality [GRR19] or automatic estimation of the perceivable differences of UHD-1/4K and Full-HD [Gör+19]. A brief overview of such extended applications will be described in Chapter 5.

Table 4.1: 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 [Gör+19]	1/F
fft	img	[KAR15]*	1/F
blur	img	own [Gör+19]	1/F
colorfulness	img	[HS03]*	1/F
tone	img	[ASG15]*	1/F
saturation	img	[ASG15]*	1/F
scene_cuts	mov	own [Gör+21a]	1/F
movement	mov	own [Gör+19]	1/F
temporal	mov	own [Gör+19]	1/F
si	img	[ITU08b]	1/F
ti	mov	[ITU08b]	1/F
blockmotion	mov	own [GRR19; Gör+19]	3 /F
cubrow.0	mov	own [GRR19]	1/F
cubcol.0	mov	own [GRR19]	1/F
cubrow.1.0	mov	own [GRR19]	1/F
cubcol.1.0	mov	own [GRR19]	1/F
cubrow.0.3	mov	own [Gör+21a]	1/F
cubcol.0.3	mov	own [Gör+21a]	1/F
cubrow.0.6	mov	own [Gör+21a]	1/F
cubcol.0.6	mov	own [Gör+21a]	1/F
cubrow.0.5	mov	own [Gör+21a]	1/F
cubcol.0.5	mov	own [Gör+21a]	1/F
staticness	mov	own [GRR19; Gör+19]	1/F
uhdhdsim	img	own [Gör+19]	1/F
blockiness	img	own [GRR19]	1/F
noise	img	[DJ94]	1/F
PSNR	img-fr		1/F
SSIM	img-fr	[Wan+04; WSB03]	1/F
VIF	img-fr	[SB06]	4/F
fps_est	mov-fr	own [Gör+21a]	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	[MMB12]	36/F

4.1.1.1 Per-frame No-reference Features

Several features that are calculated on a per-frame basis have been developed or re-implemented. For example, *colorfulness* [HS03], *tone* [ASG15], and *saturation* [ASG15] are features that were already used in image aesthetics prediction, which are re-implemented 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 quality perception. Moreover, a similar argumentation follows for the own developed *contrast* feature, which is estimated using histogram equalization. Here, the normalized average difference before and after correction of the histogram based on the cumulative distribution function is used. Furthermore, spatial and temporal information are additional factors influencing video quality, for example comparing UHD with HD, usually, spatial information is increased. For this, the implementation[4] of the SI and TI measure is adapted and integrated in the feature extraction. Both feature values are in the following referred to as *si* and *ti*, both are based on ITU-T Recommendation P.910 [ITU08b].

Besides *si* or *ti*, videos are in the context of DASH re-scaled during encoding to lower resolutions to save bandwidth, such re-scaling introduces degradations in sharpness, or adds additional blurriness. Usually, users rate lower, if the images or videos lack sharpness. For this reason, a blurriness feature *blur* is included in the feature set. The feature calculation is based on Laplacian variance. Each frame is converted to a grayscale image and afterward 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, a blurriness score is estimated. As another way to recover some information about re-scaling, a re-implemented *fft* feature is included, it is based on [KAR15]. With similar motivation, especially for models that have no access to the native distorted video resolution, the similarity to the re-scaled Full-HD frame as *uhdhdsim* is measured using PSNR as a criterion. Here, for example, a UHD-1/4K frame is re-scaled to Full-HD resolution (half of the input resolution) and up-scaled to 4K (to the origin resolution), afterwards, PSNR is calculated for the re-scaled and non-re-scaled frame. In addition to typical blurriness degradations, also blockiness can be observed in the case of a badly selected encoding setting or a "fast" preset of

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

the used encoder, that occurs for example in live-streaming scenarios. To measure block artifacts introduced due to high or suboptimal compression as in a live context, measures for *blockiness* have been developed. There are already features to measure blockiness reported in the literature [Per14; QTG10], however, these features usually assume a fixed block size and are developed specifically for JPEG compression. To overcome these limitations, an own feature has been implemented. It shares some of the general ideas of the aforementioned blockiness estimation approaches. For a given frame f of a video, a canny edge detector [Bra00] is applied. This calculation results in the edges noted as e, where e is a two-dimensional array with n rows and m columns, where $e[i,j]$ refers to the value of the ith row in the jth column. As a next step, the following values are calculated: for each column j a value $cs[j] = \frac{1}{n}\sum_i e[i,j]$, and for each row i respectively the value $rs[i] = \frac{1}{m}\sum_j e[i,j]$. The estimated values cs and rs are column and row summations normalized by the number of rows/columns. Then, for a given blocksize b for each shift $s \in [0..b]$ the mean value of a subset of $cs \mid rs$ is estimated. For example, for a shift s every bth value in $cs \mid rs$ starting from s is selected. Afterward, for such a selection the mean value is calculated. As a result, mean values for all possible shifts are obtained, and it is assumed that a maximum value of the shifts indicates where possible block artifacts can be found. The difference of this mean value to the selected values in $cs \mid rs$ is measured. Using this approach values (mD_c, s_c) and (mD_r, s_r) are calculated, where mD_c is the mean difference value for blocksize b using a shift of s_c considering columns, analog for rows handled in (mD_r, s_r). Finally, for a given blocksize the following value is estimated as measure $\sqrt{|mD_c - mD_r|}/2^{|s_c - s_r|/b}$. This measure has a larger value if there are visible block artifacts in the frame. Usually, blocks have a square shape, resulting in a measurable difference $mD_c - mD_r$ in both directions x and y. The estimated value is further normalized based on the assumed blocksize and shifts. The calculation is repeated for commonly used block sizes $b \in [8, 16, 32, 64, 128]$, and the final measure *blockiness* is the maximum of all estimated values. The implemented feature is faster compared to other state-of-the-art methods, however, it relies on a fixed block alignment, which is not always the case in videos. The feature has been checked with different real-world videos, consisting of block artifacts and it was found out that it is a good approximation.

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, motion-related features are important for the characterization of a video and are for this reason included in the model pipeline. As a first feature, a motion estimation approach that calculates the RMSE to the previously played frame is implemented. 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, a foreground-background segmentation algorithm of OpenCV (see [Ziv04; ZV06]) is applied to the frames. Focusing on the foreground object, the percentage of the moving area is used as a motion indicator in the *movement* feature. Similar to a video codec, a block motion estimation algorithm is performed – *blockmotion*, the used method is part of scikit-video. For the feature implementation, the SE3SS search method is used, moreover, 10% of the video height is defined as blocksize to speed up calculations. Moreover, after extraction of moving blocks, for all directions it is counted how often a moving block was identified [GRR19; Gör+19].

Similar to what is described in [MLS18], a more global view of motion may be required. To this aim, a sliding window of 60 frames is handled, such a window usually corresponds to about 1 second of a given video in UHD-1/4K. This window is then later handled as a cuboid, where several planes are sliced 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 analogous way for columns.

In contrast to videos with high motion, also other cases can occur. For example, some videos are quite static, and to cover them a staticness measure *staticness* has been included in the model framework. To estimate a value for *staticness* a mean frame based on all currently played frames is calculated. If the video is mostly static, the estimated mean frame includes a lot of spatial information. For this reason, as the final feature value, the SI measure of the current mean frame is used.

In addition to staticness of the video, the amount of noise within a given video frame as *noise* is also estimated. This feature uses a wavelet-based estimator for noise [DJ94].

To further analyze a given video, the number of scene cuts or shots may be important. The feature *scene_cuts* estimates the number of scene cuts. It uses a resized $360p$ view of the given video frames and performs a threshold-based detection for scene cuts, similar to the method implemented in scikit-video, see [OT94; ZMM95].

All features that have been described are so far classical no-reference features, thus a reference video is not required to perform the calculation. In the case of a full-reference model, the mentioned features can be applied to the distorted and reference video. Besides these individual distorted and reference feature values, also differences in the feature values are considered in the model pipeline.

4.1.1.2 Full-reference Features

To include typical full-reference aspects, the proposed framework further uses some traditional full-reference image metrics, namely PSNR, SSIM [Wan+04; WSB03] and VIF [SB06]. The approach here is similar to Netflix's VMAF. In the development stage, a higher number of full-reference metrics were included, however, there was no noticeable increase in performance, thus a limited set has been selected. In general, the framework allows for the addition of newer or other full-reference features. 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, a framerate estimation feature *fps_est* is included. It compares frames of the distorted and reference video in a sliding window of $w = 60$ frames, assuming that in the case of distorted lower fps, there are duplicated frames stored. Using RMSE of two consecutive processed frames for the distorted and references video as an indicator, a check for the given window how many duplicated frames are presented can be performed. The final estimated frames per second measure is calculated using Equation 4.1, 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, resulting in a not necessarily accurate estimation, this is compensated by the fact that as an overall feature later several statistics are used so that the feature *fps_est* becomes quite robust.

$$fps\_est(w) = |w| - |dis_0| + |ref_0| \tag{4.1}$$

4.1.1.3 Bitstream Features

To handle hybrid mode 0 models, additional bitstream or meta-data-based features are required. For this reason, meta-data of a given video file using ffprobe is extracted, in a real-world hybrid scenario, this meta-data would be accessible on the client and can be stored while playing the video. The most important meta-data are framerate, bitrate, video height and width (resolution), and the video codec used. Including these features, some additional values are calculate, starting with resolution as height times width, logarithm of resolution, *bits-per-pixel (bpp)*, see Equation 4.2, logarithm of bitrate and framerate, normalized values for framerate, see Equation 4.3, and resolution, see Equation 4.4. Here, the normalization is based on the maximum values for framerate and resolution in the UHD-1/4K scenario and can be motivated by having UHD-1/4K as a reference scenario.

Most of these additional feature values are inspired by P.1203, where similar calculations are performed in the mode 0 parametric model part [ITU17; Raa+17].

$$bpp = \frac{bitrate}{framerate \cdot resolution} \tag{4.2}$$

$$framerate\_norm = \frac{framerate}{60} \tag{4.3}$$

$$resolution\_norm = \frac{resolution}{2160 \cdot 3840} \tag{4.4}$$

The included bitstream features are limited to meta-data because they can be easily extracted in most applications.

4.1.2 Temporal Pooling of Feature Values

In the overall machine learning pipeline, several models can be trained for video quality prediction. Due to the fact that some of the introduced 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, the pipeline includes more than mean values as statistics as a 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 [GRR19; Gör+19; Rao+20b]. 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, it is converted to individual vectors and for each of the vectors, the following calculations are performed. For f the following values are calculated: 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 considering their feature value range. In addition, the values of f are split into $n = 3$ equidistant temporal groups, and for each group, the mean and standard deviation are calculated. With this method, for each feature in total 25 statistical values are extracted. All values are time-independent and are later fed into the used machine learning pipeline.

4.1.3 Speed up and Error Compensation

There are several ways to speed up the calculation of software in general. Besides vectorization or parallelization, that better utilizes modern hardware, 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 the case of 4:2:2-10bit UHD-1/4K uncompressed video, a frame has a size of $\approx$ 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 the following only the reduction of per-frame information is considered, to not interfere with temporal or motion-related properties of the video. The general idea is based on the approach presented in [GKR19], where a specific 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 [GKR19] that a center crop of $360p$ introduces only a rather small error compared to full-frame estimated VMAF-scores. A detailed description of this center-cropping approach is presented in Section 5.4. The introduced error was below the error that occurs when repeating the same subjective test at different labs [PW03]. Moreover, the

model instances of the described framework are able to compensate for some center-cropped errors due to the used machine learning model and use some more features than would be required.

4.1.4 Model Instances

Using the introduced general model framework, which includes various features, it is possible to create several model instances. Each specific example model instance has a different application scope, which will be highlighted in the following description. The model instances focus on pixel-based and hybrid models. For all models, as the default, a 360p center crop is used (applied to the distorted and reference video as a pre-processing step within the framework). In addition, an evaluation of larger crops and uncropped model variants (see Section 4.3.4) is described.

4.1.4.1 nofu – No-reference

The first model instance is a no-reference model, in the following referred to as **nofu**. It uses all **img**, **mov** and **img-nofu** features shown in Table 4.1. 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 the 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 [Rob+17]. Example applications include quality monitoring in case of live-streaming of broadcasting channels or streaming of gaming sessions. A reduced variant of **nofu** has been also applied successfully to predict gaming video quality [GRR19]. In the evaluation experiments conducted in this work it outperformed the VMAF model. For the considered case of gaming-video streaming prediction, a reduced feature set and a lightweight temporal pooling

method have been used, because gaming videos have different properties compared to the wider range of common videos.

4.1.4.2 hyfu – Hybrid No-reference

As another model instance based on the introduced features, a hybrid model is proposed, referred to as **hyfu**. **hyfu** uses all **img**, **mov** and bitstream **bs** features listed in Table 4.1. 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 the case of YouTube, Netflix, and Amazon Prime Video, it is possible to estimate the required meta-data based on the DASH manifest file.

4.1.4.3 fume – Full-reference

Especially in encoding optimization approaches, the source video is accessible and enables the application of full-reference video quality models. For this reason, a model called **fume** is introduced. It is based on all **img**, **mov**, **img-fr** and **mov-fr** features described in Table 4.1. **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 the 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 a 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.

4.1.4.4 hyfr– Hybrid Full-reference

As the last model instance, a hybrid full-reference model called **hyfr** has been developed. It includes all features (**img**, **mov**, **img-fr**, **mov-fr** and **bs**) that are listed in Table 4.1. **hyfr** can be applied to monitoring or encoding optimization tasks, especially in cases where also knowledge of the underlying bitstream is accessible or more precise meta-data. Especially to not fully focus the model on the used encoding schemes, it was decided to only include some basic meta-data-based features as bitstream features, as in **hyfu**.

4.1.4.5 Extensions

In the previous section, four model instances are introduced. All of them use the described architecture covering feature extraction, temporal pooling, and machine learning pipeline. However, further video quality models can be developed using the described 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 for the full-reference features, here with differences regarding these no-reference features used for the overall quality estimation. Also, other prediction targets or analyses can be performed, which is shown in more detail in Chapter 5. Moreover, additional bitstream-based features could be used to enable higher modes of hybrid model variants, for example, mode 3, similar to ITU-T P.1203.1 [ITU17] using QP values, or in addition using motion statistics as employed by ITU-T P.1204.3 [ITU19b; Rao+20b].

4.2 Subjective Video Quality Datasets

To train the proposed and presented video quality models, in total four subjective tests, which have been conducted at the author's lab, are used. These tests were part of the P.NATS Phase 2 competition that resulted in the ITU-T Rec. P.1204 series of standards [ITU19a; Raa+20]. In the remainder they will be referred to as the AVT-PNATS-UHD-1 dataset. The described model instances are further validated and evaluated using the superset of the publicly available dataset AVT-VQDB-UHD-

1 [Rao+19a]. This superset comprises additional source videos employed in the tests that cannot be shared publicly. 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 [Rec08] and ITU-R BT.500-13 [ITU14b]. A viewing distance of $1.5 \times H$ was used in all tests, with H being the height of the screen. The tests were conducted in a controlled lab environment following distances, lighting and other conditions according to ITU-T P.910 [Rec08] and ITU-R BT.500-13 [ITU14b], more details are presented in [Rao+19a]. The presentation and collection of ratings were performed using the AVRateNG [AVR][5] 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 the literature, most notably for developing ITU-T Recs. P.1203 and P.1204 [ITU17; ITU19a; Raa+20]. A brief description of the subjective tests follows, to understand the AVT-VQDB-UHD-1 dataset, and also an overview of the AVT-PNATS-UHD-1 dataset, that is used to train the models instances, is provided.

4.2.1 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, for more details of the overall constriction of the PNATS dataset see [Raa+20]. 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 are 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

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

video was never higher than the source framerate. For each encoding condition, one encoding bitrate from the range 100 kbps to 50000 kbps and one resolution between 360p and 2160p were selected and several such *HRCs* 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 four 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 four tests used a 55" LG OLED screen to present the videos to the participants.

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. Overall, two 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 two detected outliers. For the third test, 52 different sources were encoded with various *HRCs*, resulting in 185 different *PVSs* rated by 30 participants, with five 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, three outliers were detected for this test.

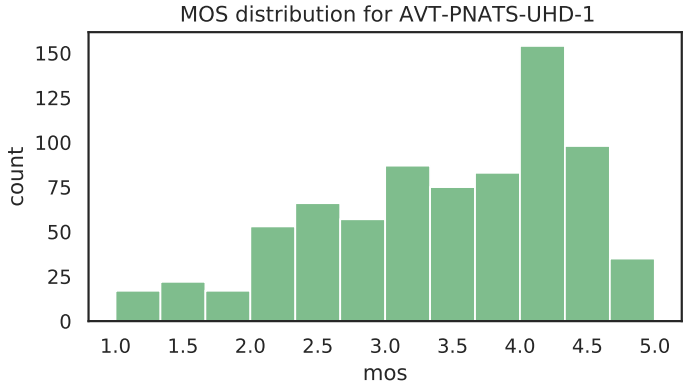


Figure 4.2: MOS distribution of all video quality tests used for training (AVT-PNATS-UHD-1).

The quality rating distribution of all the tests is as shown in Figure 4.2. 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

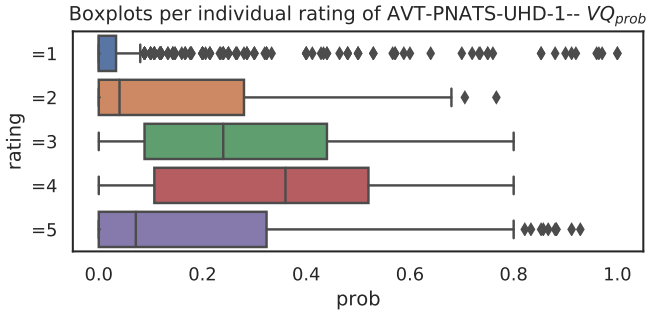


Figure 4.3: Boxplots of individual user ratings and the corresponding distribution for training (AVT-PNATS-UHD-1).

"'bad'" on the 5-point scale, e.g. with *MOS* values below 2.0. To further inspect the test subject's ratings for the AVT-PNATS-UHD-1 dataset, boxplots for each possible rating as depicted in Figure 4.3 are used. The mentioned probability refers to the VQ_{prop} problem formulation, see Section 2.2.6. Similar to the *MOS* distribution it can be concluded that high-quality ratings are the majority within this dataset.

4.2.2 Validation Dataset: AVT-VQDB-UHD-1

The publicly available AVT-VQDB-UHD-1 [Rao+19a][6] dataset including the sources that could not be shared as part of the original publication is used to validate and evaluate the proposed models. 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 subjective tests. It is noted that in the evaluation, due to processing issues, stimuli using the 10 s water_netflix sequence (this holds only for test_1) are excluded because the reference video and encoded segments are not aligned perfectly. All the source videos have a framerate of 60 fps. An overview of the videos is shown in Figure 4.4. A wide range of encoding conditions have been used in the tests, with resolutions ranging from 360p to 2160p, framerates between 15 fps and 60 fps and the encoding bitrates between 200 kbps, and 40000 kbps. In the following, each of the four subjective tests that make up the AVT-VQDB-UHD-1 dataset are briefly presented. A more detailed

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

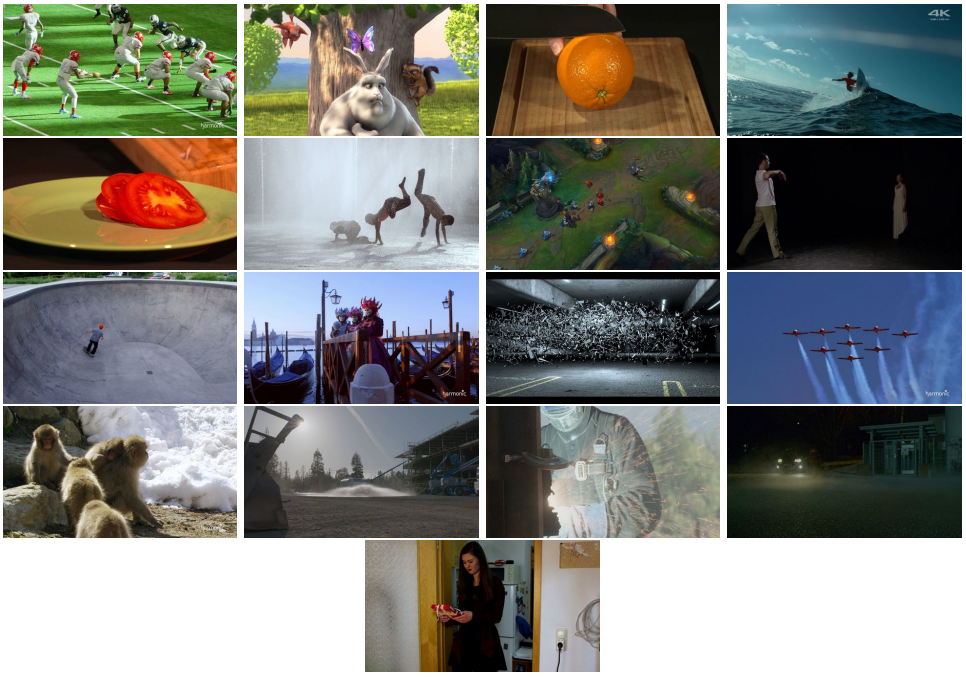


Figure 4.4: Thumbnails of source videos included in the AVT-VQDB-UHD-1.

description is included in [Rao+19a]. Like in the 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 the 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.

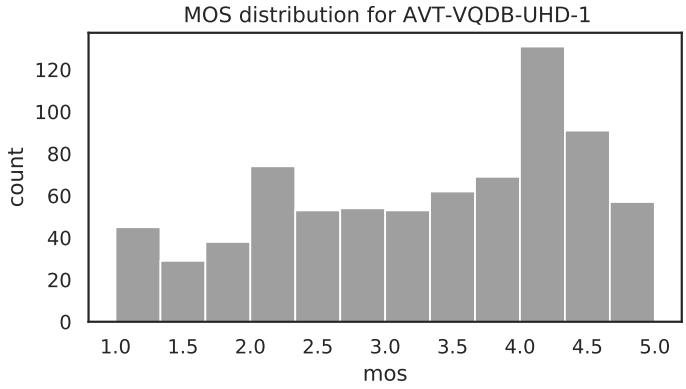


Figure 4.5: MOS distribution of all video quality tests used for validation (AVT-VQDB-UHD-1).

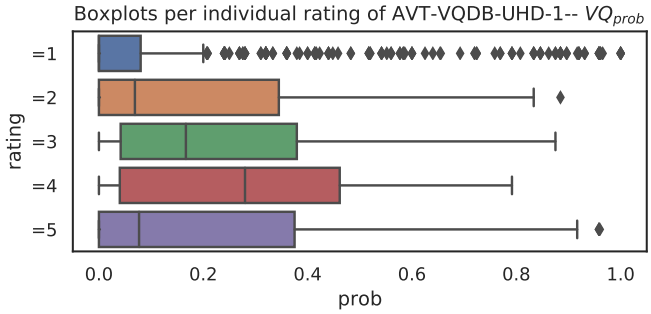


Figure 4.6: Boxplots of individual user ratings and the corresponding distribution for the dataset used for model validation (AVT-VQDB-UHD-1).

The quality rating distribution is as shown in Figure 4.5 for all four tests of 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 4.6, 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. In the following the four subjective tests are described in detail.

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, 360p, 720p, 1080p and 2160p. The videos were encoded using two different bitrates for resolutions from 360p and 720p resolutions and three different bitrates for resolutions of 1080p and 2160p. 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.

test_2 This test follows a *bpp* approach for the encoding conditions with four different *bpp* values used for the four different resolutions employed in the test. As in test_1, four different resolutions, namely, 360p, 720p, 1080p and 2160p 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 of 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.

test_3 Together with test_2, test_3 is the second of two tests forming a subset within the AVT-VQDB-UHD-1 dataset which follows a *bpp* approach to select the encoding settings. The same *bpp*-values and resolutions were used as in test_2 but with H.265 and VP9 as the 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.

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 360p and 2160p. 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.

4.3 Evaluation for Video Quality Prediction

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

Moreover, an in-depth analysis of how the proposed center cropping approach will affect the model performance will be performed. It is important to mention that training and validation do not have overlapping source videos. This enables a critical view of the performance of the proposed models because each of the models will be evaluated with unknown data.

For training, all 764 stimuli included in the AVT-PNATS-UHD-1 dataset are used. The validation is based on the videos of the publicly available database AVT-VQDB-UHD-1 [Rao+19a], with a total number of 756 stimuli. The trained models are part of the open-source software to enable reproducibility of the subsequent evaluation.

In the following, the performance will be evaluated. Thus for all models, first the classification problem is handled, 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, a $360p$ center cropping is used 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.

4.3.1 Classification Problem: VQ_{class}

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

In Figures 4.7, normalized confusion matrices for all models for the full validation data are shown. The best model clearly is **hyfr**, followed by **hyfu** and **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.

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

Table 4.2: Performance values for VQ_{class} for all models; sorted by tests and mcc, rounded to 3 decimal places.

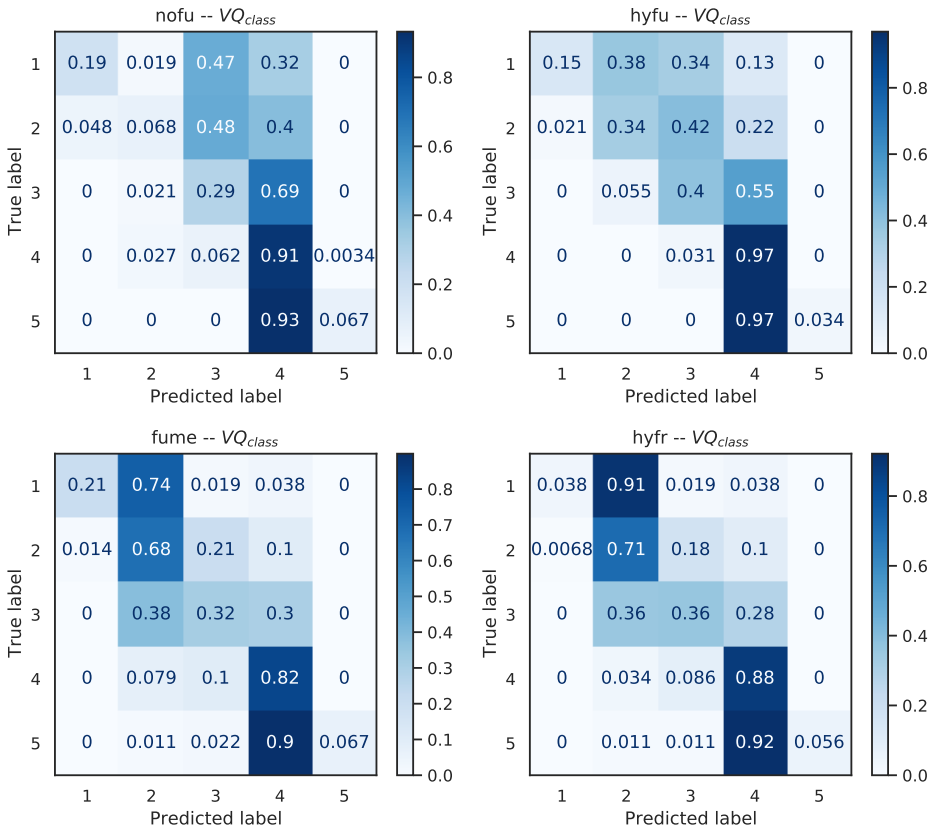


Figure 4.7: Confusion matrices for all models for VQ_{class}.

A detailed view of performance values per subjective test that are included in the AVT-VQDB-UHD-1 dataset is presented in Table 4.2. 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} seems to be more challenging. This can also be argued by the fact that the underlying video quality tests were targeted to cover video quality as mean opinion scores 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.

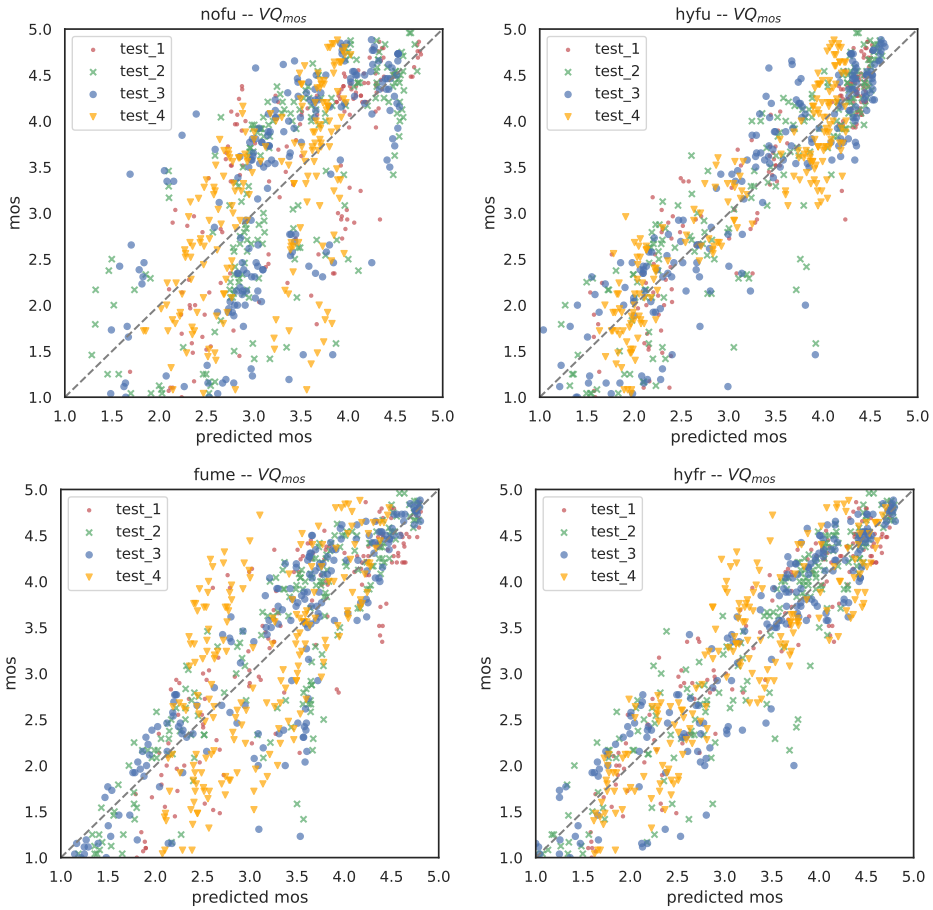


Figure 4.8: Scatter plots for all models for VQ_{mos}. For each subjective test a linear fit was performed.

4.3.2 Regression Problem: VQ_{mos}

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

In Figure 4.8, scatter plots for all four models are shown, and in Table 4.3 a detailed view for all tests. For both the scatter plots and Table 4.3, a linear fit of the predicted and ground truth ratings was performed, according to ITU-T P.1401 [ITU14a]. 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 **nofu** will perform better if it is specifically trained on the encoding and content type that is used for prediction.

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 4.3: 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.

It is already evaluated that such a specialized model in case of **nofu** for gaming videos [GRR19] performs better, here the performance of **nofu** was comparable to the performance of VMAF, which is shown in more detail in Section 5.2.1. Furthermore, it can be seen that the included mode 0 knowledge (bitrate, framerate, resolution) of the distorted video is a benefit for the developed models. In this case, the performance is increased e.g. from ≈ 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 the proposed models and because the usual video quality problem is handled as VQ_{prob}, it is possible to compare the gathered results with different state-of-the-art models.

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

Table 4.4: Performance values for VQ_{mos} for state-of-the-art models; calculated on full-frames; 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.

In Table 4.4, performance metrics for the AVT-VQDB-UHD-1 dataset for **VMAF**, **ADM2**, **MSSSIM**, **SSIM** and **PSNR** are shown. They have been calculated on the full-frame videos. Here only full-reference state-of-the-art models are considered 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 the training databases, only the unmodified versions of the models are considered, to enable reproducibility. Further, for the used objective model values that are included in the AVT-VQDB-UHD-1 dataset, 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 $\approx$ 0.81 across all tests, and a maximum value of $\approx$ 0.94 in case of test_1. In comparison to **VMAF**, the best performing model **hyfr** has a pearson correlation of $\approx$ 0.92 for all tests and as best $\approx$ 0.94 for test_1. So **VMAF** and **hyfr** have similar performance values, 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 variations. It can further be observed that the hybrid models predict the video quality for test_4 more precisely. However, comparing all of the 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. In general, **fume** and **VMAF** are both full-reference models using several atom features for the overall quality estimation. However in contrast to **VMAF**, **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** when considering all tests together, without having access to the source video. The worst performing model **nofu** has a similar performance as **PSNR** for 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 4.10. 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 relatively good.

4.3.3 Multi-output Regression Problem: VQ_{prob}

Besides the prediction problems formulated as classification VQ_{class} and regression problems VQ_{mos}, respectively, the multi-output regression problem VQ_{prob} was further introduced. 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.

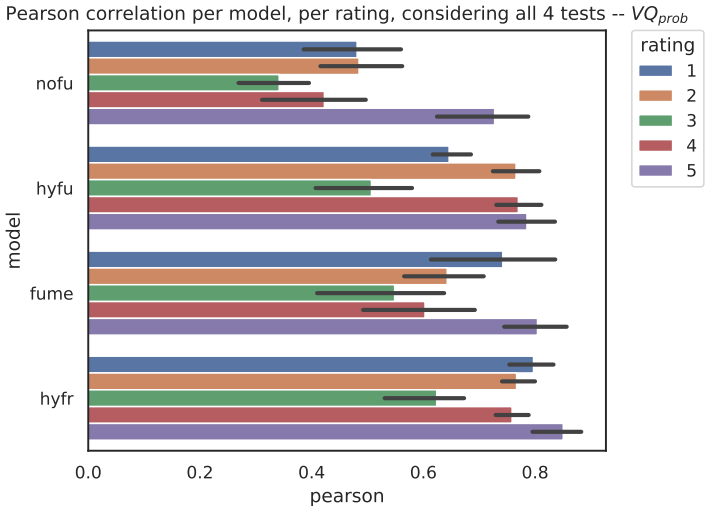


Figure 4.9: Performance across all tests in case of VQ_{prob} considering all four models, with 95% confidence intervals.

In Figure 4.9, 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 the 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 4.5. For each rating target r, Pearson, Kendall, and Spearman correlation values with regard to the ground truth data are included. 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

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

Table 4.5: Mean performance values for VQ_{prob} for all tests; sorted by rating and Pearson, rounded to 3 decimal places.

Figure 4.9, 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.

4.3.4 Center Crop Evaluation

As mentioned in Section 4.1.3 and 4.1.4 all own model instances, namely **nofu**,**fume**, **hyfu**, and **hyfr**, use a center cropped version of the input videos to calculate features. This approach is similar to the **cencro** approach proposed in [GKR19], and which is presented in more detail in Section 5.4. However, in that previous work,

several full-reference models were applied on full-frames, and an additional evaluation using center-cropped frames was performed. Here, a further evaluation of the proposed center cropping approach and its impact on the performance and feature calculation speed is required. In total, five different center cropping settings namely 240p, 360p, 720p, 1080p, and 2160p have been selected, where the last setting refers to the full-frame, thus no center cropping being used. For each of the center cropping settings, all four models are trained with the training dataset described in Section 4.2. In Figure 4.10, the performance values for all models and cropping settings are shown, considering 10-fold-cross validation of the employed training data. A separate evaluation with the validation dataset is skipped because it will show similar performance values. In total 32 training repetitions are performed. In this part, the only focus is 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 4.10: 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 360p center crop for **nofu** results in a Pearson correlation value of around 0.73, whereas the center crop setting of 720p improves it to 0.75, 1080$p \approx 0.76$ and 2160p results in 0.76. The worst performance of around 0.70 is for the case of a 240p 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 it was decided to also use a 360p center crop for **nofu**, even if the performance is slightly lower than for a 720p center crop value, the difference in Pearson correlation is of 0.75 vs. 0.73.

The processing time is an important factor in addition to the overall prediction performance of all models considering the used center cropping parameter. For this reason, the overall model prediction time is measured, 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 the introduced model instances.

Here, one video sequence (american footbal, 360p resolution target encoding resolution, bitrate = 200 kbit/s, video codec vp9) was selected as a test sequence, and the overall processing time of all center crop variants was measured for 32 repetitions, where each run removes all cached files of the previously performed run. Different videos will end up with a 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 will be shown in Section 5.4. Here, it should be mentioned that all of the 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, which is not employed in this evaluation. All measurements were performed on the same computer, with an Intel Core i7-9700 CPU (3.00 GHz) with 64 GB of main memory and local file access using an SSD.

In Figure 4.11 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 4.6. For example, the **hyfu** model requires about 70 s for 240p and $\approx$ 2466 s for 2160p. Thus 9 times the center cropping height results in about $\approx$ 35 times the processing time. The other models behave similarly across several center crop values. In general the full-reference models need about

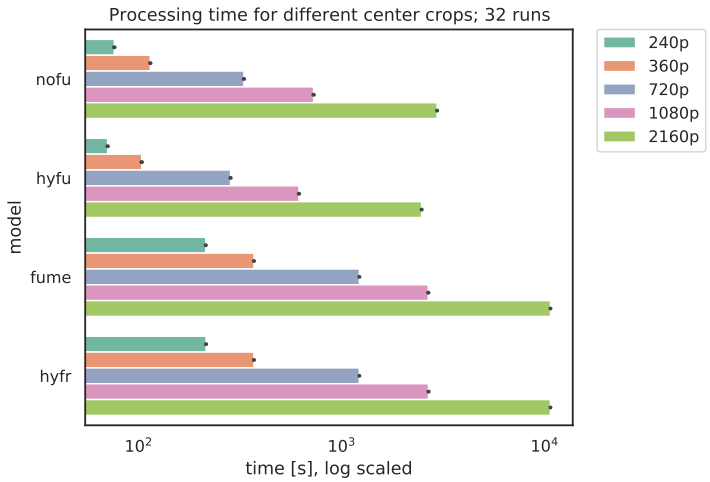


Figure 4.11: Overall processing time for quality prediction considering different center cropping values. Shown are mean values and 95% confidence intervals across 32 repetitions each.

3-4 times the processing time, e.g. for **hyfr** in case of 720p it takes around 1216 s, compared to $\approx$ 282 s for **hyfu**.

Considering the speed up that can be achieved using a center crop and the negligible performance reduction for most of the models (except **nofu**), a center crop setting of 360p has been selected as the best trade-off between speed and prediction performance. This conclusion is in line with the results of other full-reference models [GKR19], where a 360p center crop was able to speed up calculation time significantly, while still preserving the high prediction accuracy of the models.

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

Table 4.6: Mean processing time [s] for each model for different center crop settings; values are rounded to integers.

4.4 Result Discussion

A framework has been introduced for video quality prediction and furthermore, four different model instances are described 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 it seems to be harder for the proposed models to achieve good performance, in comparison to other task formulations. The main reason for this is that for such a formulation a more uniformly distributed training dataset is required. A more suitable training dataset could, e.g., 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, the model **nofu** has 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 [GRR19], which is shown in Section 5.2.1. 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 the second prediction target, the focus was on the commonly used problem formulation, namely video quality as a single continuous score VQ_{mos}, as a regression problem. Here, it is shown that three of the 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 comparable performance to PSNR and SSIM, which are also commonly used video quality models. In addition, the evaluation shows, that the defined features are suitable for the prediction tasks.

As the last prediction target, the video quality task is handled as a multi-output regression problem VQ_{prob}, where several models are trained to 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 from the knowledge of the other ratings, thus further analysis is required.

In addition to the three different video quality prediction variants, the used center cropping approach is evaluated. Center cropping enables to speed up the 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 [PW03]. Only the model **nofu** could benefit from a larger used center crop. However, it was decided to use a $360p$ center crop for this model, too, to have a unified model architecture. Besides the model performance, also the required processing time is evaluated, and it can be seen that there is a huge cpu-time saving when center-cropping is used, which confirms and extends the observations in [GKR19].

4.5 Summary

In general, there are only a few video quality models available and even fewer are specifically trained for UHD-1/4K video content. Moreover, 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, a general video quality modeling pipeline was introduced. The overall framework consisting of features, pooling methods, and machine learning models is made available as open-source projects. The 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, two further approaches are described, 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, four different video quality models are instantiated. All trained models 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 them, a hybrid model extension is proposed, namely **hyfu** and **hyfr**. Both hybrid models incorporate additional video metadata about the codec used, resolution, bitrate, and framerate. Such meta-data is typically accessible during play out of a given video, while other bitstream related data requires specifically designed extractors.

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

Promising extensions of the 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 [Rao+20b; Raa+20]. 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.

The introduced pipeline can even be used for different types of video analyses, ranging from video classification, genre classification for games, gaming video quality prediction, to encoding parameter estimation, or using the center cropping approach for other models. Examples of such extensions are described in the next Chapter 5 “Other Applications of the Model Pipeline”.

Chapter 5

Other Applications of the Model Pipeline

The previously introduced video quality modeling architecture is not limited to the prediction of video quality alone. In the following, other video-related aspects and problems are introduced. Using variants of the general architecture, several machine learning models have been trained and evaluated for the specific problem. Such problem instances are for example source video classification based on the native resolution, gaming video quality, genre prediction for gaming videos, the estimation of specific encoding parameters, and approaches to speed up full-reference video quality calculations. Each of the problem instances requires a different training and validation dataset that is briefly described.

The following questions will be answered in the Chapter:

- ▷ Can the proposed machine learning pipeline and features be applied to other video-related questions? (**Research Question** 4)
- ▷ Is it possible to automatically classify videos that have a benefit of using UHD-1/4K resolution? (**Research Question** 3)
- ▷ How can the processing time of state-of-the-art video quality prediction be reduced? (**Research Question** 2)

The chapter is mostly based on the following publications:

[Gör+19] **Steve Göring**, Julian Zebelein, Simon Wedel, Dominik Keller, and Alexander Raake. "Analyze And Predict the Perceptibility of UHD Video Contents". In: *Electronic Imaging, Human Vision Electronic Imaging* 2019.12 (2019)

[Gör+20] **Steve Göring**, Robert Steger, Rakesh Ramachandra Rao Rao, and Alexander Raake. "Automated Genre Classification for Gaming Videos". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6

[GRR19] **Steve Göring**, Rakesh Rao Ramachandra Rao, and Alexander Raake. "nofu - A Lightweight No-Reference Pixel Based Video Quality Model for Gaming Content". In: *Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*. Berlin, Germany, June 2019

[GRR20] **Steve Göring**, Rakesh Rao Ramachandra Rao, and Alexander Raake. "Prenc – Predict Number Of Video Encoding Passes With Machine Learning". In: *Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. Athlone, Ireland, May 2020

[GKR19] **Steve Göring**, Christopher Krämmer, and Alexander Raake. "cencro – Speedup of Video Quality Calculation using Center Cropping". In: *21st IEEE International Symposium on Multimedia (IEEE ISM)*. Dec. 2019, pp. 1–8

5.1 Source Video Classification for UHD-1/4K

In several studies, it has been shown that it can be hard for users to identify videos using UHD-1/4K or Full-HD resolutions. In [Ber+15], Berger et al. use compressed videos and performed a subjective evaluation for video quality using the ACR rating scheme. Results show that while some videos may have a clear benefit of using UHD-1/4K resolution and some do not. Additional subjective tests indicate similar results for the visual experience considering UHD-2/8K as compared to UHD-1/4K [SZM20]. However, the overall experience of UHD-type content is usually a combination of a reduced viewing distance, the content itself, the quality of the source video, the used video encoding settings, and user's expectations. Moreover, also the used test method has an influence on the results. For this reason in [Gör+19] it is analyzed whether the user can perceive a difference of videos using UHD-1/4K and Full-HD resolution for uncompressed videos. In total, two subjective tests have been carried out using two different test methods. The overall idea is to train an automated system to identify source videos that show a benefit in terms of the visual

experience using UHD-1/4K resolution . Such a system could help to reduce video streaming bandwidth, for example in the case that a video shows no benefit of the higher resolution. Further, the system can be used to verify that source videos are in their native UHD-1/4K resolution.

5.1.1 Conducted Subjective Tests

In total, 10 uncompressed UHD-1/4K videos with chroma sub-sampling 4:2:2, 10 bits video depth and 10 s duration have been used in two different subjective tests. The video contents are selected using SI and TI according to ITU-T Recommendation P.910 [ITU08b] covering a wide range of realistic video scenes. The calculation was performed using the publicly available implementation for SI/TI[1]. The UHD-1/4K video sequences have been down-sampled to a lower resolution and afterward up-sampled to UHD-1/4K again using the Lanczos-3 algorithm. The Lanczos-3 algorithm was selected because it shows the best quality for up-scaling according to [Li+14]. Several resolution pairs have been considered in the subjective evaluation experiment, i.e., UHD-1/4K vs. Full-HD, UHD-1/4K vs. 900p, and UHD-1/4K vs. 720p. The stimuli were presented with a Panasonic 65 " screen in a lab-based setting following ITU-R BT.500-13 [ITU14b], with a viewing distance of 1.5 · screen height. The video sources consist of 10 videos from harmonic [Har], one from big buck bunny [Blea], one from Bennu [NAS17] and 8 recorded sequences.

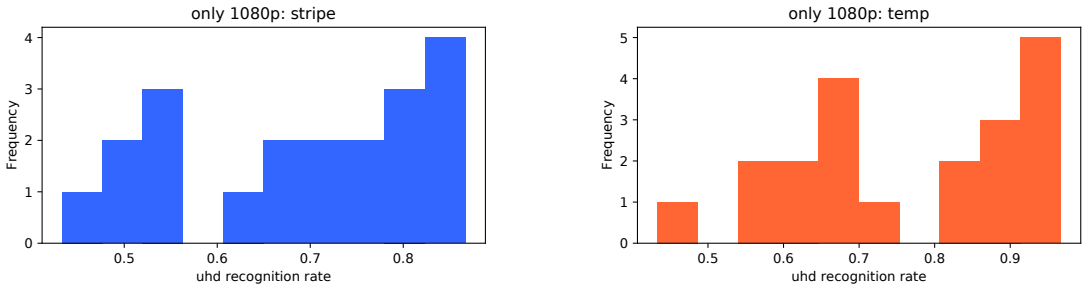
To enable a direct comparison of the UHD-1/4K and lower resolution, an *ACR* approach similar to Berger et al. [Ber+15] is not necessarily the best approach, because participants may not be able to remember the original high-resolution videos. On the other side in [Li+14; Van+16], a one stripe method is used for such an evaluation. Here, the video signal is split in the middle into two separate views, e.g., on the left is the re-scaled Full-HD version, and on the right side is the UHD-1/4K version of the video. Because such an approach may be limited to the middle part of the video, this method was extended to a multi-stripe method using in total 12 stripes with a color-coding (A or B) scheme. This method is referred to in the following as *STRIPES*. Besides other possible methods, that have been evaluated in pre-tests, a temporal changing of both representations showed promising results. In the temporal switch

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

method *TEMP* the quality levels are switched periodically. The same color scheme as for the *STRIPES* method was used. Every two seconds the video representation either UHD-1/4K or lower resolution was changed. The *TEMP* method is a specialized version of the ITU-R BT.500-13 [ITU14b] method for adaptive content switching in a subjective test without the possibility to manually change the stimuli. After stimulus presentation using AVRateNG[2], the participant was asked to judge which of the two versions is of higher quality (A vs. B). For all stimuli, two variants, namely the (UHD-1/4K, lower resolution) and (lower resolution, UHD-1/4K) were included in the test, which avoids a learning effect of the used color scheme. In total two subjective tests were conducted, one for the *STRIPES* and one for the *TEMP* method. Overall 60 participants took part in both tests.

5.1.2 Analysis of Results

The two conducted tests showed that users can clearly distinguish UHD-1/4K and 720p. The 900p case was more challenging for the participants, however still manageable. Most interesting is the comparison of UHD-1/4K and Full-HD to evaluate whether there is a benefit of this higher resolution for certain source videos.



(a) *STRIPES* **(b)** *TEMP*

Figure 5.1: Histograms for UHD-recognition rate for both used test methods for the 1080p case.

As a result, a deeper analysis for the UHD-1/4K versus Full-HD case was performed. In Figure 5.1 for both methods, namely *STRIPES* and *TEMP*, histograms for the UHD-recognition rate are shown for the UHD-1/4K versus Full-HD comparison. The UHD-recognition rate is the number of cases where UHD-1/4K was correctly identified

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

as the highest quality in relation to all shown videos. It is further important to mention that each pair (UHD-1/4K, Full-HD) and (Full-HD, UHD-1/4K) is handled separately in the plot, resulting in overall 20 video comparisons. Both methods show that there are source videos where the differentiation is clearly possible. For example, assuming that a UHD-1/4K content is identifiable with a recognition rate of 80%, then it can be stated that about 10 videos are correctly recognized for the *TEMP* method and 7 videos in case of *STRIPES*. Important to mention here is that the *TEMP* method is similar to typical *DASH* segment transitions, where on the other side the *STRIPES* method is probably more unfamiliar for participants. To sum up, both methods seem to be valid approaches to compare the benefits of higher resolution videos. A detailed analysis of the corresponding videos indicated that there is no direct connection between the spatial plus temporal complexity and the identifiability of the UHD-1/4K version. To conclude, in total around 7-10 out of 20 video comparisons show a benefit of using a UHD-1/4K version in contrast to Full-HD thus there is a visible difference for some source videos. A more detailed description of the conducted tests and results can be found in [Gör+19].

5.1.3 Prediction Model

The introduced problem of identifying videos where users have a benefit in UHD-1/4K resolution is a typical binary classification problem. In the following, it is handled as a pixel-based no-reference problem similar to **nofu**. However, instead of a continuous value as prediction, a classification is performed, see [Gör+19].

For this classification the subjective dataset has been transformed using a threshold for the calculated UHD-1/4K recognition rate of 80%, based on the ratings of the *TEMP* method. Thus, when a video has a UHD-1/4K recognition rate above the threshold it is labeled as $class = 1$ otherwise $class = 0$ (no benefit of UHD-1/4K over Full-HD). In the following, the constructed dataset is referred to as *PER*, because it is based on the perception test.

For the feature estimation, a subset and extension of the features listed in Table 4.1 is used. The used features are namely *contrast*, *blur*, *fft*, *si*, *colorfulness*, *tone*, *saturation*, *uhdhdsim*, *ti*, *temporal*, *blockmotion*, *movement*, and *staticness* with *niqe* [MSB13] as an additional feature. After the features are extracted for the given videos, a temporal

pooling of the feature values is performed. This pooling method is based on the approach introduced in Section 4.1.2 and slightly modified. The modification is that instead of $n = 3$ temporal groups 5 are used because $n = 5$ temporal groups showed better prediction results. In contrast to the **nofu** model, the feature set is slightly changed, the temporal pooling adapted and most importantly no center cropping was performed. The reason to skip center cropping is that some properties of the videos are not visible in the center region of the frames.

Because the subjective dataset *PER* is limited in size, a synthetic dataset namely *SYN* has been created. The *SYN* dataset covers 36 different source videos that are not included in the *PER* data. Each of the videos has been down-scaled to Full-HD and re-upscaled to UHD-1/4K, leading to 72 videos in total. Videos with a native Full-HD resolution are handled as $class = 0$ and videos with UHD-1/4K as $class = 1$. The prediction target was to identify whether such a re-scaling has been performed or not. Based on the dataset a random forest model using 10 trees and a feature selection criterion of $0.5 \cdot mean$ was trained. The random forest model showed promising results using this approach in a 10-fold cross-evaluation scenario, compare Table 5.1.

Table 5.1: Classification results for the *SYN* dataset using 10-fold cross-validation.

class	precision	recall	f1-score	support
0	0.77	0.92	0.84	36
1	0.90	0.72	0.80	36
avg / total	0.83	0.82	0.82	72

The overall prediction system has an f1-score of approximately 0.82. This means that the system is able to correctly predict approximately 80% of all cases. This result is similar to the results from the subjective dataset *PER*. To check the wrongly classified cases a confusion matrix is used.

In Figure 5.2 the confusion matrix for the *SYN* evaluation is shown. It is visible that most of the wrongly classified videos are originally $class = 1$, which indicates some limitations of the prediction system. However, the synthetic dataset *SYN* does not include any perception-based values and is just used to evaluate whether the features and machine learning pipeline are usable.

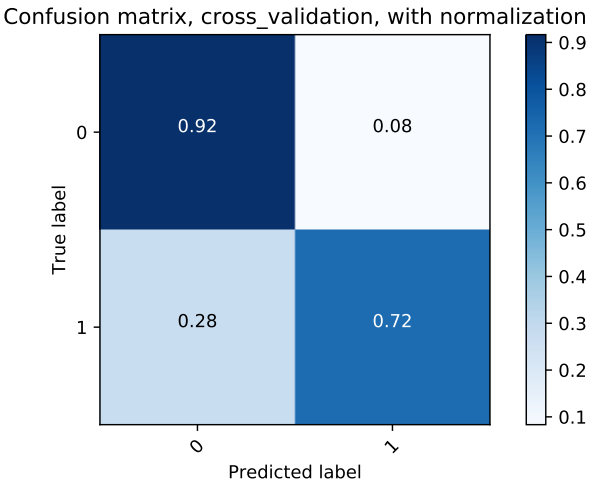


Figure 5.2: Confusion matrix of the prediction system for *SYN*.

For this reason, a second machine learning model has been trained (same parameters as before) using the *PER* dataset with 10-fold cross-validation. In this scenario, the training target was whether a video is identified correctly as UHD-1/4K or not by the subjects.

Table 5.2: Classification results for the experiment with the data from the perception test *PER*.

class	precision	recall	f1-score	support
0	1.00	0.30	0.46	10
1	0.59	1.00	0.74	10
avg / total	0.79	0.65	0.60	20

Results for the evaluation using *PER* are summarized in Table 5.2. Here, the f1-score of the final model was around 0.60 indicating that the underlying task seems to be challenging for the automated system. However, it should be noted that even though the prediction for *PER* is not outstanding, still, such a system could be used as a filter criterion for suitable UHD-1/4K content. Moreover, it is hard to train a prediction model using the gathered subjective data because only 10 source videos have been used in total. To circumvent the limitations of the dataset the synthetic dataset *SYN* has been used, however even such an approach is limited, because perceptual factors are not included in the synthetic dataset. In general, a larger dataset including subjective data would be required to reach a more robust prediction model.

5.2 Gaming Video Quality and Genre Prediction

The focus of the aforementioned video quality prediction and video classification tasks was classical video content such as movies, TV series, or user-generated content, that can be streamed using Netflix or Youtube. The recent development in video-on-demand streaming indicates newer use cases for video streaming, e.g., video gaming sessions or tournaments [Pan] are streamed around the world and followed by millions of users on platforms like Twitch[3] or Youtube Gaming[4]. Usually such gaming content has specific properties that are not necessarily similar to traditional video content. Such differences are for example that the content is computer-generated, the motion patterns are different (due to the underlying game mechanics) and the encoder settings differ because the content may be streamed live and only a minor performance decrease for the recording is accepted by gamers.

5.2.1 Gaming Video Quality Prediction

To tune or monitor video quality in the case of gaming content it is crucial to have appropriate models with different properties, such as being fast, no-reference based, and specifically trained for the gaming scenario. According to this criteria, **nofu** is the best fitting candidate of the introduced models and can be used even for gaming content.

For this reason, a modified version of **nofu** has been proposed, compare [GRR19]. The modifications include a subset of the listed features in Table 4.1 namely namely *fft*, *ti*, *si*, *blockiness*, *blockmotion*, *staticness*, *cubrow-0*, *cubrow-1.0*, *cubcol-0* and *cubcol-1.0*. In contrast to **nofu**, this further means that the *brisque* feature is not used, mostly because the calculation of this feature is more time-consuming. Similar to the general **nofu** pipeline, temporal pooling has been applied. Here, the modifications cover that only mean value, standard deviation, the first value, and for the $n = 3$ temporal groups mean values and standard deviations are calculated. Pre-tests indicated that this reduction is not affecting the model performance and decreases the required computation time. Additionally, a $360p$ center crop has been used to speed up the

[3] https://www.twitch.tv/
[4] https://www.youtube.com/gaming

feature calculation even more. In the following, the modified model is referred to as **nofu-gaming**.

Table 5.3: Performance values for **nofu-gaming**, VMAF predictions; 576 videos

model	pearson	kendall	spearman	RMSE
nofu-gaming	0.96	0.82	0.95	0.22
brisque+niqe	0.94	0.80	0.94	0.24
PSNR	0.87	0.68	0.87	28.58
SSIM	0.71	0.55	0.74	2.31
STRRED	-0.53	-0.42	-0.61	151.44
SpeedQA	-0.55	-0.45	-0.63	446.75

The model shows promising results in two 10-fold-cross-validation setups. For example, the **nofu-gaming** has a Pearson correlation of 0.96 for VMAF score prediction in case of the GamingVideoSET [Bar+18b], see Table 5.3. **nofu-gaming** outperforms other state-of-the-art models and in addition a re-trained **brisque+niqe** model.

Table 5.4: Performance values for **nofu-gaming**, *MOS* predictions; 90 videos

model	pearson	kendall	spearman	RMSE
nofu-gaming	0.91	0.75	0.91	0.42
brisque+miqe	0.89	0.73	0.90	0.44
VMAF	0.86	0.69	0.86	0.64
SSIM	0.79	0.61	0.80	2.03
PSNR	0.74	0.57	0.74	29.37
SpeedQA	-0.71	-0.56	-0.74	488.83
STRRED	-0.72	-0.55	-0.74	160.48

Furthermore, using the subjective scores included in the GarmingVideoSET [Bar+18b] a second evaluation was performed. Only a subset (90 out of 576 videos) of the GarmingVideoSET has subjective annotations. The retraining and 10-fold cross-evaluation resulted in a Person correlation value of 0.91 that is better or similar to state-of-the-art full-reference metrics, compare Table 5.4. Scatterplots for the top-3 models are shown in Figure 5.3 For example, **nofu-gaming** outperforms VMAF that has been showing good results in the gaming domain before [Bar+18b]. In addition, **nofu-gaming** shows a better prediction performance compared to a re-trained no-reference **brisque+niqe** variant.

Overall it can be concluded, there is a need for specialized no-reference models, and even though the performance of **nofu** model for the general video quality

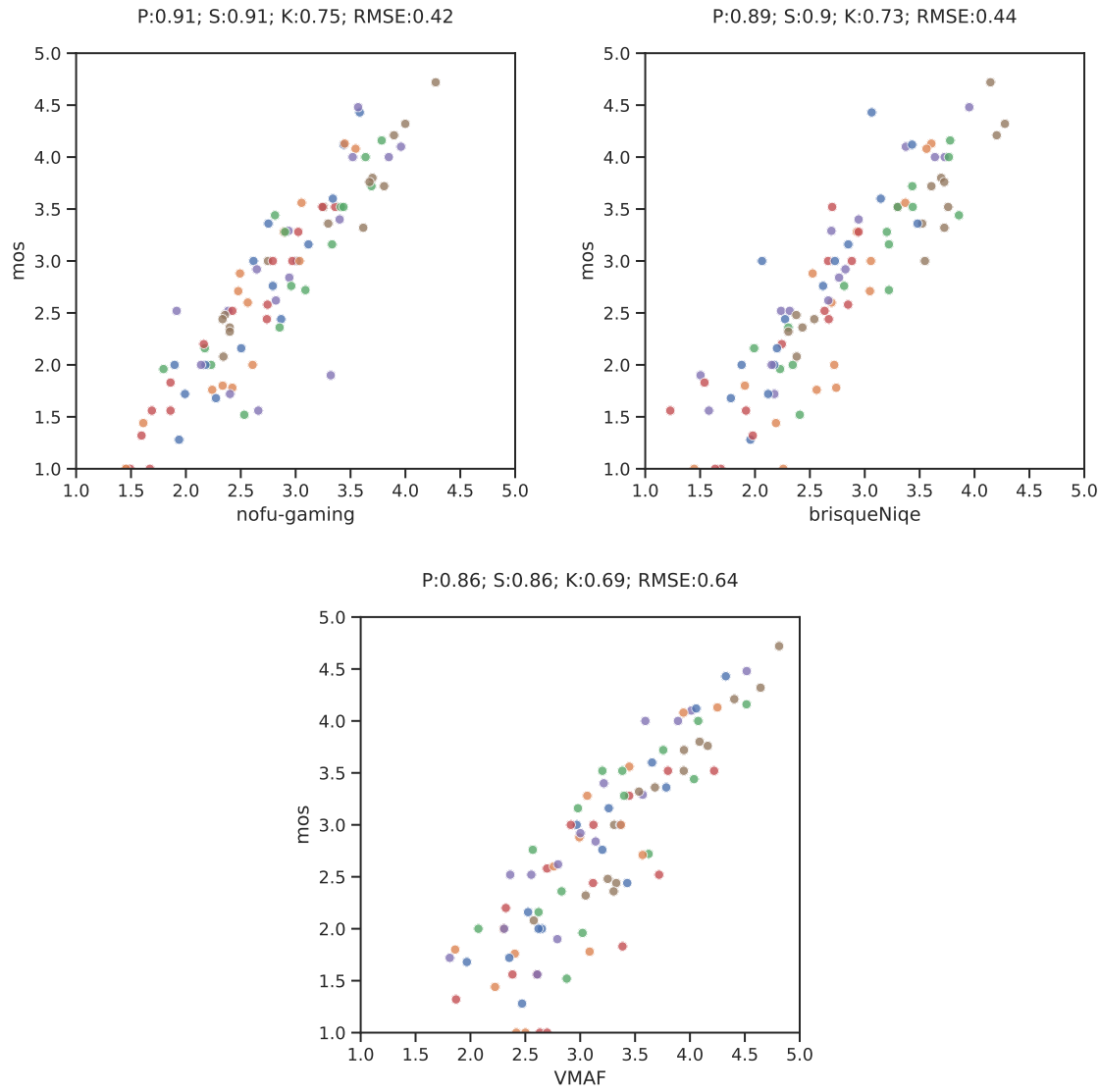


Figure 5.3: Scatter plots for top-3 models, colors corresponds to different source videos

task was not optimal, it shows to have other suitable applications as shown by the **nofu-gaming** variant. Furthermore, other models, e.g., DEMI [Zad+20a] or P.1204.3 [Rao+20a] covering traditional and gaming content have been proposed and are indicating good results.

5.2.2 Genre Classification for Gaming Videos

One important aspect of gaming videos is the used game and the associated gaming genre. Knowing whether the streamed game is, for example, a first-person shooter or a platformer has an impact on the encoding settings and overall quality.

The estimation of a specific gaming genre is not simple, because there are various genres available [App06]. Training an automated system to classify a given gaming video according to the gaming genre requires a specific dataset. The popularity listing of the Twitch platform [Twi] gives an overview of possible gaming genres. In total, six different gaming genres have been identified for the construction of the dataset. These genres are *first-person shooter (fps*)*, *jump'n run (jnr)*, *adventure/roleplay (rpg)*, *real-time strategy (rts)*, *top-down roleplay (tdr)* and *third-person shooter (tps)*.

For each genre, at least five different games are included in the dataset. Moreover, for each game, at least three streams from three different streamers have been downloaded from Twitch. In total the dataset comprises 351 downloaded videos, each with a duration of about 50 s. For all videos, only the highest available resolution has been downloaded. This ensures that the model has access to the best possible quality because such a genre classification would be performed before encoding and would be independent of visual quality.

Similar to the gaming video quality prediction, a machine learning classification model has been developed using only no-reference features. The used features are *blur*, *colorfulness*, *contrast*, *fft*, *si*, *ti*, *blockmotion*, *staticness* and *motiontracking* plus *cameramovement*. Except for the last two features, all others are taken from the general video quality modeling pipeline, see Section 4.1.1. Both new features cover motion-related aspects of games and are specifically developed for the genre classification task, see [Gör+20]. In this case, the center cropping approach has not been applied and the unmodified temporal pooling method has been used. In a first evaluation experiment, 10-fold cross-validation and several machine learning models have been used. The goal was to evaluate the corresponding hyperparameters and suitability of machine learning models. The following models namely, random forest, gradient boosting, support vector machines, or k-nearest neighbors were evaluated. The most promising results were shown by the *RF* and *gradient boosting classifier (GBC)*.

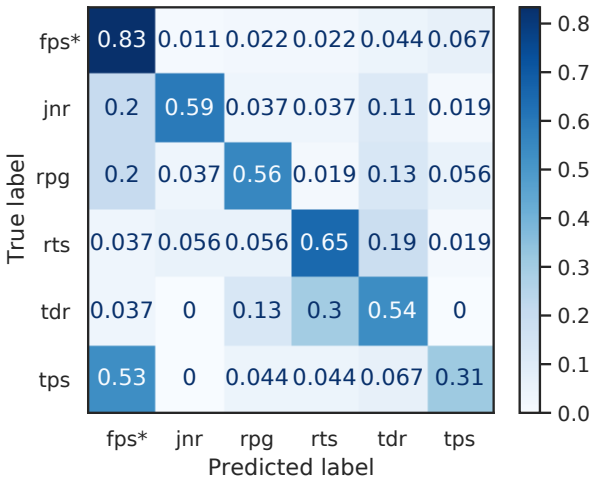


Figure 5.4: Confusion matrix for one of the best performing models, rf with $FS(0)$ and 100 trees. Values are normalized, in total 351 are used.

An example of a random forest model with no feature reduction and 100 trees is presented in the corresponding confusion matrix shown in Figure 5.4. It can be seen that most game genres are correctly classified. However for the pairs (*fps**, *tps*) and (*tdr*, *rts*) in some cases the prediction is incorrect. In case of (*fps**, *tps*), first and third person shooter, around 53% of the *tps* are classified as *fps**. One reason for this misclassification is that *fps** and *tps* games are quite similar regarding their camera movement and motion types. In the case of a *tps*, the view in the game is just based on a third person's view in contrast to the *fps** game. Furthermore, both game genres usually share similar properties, there are even games that can be switched from third-person to a first-person view. Moreover, the genre pair (*tdr*, *rts*), top-down role-play, and real-time strategy, behaves similarly to the aforementioned misclassification pair. Here, also similar camera perspectives and motion types are part of the game.

For this reason, these two mentioned genre pairs (*tdr*, *rts*) could also be joined to form a new meta-class. In this case, *mcc* will change to ≈ 0.64, f1-score to ≈ 0.73, *acc* to ≈ 0.74, *prec* to ≈ 0.73 and *rec* to ≈ 0.74. Thus, as expected, the overall performance is improved with lesser genre classes, while the performance is still comparable to the initially used 6 genres.

In the following evaluation, the dataset is split into 50% training and 50% validation parts. The idea is to further evaluate how well the models consider unknown videos.

To implement the 50%-50% split, 50% of the videos are used per genre for training and validation are sampled so that there are no overlapping streamers. Results of this evaluation are summarized in Table 5.5.

Table 5.5: Results of 50%-50% split evaluation for *RF* and *GBC* models, mean values considering 64 repetitions, sorted by *mcc*.

model	trees	FS(c)	*f1*	*acc*	*prec*	*rec*	*mcc*
RF	150	0.5	0.422	0.449	0.441	0.449	0.329
RF	100	0	0.408	0.444	0.424	0.444	0.324
RF	150	0	0.409	0.443	0.425	0.443	0.322
RF	100	0.5	0.416	0.444	0.433	0.444	0.322
RF	150	1	0.416	0.442	0.436	0.442	0.320
RF	100	1	0.409	0.435	0.427	0.435	0.310
GBC	150	1	0.371	0.387	0.371	0.387	0.251
GBC	100	1	0.366	0.381	0.366	0.381	0.245
GBC	150	0.5	0.349	0.364	0.348	0.364	0.223
GBC	100	0.5	0.344	0.359	0.341	0.359	0.216
GBC	150	0	0.319	0.335	0.315	0.335	0.184
GBC	100	0	0.318	0.333	0.314	0.333	0.183

Only the *RF* and *GBC* models were evaluated, with various hyperparameters. The number of trees was either 100 or 150. And for the feature selection criterion $FS(c)$ the threshold was varied with values $c \in [0, 0.5, 1]$, referring to $c \cdot mean$ importance in the introduced machine learning pipeline.

For each model, in total 64 repetitions were performed, and mean performance values are reported in Table 5.5. Similar to the previous evaluation, the *RF* model performs best, however, the overall performance is lower than in the 10-fold cross-validation. The reason for the general performance drop is clearly that the overall training and validation split was streamer-based. Here, it is important to know that even games with the same genre can show huge differences, for example, Overwatch and Battlefield 5 (BF5) are both first-person shooters. Still, Overwatch is colorful while BF5 is not. Furthermore, this is also the case for other genres, e.g., Minecraft and WorldOfWarcraft that are both role-play games that are quite dissimilar considering their graphics. In general, the performance values are still indicating that the features work even for unknown videos. For this reason, it can be concluded that *RF* models seem to be more robust considering new videos, while *GBC* models seem to have more problems for such a prediction.

Even though the prediction models wrongly classified some gaming genres, the analysis showed that in these cases the games themselves are quite similar, considering e.g. the camera motion and motion tracking. Hence, they can be considered of similar complexity for example for encoding or quality prediction, this possibly justifies the merging of the respective classes. In general genre classification could be included in video quality prediction models or can be used to optimize encoding settings.

5.3 Encoding Parameter Estimation

Bitstream-based video quality models show promising results considering prediction performance. One example of such models is the ITU-T P.1204.3 model [Rao+20b; ITU19b] that is based on bitstream statistics such as QP values and motion vectors. For the development of bitstream models, the diversity of possible encoding parameters can be challenging. One crucial and quality influencing encoding setting is for example the number of encoding passes. For example, typical video encoders offer a 1-pass, 2-pass, or even multi-pass encoding. Even though the specific number of encoding passes is known while encoding, the resulting bitstream has no indication of the used settings.

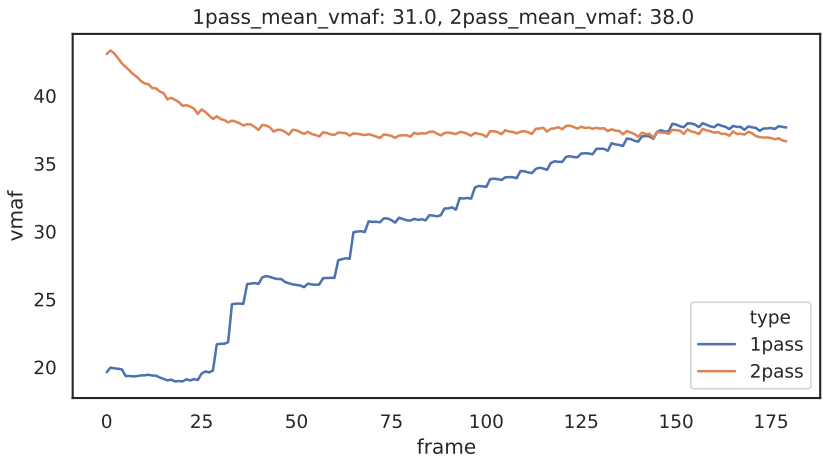


Figure 5.5: Per-frame VMAF scores for one- and two-pass H.264 encoding of one sample video at 500 kbit/s and 720p resolution.

To illustrate the differences in visual quality, an example is shown in Figure 5.5. Here, one example video with a duration of 4 s has been encoded with a 1-pass and 2-pass setting. After the encoding was performed, VMAF scores have been estimated. The video was encoded with H.264 at a resolution of 720p with a video bitrate of 500 kbit/s using both a 1- and a 2-pass encoding scheme. A clear difference in per-frame quality can be observed in Figure 5.5. Moreover, the overall mean quality is also different, with a considerable increase in quality for the 2-pass encoded version. The observed quality difference clearly indicates that knowing whether a 1-pass or 2-pass encoding scheme has been performed is a benefit for a video quality model, and also for other video analysis problems.

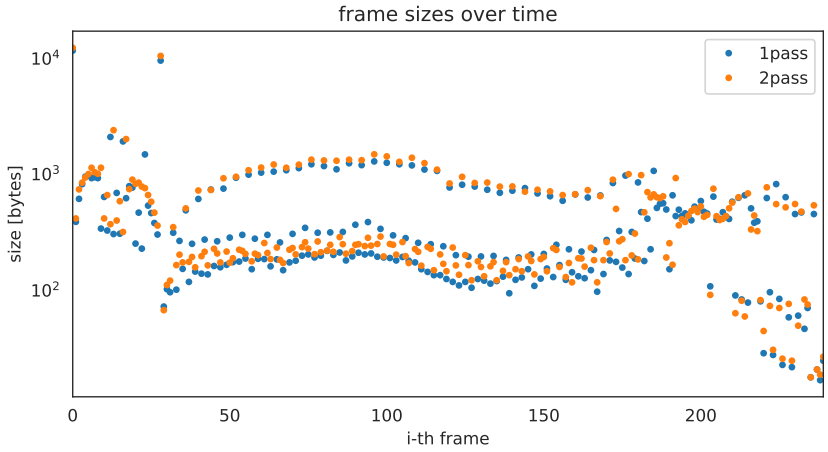


Figure 5.6: Frame sizes per frame of two example videos encoded with H.264 for one- and two-pass encoding for the same video source, framerate and bitrate.

To further investigate other differences of the 1- and 2-pass encoding scheme, the frame sizes of encoded videos have been checked. For example, Figure 5.6 shows two encoded versions of the same source video. The encoding was performed using the same settings for resolution, bitrate, framerate (60fps), preset and codec (H.264). The only difference is in the number of encoding passes. It can be seen that there are small differences in frame sizes. However, it is not directly clear to which extent only frame size and meta-data can be used to distinguish between these two encoding cases. This leads to the general idea to only use meta-data and frame sizes/types as input for a prediction system that is in the following referred to as **prenc** (prediction of encoding settings). It should be noted that this is a minimal set of possible features, increasing the difficulty of the prediction task.

5.3.1 Problem Formulation, Features and Approach

The question of whether a given video is 1- or 2-pass encoded can be formulated as a typical binary classification problem, where the 1-pass encoding setting refers to $class = 0$ and 2-pass to $class = 1$. The overall approach is similar to a no-reference bitstream video quality model.

In the first step features based on frame sizes of the given video are extracted. According to the naming scheme of P.1203 [ITU17], the features are based on Mode 1 type input data. All features can be extracted using FFprobe, a tool that is part of FFmpeg. The video codec is used as a numerical feature named *codec*, where H.264=0 and H.265=1. Furthermore, for a given video, all frame sizes are extracted and afterward normalized by the maximum frame size for the considered video. Subsequently, for all frame sizes the following statistical values are calculated; the mean $mean_{all}$, standard deviation std_{all} and quantiles q^i_{all} with $i \in [0.0, 0.1, \ldots, 0.9, 1.0]$ to characterize the distribution of frame sizes of the given video segment. This pooling is similar to Section 4.1.2 and is performed to enable a time-independent processing of the feature values. Afterward, the percentage of I, P, and B frames of the input video (r_I, r_P, r_B) are estimated. Furthermore, all statistical aggregations (mean, standard deviation, quantiles) are calculated per frame type, resulting in additional 39 values. With this approach, a total number of 56 feature values are used for each encoded video segment.

The second step consists of the machine learning pipeline, where it is possible to change the used algorithm to enable a wider range of evaluation experiments. Furthermore, a feature selection step is performed before the final machine learning approach is trained. All parameters of the used machine learning algorithms are either set to the default values of scikit-learn [Ped+11] or provided in the description of the respective evaluation experiment. The ground-truth data is labeled during the encoding process and can be used in the training phase of the algorithms.

5.3.2 Ground Truth Dataset

To ensure that the setup is similar to real-world approaches, a wide range of encoding parameters are employed. Usually, video providers offer several *DASH* represen-

tations for a smooth and high-quality video playout. For example, Youtube has for most of the videos typically 8 different representations accessible, ranging from 144p to UHD-1/4K resolution, and sometimes up to UHD-2/8K.

Each video is encoded for all combinations of codecs, bitrates, and resolutions once with 1-pass and once with 2-pass encoding. For simplification, all representations share the same framerate of 60 fps. The 2-pass encoding setting uses a fixed bitrate in both encoding passes. Furthermore, the encoding preset 'slow' is used for both codecs (H.264 and H.265) and all encoding passes. The (resolution, bitrate) pairs are selected in an overlapping manner, representing a wider range of possible real-world bitrate ladders. All settings are summarized in Table 5.6. For example, for 360p 0.25, 0.5 and 1 Mbit/s are used as bitrates. In total, 6 different resolutions are considered, and for each resolution 3 bitrate settings, consequently resulting in 18 different (resolution, bitrate) pairs. This leads to a total of $2 \cdot 2 \cdot 18 = 72$ different encoding settings for a given video. All encoding and pre-processing steps were performed using FFmpeg 4.1.3[5].

Table 5.6: Resolution and bitrate combinations for the encoding pipeline, used for both H.264 and H.265.

Resolution	Bitrates [Mbit/s]
360p	[0.25, 0.5, 1.0]
480p	[0.3, 0.6, 1.2]
720p	[0.5, 1.0, 2.0]
1080p	[2.0, 4.0, 8.0]
1440p	[3.0, 6.0, 12.0]
2160p	[4.0, 8.0, 16.0]

After encoding a given video to the described number of representations, a *DASH* segmentation with a fixed segment length of 4 s is applied.

The dataset used for the evaluation consists of a total of 16 different UHD-1/4K short films, details are listed in Table 5.7. 10 out of the 16 used videos were produced at AVT and represent typical short films. All selected clips are at least 3 minutes long and together represent various video genres. Moreover, as input to the encoding pipeline, only uncompressed video material is considered to further avoid influences of previously applied encoding steps.

[5] https://www.ffmpeg.org/

Table 5.7: Videos used in the evaluation; all 4:2:2 chroma sub-sampled; * only in 8 bit, otherwise 10 bit.

video sequence	type	source	duration [s]
a mysterious case	real	own	335
big buck bunny*	animated	blender.org [Blea]	523
daydreamer	real	own	531
dead mans hand	real	own	298
der morgen danach	real	own	304
ein abend zu zweit	real	own	461
el fuente	real	Netflix	477
fr debris	animated	own	436
geist	real	own	361
giftmord	real	own	429
meridian	real	Netflix	718
bennu's journey	animated	Nasa [Nas]	360
nightcall	real	own	619
sintel*	animated	blender.org [Bleb]	888
sparks	real	Netflix	229
splitter	real	own	377

Using the presented encoding pipeline the training and validation dataset is generated. Applying the encoding for the 16 input videos leads to a total of 131,976 *DASH* segments with a duration of 4 s, with 50% being one-pass and 50% two-pass encoded.

5.3.3 Results for 10-fold Cross Validation

To investigate the performance of several machine learning algorithms (*SVC*, *RF*, *k-nearest neighbors algorithm (KNN)*, *GBC*) the first evaluation uses 10-fold cross validation and includes variations of hyper-parameters. For example, for *RF* and *GBC* the number of trees in $[10, 50, 100, 150]$ is changed. Moreover for all algorithms the feature selection is evaluated, starting from no-selection $FS(0)$, to $FS(0.5)$ and $FS(1)$. Further, for every parameter combination 10 models are trained, and mean performance values (f1-score, accuracy, precision, and recall) are calculated afterward, resulting in a total number of 300 trained models.

In Table 5.8 the results of the 10-fold cross validation are summarized. The best performing algorithm is the *RF* model with no feature selection and 150 trees. However, the performance in the top-10 of all models is quite similar, where RF-based models

Table 5.8: 10-fold cross validation results, all values are mean values for 10 repetitions, values for the metrics are mean values across both classes. All values are sorted by f1 score.

model	FS(c)	trees	f1-score	accuracy	precision	recall
SVC	0.5	-	0.567	0.547	0.526	0.594
RF	1	10	0.572	0.607	0.567	0.525
RF	0.5	10	0.574	0.609	0.569	0.528
SVC	0	-	0.577	0.549	0.527	0.616
RF	0	10	0.583	0.616	0.574	0.536
KNN	1	-	0.587	0.586	0.550	0.589
KNN	0	-	0.594	0.591	0.554	0.598
KNN	0.5	-	0.596	0.594	0.556	0.600
GBC	0	150	0.599	0.591	0.553	0.611
GBC	0.5	100	0.600	0.587	0.550	0.619
GBC	0	100	0.600	0.587	0.551	0.619
GBC	0.5	150	0.600	0.592	0.554	0.612
GBC	1	150	0.603	0.583	0.548	0.634
GBC	0	10	0.604	0.563	0.535	0.667
GBC	1	100	0.605	0.577	0.544	0.648
GBC	0	50	0.606	0.579	0.545	0.649
GBC	0.5	10	0.607	0.563	0.535	0.674
GBC	0.5	50	0.608	0.579	0.545	0.652
GBC	1	50	0.617	0.573	0.541	0.687
GBC	1	10	0.618	0.557	0.531	0.718
RF	1	50	0.630	0.636	0.587	0.621
RF	0.5	50	0.632	0.638	0.589	0.623
RF	1	100	0.641	0.642	0.591	0.639
RF	0	50	0.642	0.647	0.596	0.632
RF	0.5	100	0.642	0.643	0.592	0.640
RF	1	150	0.644	0.643	0.592	0.645
RF	0.5	150	0.645	0.645	0.593	0.645
SVC	1	-	0.651	0.551	0.527	0.836
RF	0	100	0.652	0.653	0.600	0.650
RF	0	150	**0.655**	**0.655**	**0.601**	**0.656**

seem to outperform the other models, even with a lower number of trees and higher feature selection criteria.

Further, in Figure 5.7 the confusion matrix of an example *RF* model with 150 trees and $FS(0)$ trained using 10-fold cross validation is shown. It can be observed that for about 65% of all videos in the 10-fold cross-validation the model is able to predict the class correctly, about 43k videos each are in the correct predicted classes each.

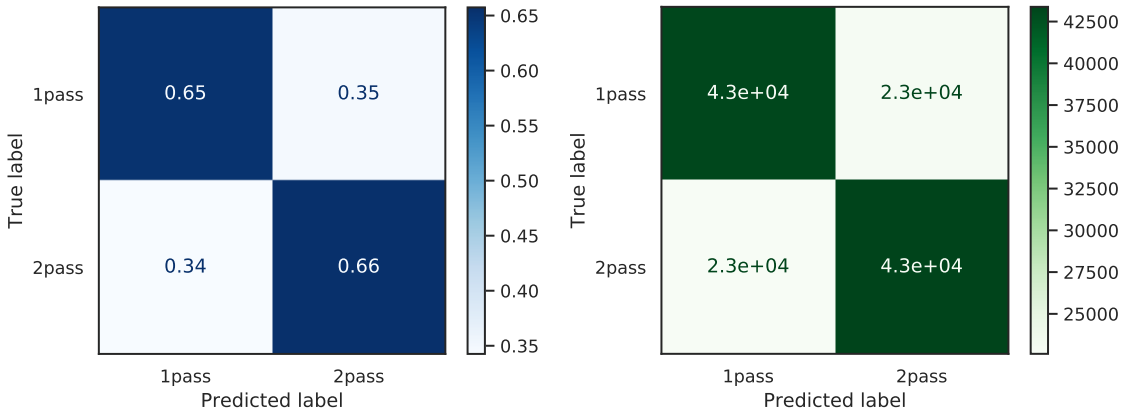


Figure 5.7: Confusion matrix of RF model: 150 trees, $FS(0)$; normalized values (left), absolute (right).

5.3.4 Results for 50%-50% Split Validation

For the following evaluation experiment, only the following models are considered: *RF* models with $FS(0)$ and 150 trees, *svc* with $FS(1)$ and *GBC* with $FS(1)$ and 50 trees, because these models showed promising results in the 10-fold-cross validation experiment. All reported values are mean values for 10 repetitions. Each run consists of a random sampling of training and validation data, forcing that no source videos are shared across training and validation data.

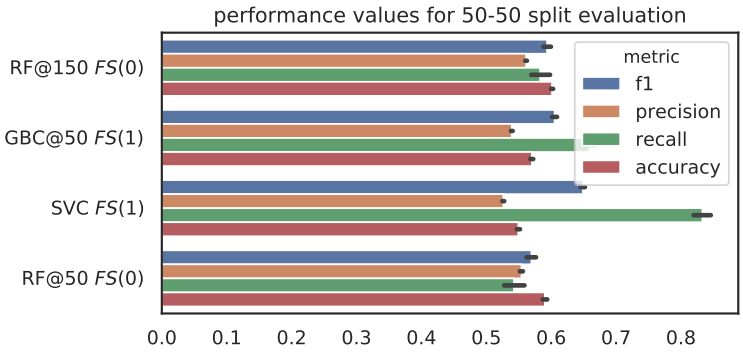


Figure 5.8: Mean of performance metrics using a 50-50 train validation split without overlapping source videos, including 95% CI values.

In Figure 5.8, mean values for the three identified machine learning models are shown. Based on the error bars, it can be clearly seen that there is only a small variation due to the different selected splits for all calculated metrics. Moreover, in terms of performance, the *svc* model is the best, followed by the *GBC* and both

RF models, with a range of 0.6 to 0.65 considering f1-score. The *RF* models behave similarly in terms of performance.

Comparing these results with the 10-fold-cross validation findings, a similar characteristic of the considered models can be observed. This leads to the conclusion that the introduced approach can also be used with *svc*, *GBC* or *RF* models. It also shows that the feature set is promising for the defined classification tasks. Besides this, even in the case where the videos are unknown to the system as in the 50-50-split, the models can produce similar results compared to the 10-fold-cross validation experiment.

5.4 Speed up Approaches

The general method introduced in Chapter 4 describes in Section 4.1.3 an approach to speed up feature calculation for video quality prediction. From a high-level point of view, two different ways would be possible to reduce the computation time of general video (quality) prediction models without modifying the underlying features. The first possible technique could reduce the number of video frames required to be processed by the model. VMAF offers such an approach [Net21], where only a subset of frames is handled for video quality estimation. The idea here is that only a subset of frames are required to estimate the overall mean quality of a short duration video sequence, and the focus is that this quality is depending on the spatial information of each of the frames. In complement, the second approach would focus on motion-related aspects of quality, here the information to be processed for each frame would be reduced, e.g., focusing on a specific area of the video. This approach would reduce the spatial information per frame, however it would still allow to consider motion features for the quality estimation. Both approaches offer speeding up of the underlying quality estimation under the assumption that the error is only minor. For example, a better selection of sub-sampled frames could improve the overall video quality estimation, or using the second approach, a focus on specific (quality) important regions of the video may have a similar effect. In Section 5.4.3 a simple center cropping approach has been evaluated for the developed video quality models, the question arises whether such an approach would also work with other

state-of-the-art video quality models. For this reason, **cencro** has been developed. **cencro** is a generic center cropping approach that builds upon Netflix's VMAF tool and allows to evaluate several center cropping variants. The code for **cencro** is publicly available[6].

5.4.1 Frame Reduction to Speed up Video Quality Calculations

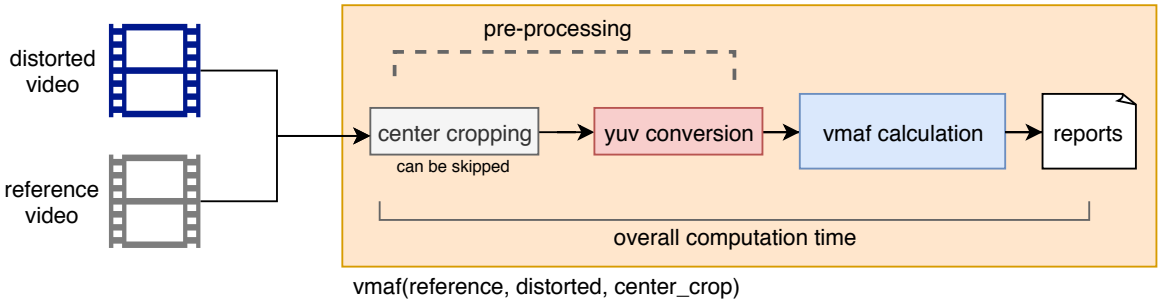


Figure 5.9: General structure of *cencro*; based on a distorted and reference video, center crops are estimated. These center crops are converted to yuv and later used for VMAF calculation. It is possible to deactivate the center crop step; the overall processing time is measured for later analysis.

The per-frame reduction approach consists of several steps and is based on a full-reference video quality model. In Figure 5.9, the general structure of the *cencro* method is shown. First, a pair of videos (d,r), a distorted d and a reference video r, are rescaled to the same resolution, framerate, and color space based on the reference video. In total 18 different center crop values are considered. The values range from 144p to 1800p (compare Figure 5.10; commonly used streaming resolutions highlighted in **bold** indicating that these are used as starting points). Only center crops are considered following the intuition that the most important video content is presented in the middle of the video – at least over a number of frames. For all analyzed center crops the used width is automatically adapted based on the aspect ratio of the given reference video. A maximum height 2160p is assumed because it fits all of the videos used in the evaluation. However, the concept could also be used for UHD-2/8K video quality estimation and the associated speed up. In the case of UHD-2/8K video, the advantage of a faster computation is even more important.

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

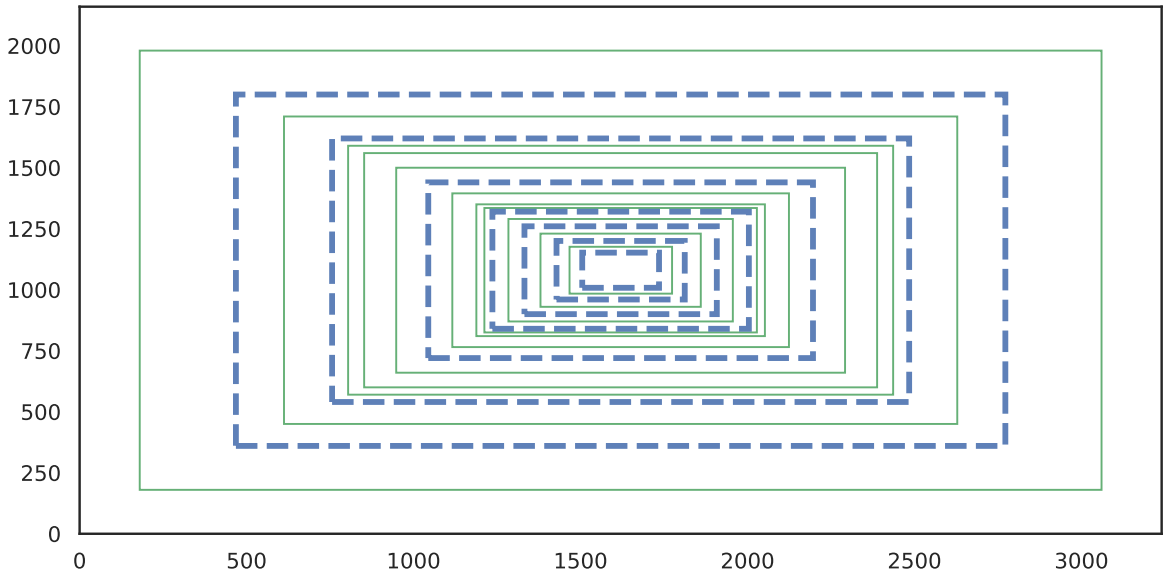


Figure 5.10: Visualization of used center crops, blue are highlighted resolutions; **144**, 192, **240**, 300, **360**, 420, **480**, 510, 540, 630, **720**, 840, 960, 1020, **1080**, 1260, **1440**, 1800; widths are adapted based on aspect ratio of reference video.

For the distorted and reference video, selected center cropped versions are extracted, referred to as d_{cc}, r_{cc}, respectively. Moreover, it is important that these center crops cover the same area of the video, which is why the distorted videos are rescaled before to the corresponding reference video resolution. Both cropped videos d_{cc} and r_{cc} cover the same area of the video and are converted to a yuv representation. In case the approach is used without center cropping, the center-cropping part is just skipped and videos are directly converted to yuv. The resulting yuv versions of the videos ($d_{cc} = d, r_{cc} = r$) are passed to VMAF to estimate video quality. Further measurements for calculation-time are performed, starting from the first pre-processing step. Usually in real-world applications, such pre-processing is required due to the fact that videos are not archived in yuv format, or such a conversion is done internally in VMAF (using ffmpeg). The VMAF tool is not only used to extract VMAF scores, it also includes PSNR, ADM2/DLM [Li+11], SSIM [Wan+04] and VIF [SB06]. All extracted video quality values are evaluated later because they are calculated along with VMAF. The processing time for the other metrics is harder to extract because the calculation is performed within the VMAF tool and thus only the overall time that VMAF took with the pre-processing time is considered in the following. As time measure, Wall Clock Time is used. In total three different parts

of the processing steps are measured: reference video conversion time t_r, distorted video conversion time t_d and VMAF runtime t_{VMAF}. The overall time follows as $t_{all} = t_d + t_r + t_{VMAF}$. Although some parts of the approach could be run in parallel, this is avoided to ensure a fair single core comparison. To obtain one comparison quality value similar to a *MOS*, the mean value of all VMAF per-frame scores is used for comparison. In the following several center crops of different sizes are considered, because the evaluation targets to figure out the best performing center crop. Clearly, a smaller center crop will lead to a higher quality prediction error. However, whether the error is acceptable or not is so far not clear, because this approach is a trade off between speed up and suitable prediction error.

5.4.2 Evaluation of different Crop-settings

For evaluating the **cencro** approach, it is required to have a proper video database with several conditions, e.g., different codecs, resolutions, and with included subjective scores. A subset of the AVT-VQDB-UHD-1 [Rao+19a] forms the dataset. More precisely, test#1 (introduced in Section 4.2.2) without the water_netflix sequence is used. Thus, this subset of test#1 consists of in total 5 source videos with a duration of 10 seconds each. Each of the source videos was encoded to 150 different stimuli and include subjective scores.

In Figure 5.11 the correlation of subjective scores with uncropped VMAF scores is shown. VMAF performs quite well for subjective score estimation with, e.g., a pearson correlation of around 0.93 and a small *RMSE* of 0.43. For all *RMSE* calculations the VMAF scores are linearly transformed to a 1-5 *MOS* scale, similar to [GSR18]. The uncropped VMAF scores are used in the following as reference VMAF scores, to compare center-crop estimated scores with these.

To analyze the impact of center cropping on VMAF scores a total number of 18 different center crops are evaluated. Moreover, it should be noted that the 150 distorted videos lead to 2700 VMAF calculations using all possible center crops. As a first starting point, the correlations of uncropped VMAF and center cropped VMAF values are checked.

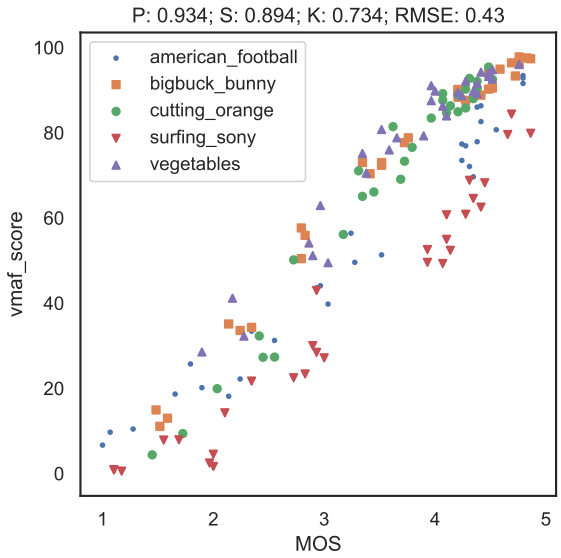


Figure 5.11: Correlation of MOS ratings with uncropped VMAF scores; P=pearson, S=spearman, K=kendall correlation coefficients.

In Figure 5.12 scatter plots for a selection of 9 center crops (144p, 240p, 360p, 480p, 720p, 1080p, 1260p, 1440p, 1800p) are shown. The estimated VMAF scores with applied center cropping are compared with the uncropped VMAF scores. Considering these center crops, it can be concluded that even the smallest crop has quite a high correlation with the original VMAF scores. For center crops with at least 1080p there are nearly no differences visible.

Further, in Figure 5.13 mean absolute errors with 95% confidence intervals are shown, the grouping is based on different source videos to analyze the source-video impact on the center-crop approach, mean aggregation is performed for each different source video including all distorted versions. It can be concluded that for most videos of the dataset starting from a 720p crop only a small difference is measurable. However, for example, the surfing_sony sequence seems to be a bit harder to handle with the center crop approach. The surfing_sony sequence is characterized by several compression artifacts that are not always visible in the middle of the video, as it can be seen that the error is reduced for larger center crops.

On the other side, for the american_football sequence starting with 360p center cropping there is no change visible. This specific sequence consists of high motion where compression artifacts are uniformly distributed across the video frame. There

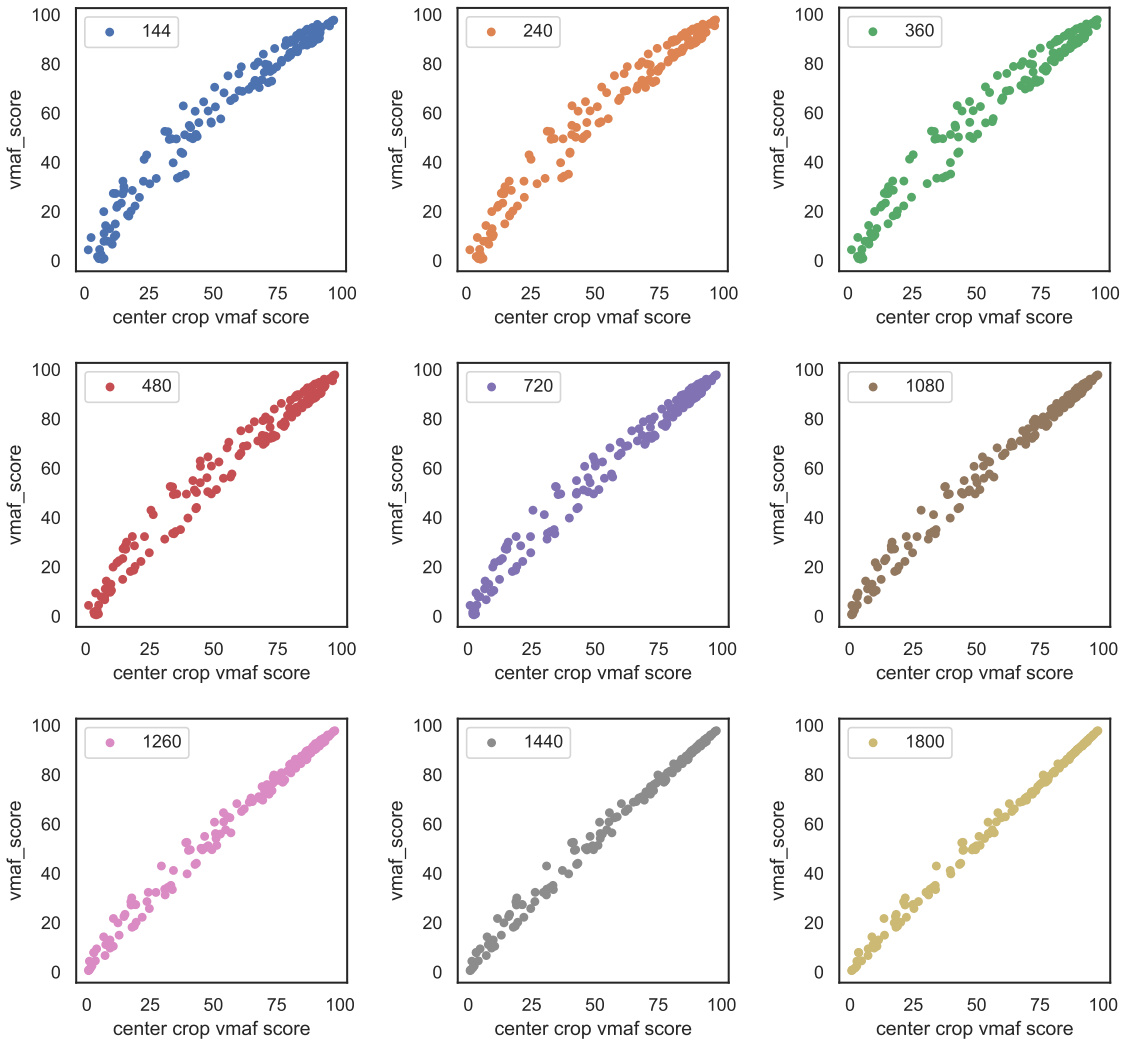


Figure 5.12: Selected example scatter plots for several center crop parameters, y axis always the uncropped VMAF score, x axis VMAF score with applied center cropping; all 150 distorted videos of the database are shown.

is a clear content-dependency visible for the used center cropping setting. However, especially for video encoding optimization, an "optimal" crop could be estimated in a first small run, and later all other calculations could be done using the estimated crop.

So far, the evaluation was focused on cropped and uncropped VMAF scores, as next the comparison will be performed with the *MOS* values.

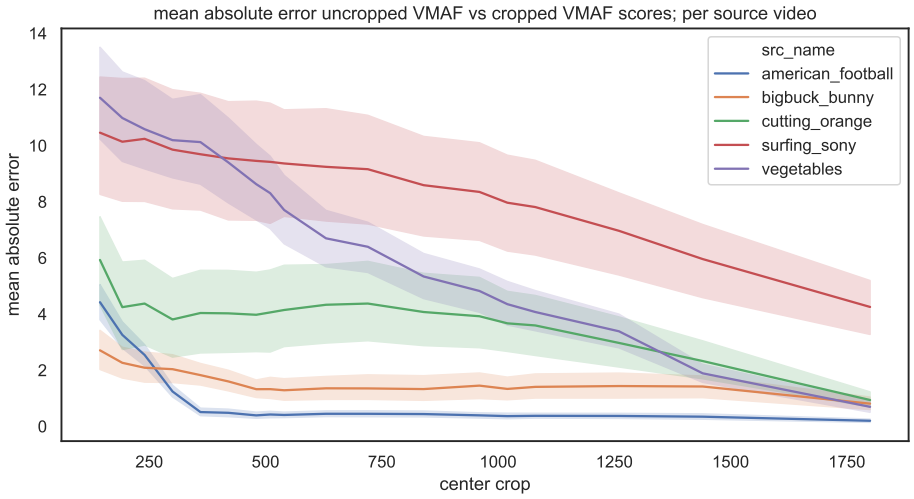


Figure 5.13: Mean absolute errors of uncropped and cropped VMAF scores considering different source videos.

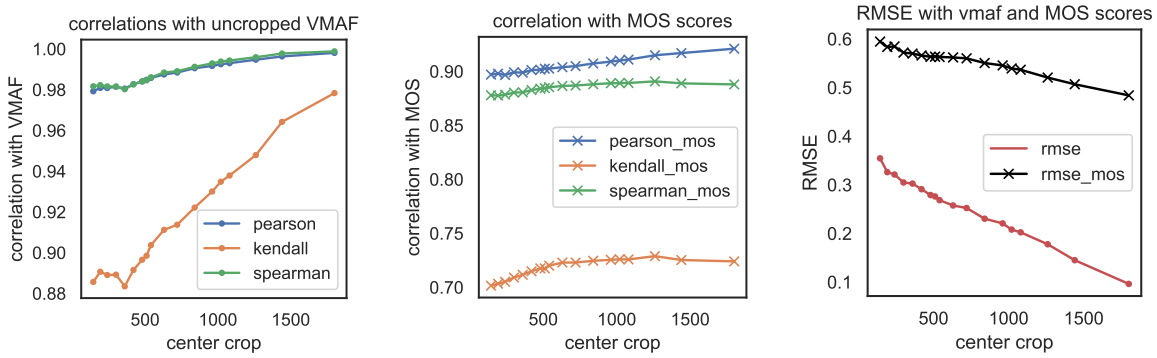


Figure 5.14: Correlations of center crops with VMAF scores (left), MOS (middle) and RMSE (right).

In Figure 5.14 the calculated correlations and RMSE values are presented. From the correlation of uncropped VMAF calculations with center-crop-based scores, it can be concluded that if the crop is larger the correlation increases, and all three correlations behave similarly in this regard. Only the kendall correlation improves strongly if center crops are larger. It can also be argued that starting from a 500p crop, pearson and spearman values are not improving strongly (pearson from 0.985 for a 510 center crop to 0.998 for a 1800 center crop; spearman 0.985@510cc to 0.999@1800cc) similar with RMSE for *MOS* values (right figure, RMSE_mos, 0.564@510cc to 0.484@1800cc). In the other part of the plot, the correlation with *MOS* values can be seen. Comparing the correlations of uncropped VMAF pearson 0.934, spearman 0.894 and kendall of

0.734 it is notable that none of the crops is able to reach these values, due to the loss of information. However, considering the introduced error of each crop, it is visible that even a 360p center crop has a quite good performance, and theoretically reduces the processing time enormously. For a 360p center crop, the correlation values with *MOS* are pearson=0.899, spearman=0.881, and kendall=0.712, they are highly similar to the uncropped VMAF performance, for pearson a reduction of 4%, for spearman 2% and kendall 3% can be concluded as an error. So it follows that a 360p center crop would lead to a maximum loss of 4% in terms of pearson correlation. Comparing this result with, e.g., the cross-lab evaluation with reported pearson correlations of Pinson and Wolf [PW03] ranging from 0.902 to 0.935 ($\approx$ 4% error), it can be stated that the 360p center crop error is negligible. In addition, also the RMSE for VMAF and *MOS* comparison decreases starting from a 500p center crop, in case of 360p center crop an RMSE of 0.57 compared to *MOS* can be estimated, the original VMAF RMSE to MOS values was 0.43 (compare Figure 5.11).

5.4.3 Speed up for Video Quality Metric Calculation

Next to the introduced prediction error using the center crop approach, also the cpu-time savings are important to analyze.

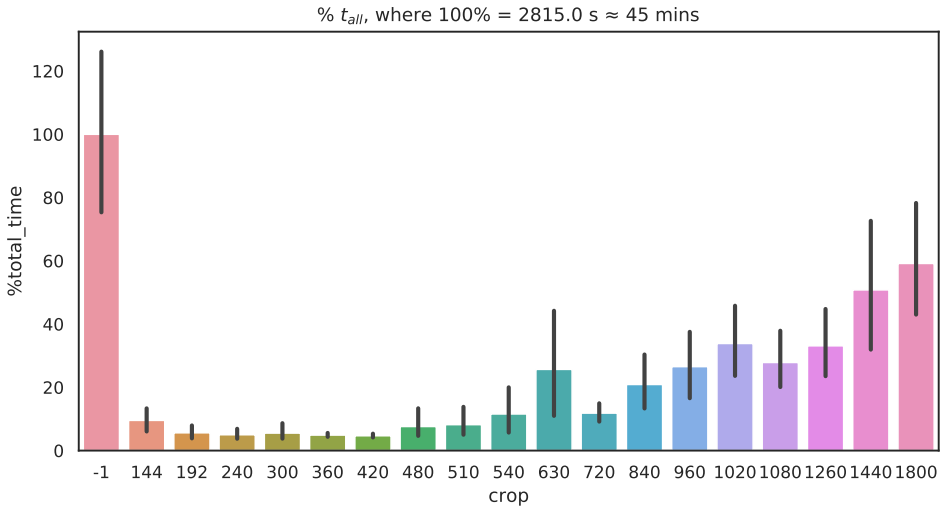


Figure 5.15: Mean % total computation time for all crops, -1 refers to an uncropped calculation, for this the calculation needed in average $\approx$ 45 minutes per video, also 95% confidence intervals are shown.

In Figure 5.15 the cpu processing time for all conducted video quality estimations is shown, with a total of 18 different crop settings along with the calculation without cropping. All of the runs were performed on the same machine, a server system with 128 GB RAM, 40 logical cores (20 physical), and video stored on a local and fast raid-5 disk.

Figure 5.15 summarizes quite well that every used center crop is leading to a cpu-time improvement, even with the additional pre-processing and source-dependent steps involved. Moreover, the percentage of changes in total runtime t_{all} is calculated, here the mean value of all calculations without a center crop approach is used as reference. As it can be seen, already the largest center crop of 1800p will save up to 40% processing time. Moreover, a 360p center crop will lead to around 5% of processing time (saving of 95% of time), in the experiments this is the difference from $\approx$ 45 minutes to $\approx$ 3 minutes per sequence. It should be mentioned that even though the t_{all} is evaluated the most time-consuming part is t_{VMAF} and the processing time for t_d and t_r are nearly constant for a specific center cropping setting.

5.4.4 Evaluation of Time and Error

Combining the results of the previous two sections, the focus is now the evaluation of cpu-time and the introduced error of the **cencro** approach.

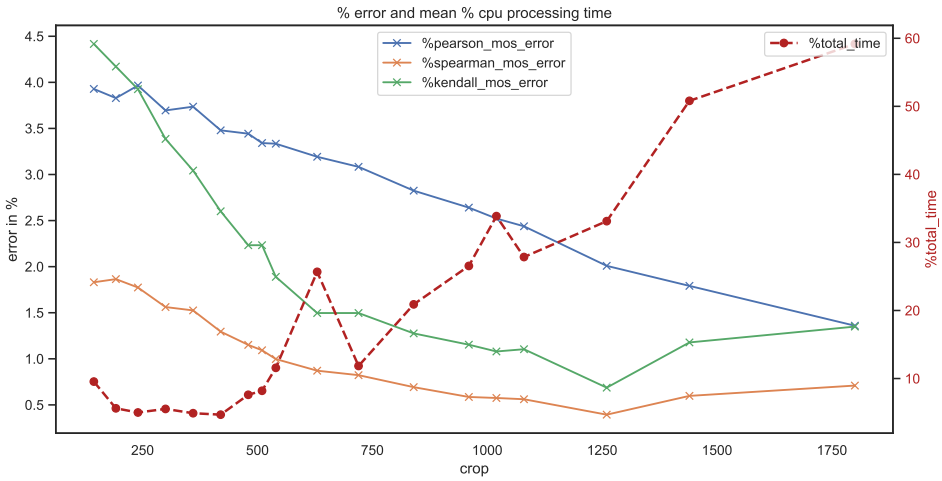


Figure 5.16: Overview of introduced percentage error (left axis) for several correlations and required mean percentage cpu-time (right axis).

In Figure 5.16 an overview of used mean percentage cpu-time (compared with the uncropped processing) and the introduced percentage error for pearson, spearman, and kendall compared with MOS values are shown. For example for a 1020p center crop, there is only 34% of overall cpu processing time required, where the introduced error is around 2.5% for pearson, 0.6% for spearman, and 1.1% for kendall correlation. It can be concluded that a 1020p center crop introduces only a small error. On the other side, for smaller center crops, processing time decreases, and error in all cases increases. In the case of a 360p center crop approximately 4.9% of cpu-time is required, where for pearson approximately 3.7%, spearman 1.5%, and kendall 3.0% error is introduced. It shows that even for a 360p crop the approach is able to produce relatively good results.

5.4.5 Cencro applied to other Video Quality Metrics

Table 5.9: Comparison of **cencro** approach with other full-reference metrics, correlations are calculated in comparison to MOS values; a negative error indicates an improvement to the uncropped metric correlation. P=pearson, S=spearman, K=kendall correlation

metric	crop	P	S	K	$\mathbf{E}(P)$	$\mathbf{E}(S)$	$\mathbf{E}(K)$
ADM2	-1	0.927	0.877	0.715			
	360	0.892	0.859	0.680	3.835	2.002	4.91
	1080	0.907	0.869	0.700	2.17	0.846	2.115
	1440	0.913	0.867	0.699	1.518	1.034	2.317
SSIM	-1	0.768	0.762	0.597			
	360	0.661	0.699	0.544	13.877	8.344	8.785
	1080	0.797	0.769	0.607	-3.887	-0.927	-1.691
	1440	0.790	0.770	0.606	-2.875	-0.984	-1.66
PSNR	-1	0.745	0.705	0.541			
	360	0.759	0.751	0.576	-1.855	-6.597	-6.353
	1080	0.758	0.729	0.563	-1.725	-3.429	-4.025
	1440	0.749	0.712	0.554	-0.488	-1.045	-2.295
VIF	-1	0.727	0.706	0.538			
	360	0.759	0.767	0.590	-4.341	-8.594	-9.5
	1080	0.757	0.745	0.575	-4.081	-5.538	-6.724
	1440	0.745	0.726	0.566	-2.442	-2.843	-5.151

In addition to VMAF, the performance of **cencro** regarding different other full-reference metrics (PSNR, SSIM, ADM2 and VIF) is analyzed. Besides pearson, spearman, and kendall correlations to *MOS* values the percentage of error with the uncropped correlation is calculated as $\mathbf{E}(X)$, where $\mathbf{E}(X) = 100 - (X/X_{-1})$ with X_{-1} is the corresponding correlation in the uncropped case and X is the correlation for the specified center cropping setting. In Table 5.9 an overview of important selected center crops is presented. As crop settings 360p, 1080p, and 1440p have been selected, all the other center crops behave similarly. The ADM2 score without cropping shows similar performance as VMAF, because it is also used in VMAF. Comparing the 360p center crop performance of ADM2, it can be concluded that the maximum error is for kendall correlation of about 5%, where pearson error is lower than 4%. In general, all crops seem to have a high correlation than the other full-reference metrics. The other metrics, namely SSIM, PSNR, and VIF, show for some crops even an improvement in performance compared to *MOS*, however, their overall correlations to *MOS* are not good considering VMAF and ADM2.

5.4.6 Cencro for Gaming Video Quality

Table 5.10: Performance of **cencro** for GamingVideoSET; a negative error indicates an improvement to the uncropped metric correlation. P=pearson, S=spearman, K=kendall correlation

metric	crop	P	S	K	$\mathbf{E}(P)$	$\mathbf{E}(S)$	$\mathbf{E}(K)$
VMAF	-1	0.827	0.825	0.639			
	240	0.888	0.889	0.715	-7.323	-7.777	-11.899
	360	0.889	0.890	0.716	-7.518	-7.987	-12.136
	720	0.882	0.889	0.713	-6.619	-7.853	-11.584

Besides classical *DASH* streaming, another evaluation scenario namely recorded gaming sessions should be considered. Such sequences have different properties than classical movie scenes, mostly because of the computer-generated textures used in games and a different encoding.

For the evaluation the GamingVideoSET [Bar+18b][7] is used, compare with Section 5.2.1. This dataset consists of 24 Full-HD 30 fps source videos [Bar+18b] from

[7] download https://kingston.box.com/v/GamingVideoSET

different gaming genres encoded with several distortions each with a duration of 30 s. Because the GamingVideoSET consists of Full-HD videos, only the following center crops 240p, 360p, 720p, and the uncropped case are considered. The uncropped VMAF values are re-calculated and vary in comparison to the originally published study [Bar+18b] due to the different VMAF versions used.

In Table 5.10 the results for the GamingVideoSET are presented. All uncropped VMAF correlations are still reasonably high, however, it can be seen that in all of the used center crops, the performance increased compared to subjective scores. For example, having a view on the pearson correlation, the percentage error is around -6% to -7% what shows an improvement from a correlation value of 0.83 to around 0.88. There are several reasons for the improvement in all center crop cases possible. One is for example that gaming videos are more center-oriented, due to the fact that usually heroes/game characters or important parts of the game are in the middle of the screen positioned. Another reason is that the used encoder is H.264 where, e.g., block sizes are fixed, and thus block artifacts are more uniformly distributed in one video frame. In addition to correlation, cpu-time is checked for the calculations. Here, the focus is only on $\%t_{all}$, where t_{VMAF} has the largest impact.

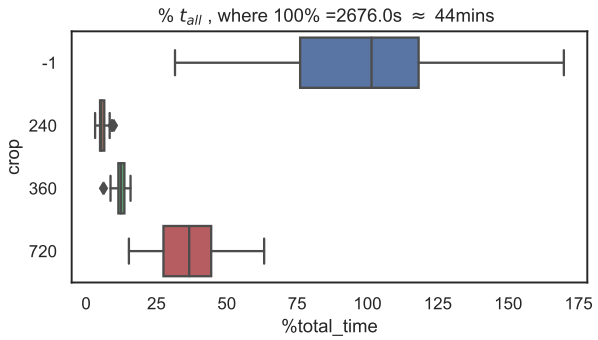


Figure 5.17: Percentage of processing time in case of GamingVideoSET; as 100% the mean calculation time of the uncropped processing is assumed.

In Figure 5.17 the percentage of cpu processing time for each analyzed crop are shown. On average the uncropped VMAF calculation cpu-time was around 44 minutes, similar to the UHD-1/4K video quality calculation because the gaming videos have a duration of 30 seconds in contrast to the 10 seconds of the used UHD-1/4K videos. It is clearly visible, that it can be up to 60% or even around 90% of calculation time saved with different crop settings. It was further shown that there are no prediction

performance disadvantages using **cencro** for gaming videos that are included in the GamingVideoSET.

5.5 Summary

The general model pipeline introduced in Chapter 4 was developed for the video quality prediction problem. However, other video problems exist, ranging from source video analysis to speed up of state-of-the-art video quality models. It was shown that using the features it is possible to answer several other video-related questions. The evaluation was usually performed with additional datasets, 10-fold cross-validation or other split evaluations where the settings have been chosen in a way that the model has unknown data in the validation step.

One of the several video problems is for example the question of whether users are able to distinguish between UHD-1/4K and Full-HD videos. To answer this question, subjective tests have been carried out and a machine learning model has been trained and evaluated using the subjective data.

Moreover, non-traditional video content such as gaming sessions that are streamed is increasing in popularity. To also cover such computer-generated content and their specific properties, a video game genre classifier and a specifically optimized video quality model have been described and evaluated showing promising results.

Furthermore, in the field of video processing tuning of encoding parameters is a crucial part. To estimate which settings have been used after the processing has been performed a system called **prenc** was described. The system is based on frame statistics and is able to estimate when a video was 1- or 2-pass encoded.

Encoding optimization is usually performed with objective video quality metrics, however, it can be observed that accurate metrics usually require even more processing time, making the overall task challenging. As an approach to reduce the processing time with a minor or insignificant error the **cencro** approach has been introduced and evaluated. With a specific center cropping setting of 360p in the case of a traditional UHD-1/4K video or Full-HD gaming video, a significant amount of processing time can be saved with an error that is comparable to inter-lab errors

while conducting subjective tests. Extensions of this approach would include the estimation of most important regions using, e.g., saliency models.

Chapter 6

Conclusion and Future Work

Due to the increase of uploaded images and streamed videos, the importance of evaluating the visual quality has increased, for example, to improve compression methods and thus reduce the overall bandwidth needed for streaming applications or storage of photos. Moreover, beside more photos and videos uploaded or consumed, higher-resolution images and videos are getting accessible for everyone and which also necessitates an evaluation of quality considering newer methods for compression. Within the context of higher resolutions, visual quality perception and prediction, various research questions can be derived. The focus of the identified research questions are based around the following topics, namely, machine learning models, differences of higher resolutions, speedup of calculation, and the interlinking of image and video quality and compression. Important for the scope of the presented work are subjective tests which have been conducted considering image and video quality. Moreover a generic machine learning framework has been developed to analyze several video and image prediction problems. The main "tools" that are used to answer the identified research questions in this thesis are the following. Firstly, data is essential for the evaluation of quality perception. Thus suitable datasets are required and necessitated the need to conduct subjective tests or use publicly accessible data or other approaches. For example, some of the used datasets in this thesis are synthetically generated and extended by subjective ratings later. Secondly, machine learning models with suitable features can be applied to various image and video analysis problems. Here, it should be mentioned that a proper definition of the features considering the scope is required and that the evaluation of such models should avoid over-fitting.

In the following section, a brief overview of the results of this thesis is summarized, and finally, an outlook is presented on how the provided work can be used and extended to cover future topics in the field of quality prediction.

At first, image compression and quality assessment have been evaluated in this thesis using several large-scaled datasets in Chapter 3, with a specific focus on higher resolutions. As an outcome, it can be concluded that modern video encoders (AV1, H.265) can compress images in a better way than usual JPEG considering filesize and quality. Furthermore, lab tests and online or crowdsourcing-based tests can be used to evaluate image quality, even for higher resolution images considering some adjustments. The required adaptations and modifications of the crowdsourcing paradigm to implement higher-resolution images are described in addition and have been already applied to other online tests [RGR21b] as a consequence, which resulted also in the extension of the AVT-VQDB-UHD-1 [Rao+19a] . One of the proposed modifications is to use Full-HD sized image patches, because for example usual crowd users may only have access to lower (HD or Full-HD) resolution screens. In addition to this, VMAF can be used to evaluate image quality for higher-resolution images, as the subjective lab and online tests and the corresponding evaluation indicate. Moreover, it has been shown that deep learning-based image quality models can be trained to predict users' quality perception. An extension of such a model can be further used to tackle the video quality problem, however, the high processing time is one reason that such models may not be practically applicable. Chapter 3 is motivated by research question 1 and 5. The question 1 is about using machine learning for quality prediction, and as it has been shown, image quality can be well predicted using random forest models or deep neural networks. For research question 5, which covers how video compression and video quality models can be applied to images, it can be concluded that VMAF is a suitable candidate for image quality prediction and that recently developed video compression methods outperform state-of-the-art image compression methods.

To solve the issue of practical usable video quality prediction models, several models have been proposed in Chapter 4. The questions 1 (machine learning models for video quality), 2 (speedup of calculation), and 4 (prediction of more than mean opinion scores) are handled in Chapter 4 with a common and generic model architecture. The variants of the models introduced range from no-reference to full-reference and

hybrid versions, which use meta-data as only additional input to the pixel data. The models are publicly available as open-source packages and researchers are invited to use, extend and build upon them. The training has been performed on the AVT-PNATS-UHD-1 dataset. The evaluation was performed using the publicly available dataset AVT-VQDB-UHD-1 [Rao+19a] to enable reproducibility. The training and validation datasets do not share common source videos and are only partially similar in their design. The models outperform common state-of-the-art models on the AVT-VQDB-UHD-1. All models can be applied to traditional mean opinion score prediction, or handle the video quality prediction as a classification or multi-instance regression problem. Therefore video quality may not be summarized as a mean opinion score only, and that for example, a prediction of rating distribution may be helpful in understanding quality prediction even better. The developed video quality models follow a common structure, features, and all use machine learning. The introduced architecture is generic enough that it can be even used for other video analysis problems.

To demonstrate other applications, in Chapter 5 various example instances using the general architecture are described and evaluated with different datasets, following the questions 4, 3 (perception of higher resolutions), and 2 (speedup of calculations). These instances cover topics like video classification considering UHD-1/4K, gaming video quality prediction and genre prediction for gaming videos, encoding parameter estimation, or speeding up traditional state-of-the-art video quality models. For most of the presented video problems, the introduced architecture can be used with good prediction results, which were either comparable to or better than state-of-the-art models. For example, the approach to speedup video quality models is based on a per-frame reduction using a center crop. It is shown that center cropping can significantly reduce the overall processing time of quality calculations, for example by 95% for a 360p center-crop of videos with UHD-1/4K resolution with a negligibly low error for VMAF.

The focus of this thesis was to evaluate higher-resolution image and video quality using machine learning. However, the increase of resolution and other technology improvements have not stopped, for example, UHD-2/8K videos are already streamed [NHK20] or 360° videos are widely accessible. Thus the presented work is a snapshot. Even though the developed features and modeling pipeline have been

only evaluated on UHD-1/4K videos or specific high resolutions for images, the concepts may be applicable even for higher resolutions or other media extensions. Further, deep learning for images is an important part, and image enhancement may lead to different distortions, that still are required to be evaluated also for the video case. As it is to be expected that deep neural networks for enhancements are applied in the future to videos with higher resolutions. Other aspects, such as HDR, higher framerates, and viewing distance variations are additional factors that could be included in the proposed models. Moreover, the video quality prediction models only target short-term video quality. For long-term video quality covering several minutes, extensions are required, where the models could be used as components, similar to the ITU-T P.1203 [ITU17] integration module. However, most of the problems that have been investigated were solved using traditional machine learning, thus deep learning may not be required for all problem instances and would also limit the interpretability of the trained models.

Even though several video problems have been evaluated in this thesis, the end-to-end chain has more open challenges. One example is the network-centric prediction of video quality in the case of encrypted video streams, for example, see [GRF17; OS20], or a camera sensor oriented quality evaluation. Another challenge could be focusing on even higher resolutions than UHD-1/4K in all parts of the delivery and processing chain. In this context, the proposed approaches are to be seen as a starting point for extensions, for example, the **cencro** approach may be used to further evaluate UHD-2/8K resolution videos, other formats such as 360° video (where **cencro** has been already applied in follow up work [Fre+20a]) or the center cropping could be replaced by fast saliency-based region of interest estimation. Furthermore, the introduced video quality models can be extended by new features or re-trained with different datasets or mapped to other parts of the end-to-end video processing chain.

In general, there is an increasing tendency to replace or extend traditional videos or images by immersive media technology. For example, using head-mounted displays and controllers users can have more interactivity, light field images enable us change the focus within an image, or it is possible to present more than audiovisual content, for example including room lighting, haptic feedback, or other perceivable elements. The conducted work within this thesis may then be just one building

block in the overall integration and prediction of more comprehensive media experience. However, there are further factors that may be included in the evaluation of media experiences, such as personal preference, social signals, group experience of audiovisual content, cinematographic style, energy consumption, and more.

Picard: "Mister Data, lay in a course for the twenty-fourth century. I suspect our future is there waiting for us."
Data: "Course laid in, sir."
Picard: "Make it so."
(Star Trek: First Contact)

Bibliography

[99f20] 99firms. *Instagram Marketing Statistics*. 2020. URL: https://99firms.com/blog/instagram-marketing-statistics/%5C#gref (visited on 03/07/2020).

[Abo+10] Mohamed Abomhara, Othman Khalifa, Oula Zakaria, A. Zaidan, Bilal Bahaa, and Rame A. "Video compression techniques: An overview". In: *Journal of Applied Sciences(Faisalabad)* 10.16 (2010), pp. 1834–1840.

[AE18] Pinar Akyazi and Touradj Ebrahimi. "Comparison of Compression Efficiency between HEVC/H. 265, VP9 and AV1 based on Subjective Quality Assessments". In: *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2018, pp. 1–6.

[AMK15] Umar Albalawi, Saraju P Mohanty, and Elias Kougianos. "A hardware architecture for better portable graphics (BPG) compression encoder". In: *2015 IEEE International Symposium on Nanoelectronic and Information Systems*. IEEE. 2015, pp. 291–296.

[App06] Thomas H Apperley. "Genre and game studies: Toward a critical approach to video game genres". In: *Simulation & Gaming* 37.1 (2006), pp. 6–23.

[ASG15] Tunç Ozan Aydın, Aljoscha Smolic, and Markus Gross. "Automated aesthetic analysis of photographic images". In: *IEEE transactions on visualization and computer graphics* 21.1 (2015), pp. 31–42.

[AVR] **AVRateNG**. *AVRateNG – github project*. URL: https://github.com/Telecommunication-Telemedia-Assessment/avrateNG (visited on 03/07/2020).

[Bam+17] Christos G Bampis, Praful Gupta, Rajiv Soundararajan, and Alan C Bovik. "SpEED-QA: Spatial efficient entropic differencing for image and video quality". In: *IEEE signal processing letters* 24.9 (2017), pp. 1333–1337.

[Bam+18] Christos G. Bampis, Zhi Li, Ioannis Katsavounidis, and Alan C. Bovik. "Recurrent and Dynamic Models for Predicting Streaming Video Quality of Experience". In: *IEEE Transactions on Image Processing* 27.7 (2018), pp. 3316–3331. DOI: 10.1109/TIP.2018.2815842.

[Bar+15] Marcus Barkowsky, Iñigo Sedano, Kjell Brunnström, Mikołaj Leszczuk, and Nicolas Staelens. "Hybrid video quality prediction: reviewing video quality measurement for widening application scope". In: *Multimedia Tools and Applications* 74.2 (2015), pp. 323–343.

[Bar+18a] Nabajeet Barman, Steven Schmidt, Saman Zadtootaghaj, Maria G Martini, and Sebastian Möller. "An evaluation of video quality assessment metrics for passive gaming video streaming". In: *Proceedings of the 23rd Packet Video Workshop*. ACM. 2018, pp. 7–12.

[Bar+18b] Nabajeet Barman, Saman Zadtootaghaj, Steven Schmidt, Maria G Martini, and Sebastian Möller. "GamingVideoSET: a dataset for gaming video streaming applications". In: *2018 16th Annual Workshop on Network and Systems Support for Games (NetGames)*. IEEE. 2018, pp. 1–6.

[Bar+19] Nabajeet Barman, Emmanuel Jammeh, Seyed Ali Ghorashi, and Maria G Martini. "No-reference video quality estimation based on machine learning for passive gaming video streaming applications". In: *IEEE Access* 7 (2019), pp. 74511–74527.

[Bel18] Fabrice Bellard. *BPG Image format*. 2018. URL: https://bellard.org/bpg/ (visited on 03/07/2020).

[Ber+15] Kongfeng Berger, Yao Koudota, Marcus Barkowsky, and Patrick Le Callet. "Subjective quality assessment comparing UHD and HD resolution in HEVC transmission chains". In: *Quality of Multimedia Experience (QoMEX), 2015 Seventh International Workshop on*. IEEE. 2015, pp. 1–6.

[BK97] Vasudev Bhaskaran and Konstantinos Konstantinides. *Image and video compression standards: algorithms and architectures*. Vol. 408. Springer Science & Business Media, 1997.

[Blea] Blender Foundation. *Bick Buck Bunny RAW Distribution*. URL: http://distribution.bbb3d.renderfarming.net/video/ (visited on 08/09/2021).

[Bleb] Blender Foundation. *Sintel, the Durian Open Movie Project*. URL: https://media.xiph.org/sintel/sintel-4k-tiff16/.

[BLR14] O. Bousquet, U. von Luxburg, and G. Ratsch. *Advanced Lectures on Machine Learning*. Lecture notes in computer science. Springer, 2014. ISBN: 9783662185483. URL: https://books.google.de/books?id=h5MmswEACAAJ.

[BM19] Nabajeet Barman and Maria G Martini. "QoE Modeling for HTTP Adaptive Video Streaming–A Survey and Open Challenges". In: *IEEE Access* 7 (2019), pp. 30831–30859.

[BM20] Nabajeet Barman and Maria Martini. "An evaluation of the next-generation image coding standard AVIF". In: *Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. Athlone, Ireland, May 2020, pp. 1–4.

[Bos+16a] Sebastian Bosse, Dominique Maniry, Klaus-Robert Müller, Thomas Wiegand, and Wojciech Samek. "Deep Neural Networks for No-Reference and Full-Reference Image Quality Assessment". In: (2016).

[Bos+16b] Sebastian Bosse, Dominique Maniry, Klaus-Robert Müller, Thomas Wiegand, and Wojciech Samek. "Neural network-based full-reference image quality assessment". In: *Picture Coding Symposium (PCS), 2016*. IEEE. 2016, pp. 1–5.

[Bos+17] Sebastian Bosse, Dominique Maniry, Klaus-Robert Müller, Thomas Wiegand, and Wojciech Samek. "Deep neural networks for no-reference and full-reference image quality assessment". In: *IEEE Transactions on Image Processing* 27.1 (2017), pp. 206–219.

[Bra00] G. Bradski. "The OpenCV Library". In: *Dr. Dobb's Journal of Software Tools* (2000).

[Bre01] Leo Breiman. "Random forests". In: *Machine learning* 45.1 (2001), pp. 5–32.

[Bru+13] Kjell Brunnström, Sergio Ariel Beker, Katrien De Moor, Ann Dooms, Sebastian Egger, Marie-Neige Garcia, Tobias Hossfeld, Satu Jumisko-Pyykkö, Christian Keimel, Mohamed-Chaker Larabi, et al. "Qualinet white paper on definitions of quality of experience". In: (2013).

[Cha+18] Haw-Shiuan Chang, Chih-Fan Hsu, Tobias Hoßfeld, and Kuan-Ta Chen. "Active learning for crowdsourced QoE modeling". In: *IEEE Transactions on Multimedia* 20.12 (2018), pp. 3337–3352.

[Che+17] Chao Chen, Yao-Chung Lin, Anil Kokaram, and Steve Benting. "Encoding bitrate optimization using playback statistics for HTTP-based adaptive video streaming". In: *arXiv preprint arXiv:1709.08763* (2017).

[Che+20] Li-Heng Chen, Christos G. Bampis, Zhi Li, Andrey Norkin, and Alan C. Bovik. *Perceptually Optimizing Deep Image Compression*. 2020. arXiv: 2007.02711 [eess.IV].

[Cho] Francois Chollet. *Transfer learning and fine-tuning*. URL: https://keras.io/guides/transfer_learning/ (visited on 03/07/2021).

[Cho+15] François Chollet et al. *Keras*. https://github.com/keras-team/keras. 2015.

[Cho16] François Chollet. "Xception: Deep Learning with Depthwise Separable Convolutions". In: *CoRR* (2016).

[Cho17] François Chollet. "Xception: Deep learning with depthwise separable convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.

[Cis18] Cisco. *Cisco visual networking index: Forecast and trends, 2017–2022 (White Paper)*. 2018.

[CL17] Manri Cheon and Jong-Seok Lee. "Subjective and objective quality assessment of compressed 4K UHD videos for immersive experience". In: *IEEE Transactions on Circuits and Systems for Video Technology* 28.7 (2017), pp. 1467–1480.

[Dia18] Anton Garcia Diaz. *Picture this: the best image format for the web in 2019*. 2018. URL: https://www.freecodecamp.org/news/best-image-format-for-web-in-2019-jpeg-webp-heic-avif-41ba0c1b2789/ (visited on 03/07/2020).

[DJ94] David L Donoho and Jain M Johnstone. "Ideal spatial adaptation by wavelet shrinkage". In: *biometrika* 81.3 (1994), pp. 425–455.

[DMW16] Prajna Paramita Dash, Akshaya Mishra, and Alexander Wong. "Deep Quality: A Deep No-reference Quality Assessment System". In: (2016).

[DWM17] Prajna Paramita Dash, Alexander Wong, and Akshaya Mishra. "VeNICE: A very deep neural network approach to no-reference image assessment". In: *Industrial Technology (ICIT), 2017 IEEE International Conference on*. IEEE. 2017, pp. 1091–1096.

[Egg+15] Nathan E Egge, Jean-Marc Valin, Timothy B Terriberry, Thomas Daede, and Christopher Montgomery. "Using Daala intra frames for still picture coding". In: *Proceedings of Picture Coding Symposium*. 2015.

[Egi+06] Karen Egiazarian, Jaakko Astola, Nikolay Ponomarenko, Vladimir Lukin, Federica Battisti, and Marco Carli. "New full-reference quality metrics based on HVS". In: *Proceedings of the Second International Workshop on Video Processing and Quality Metrics*. Vol. 4. 2006.

[Eri15] Ericsson. *Understanding Ultra High Definition Television*. 2015. URL: https://www.ericsson.com/496d2f/assets/local/news/2015/4/wp-uhd.pdf.

[Est+96] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." In: *kdd*. Vol. 96. 34. 1996, pp. 226–231.

[Fec48] Gustav Theodor Fechner. "Elements of psychophysics, 1860." In: (1948).

[FHT10] Markus Fiedler, Tobias Hossfeld, and Phuoc Tran-Gia. "A generic quantitative relationship between quality of experience and quality of service". In: *IEEE Network* 24.2 (2010), pp. 36–41.

[Fre+19a] Stephan Fremerey, Frank Hofmeyer, **Steve Göring**, and Alexander Raake. "Impact of Various Motion Interpolation Algorithms on 360° Video QoE". In: *Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*. June 2019, pp. 1–3. DOI: 10.1109/QoMEX.2019.8743307.

[Fre+19b] Stephan Fremerey, Rachel Huang, **Steve Göring**, and Alexander Raake. "Are people pixel-peeping 360° videos?" In: *Electronic Imaging, Human Vision Electronic Imaging* (2019).

[Fre+20a] Stephan Fremerey, **Steve Göring**, Rakesh Ramachandra Rao Rao, Rachel Huang, and Alexander Raake. "Subjective Test Dataset and Meta-data-based Models for 360° Streaming Video Quality". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6.

[Fre+20b] Stephan Fremerey, Frank Hofmeyer, **Steve Göring**, Dominik Keller, and Alexander Raake. "Between the Frames - Evaluation of Various Motion Interpolation Algorithms to Improve 360° Video Quality". In: *22st IEEE International Symposium on Multimedia (IEEE ISM)*. Dec. 2020, pp. 1–8.

[GB15] Deepti Ghadiyaram and Alan C Bovik. "Massive online crowdsourced study of subjective and objective picture quality". In: *IEEE Transactions on Image Processing* 25.1 (2015), pp. 372–387.

[GB16] Deepti Ghadiyaram and Alan C Bovik. "Massive online crowdsourced study of subjective and objective picture quality". In: *IEEE Trans. Image Process* 25.1 (2016), pp. 372–387.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[GBR18] **Steve Göring**, Konstantin Brand, and Alexander Raake. "Extended Features using Machine Learning Techniques for Photo Liking Prediction". In: *Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. Sardinia, Italy, May 2018.

[GKR19] **Steve Göring**, Christopher Krämmer, and Alexander Raake. "cencro – Speedup of Video Quality Calculation using Center Cropping". In: *21st IEEE International Symposium on Multimedia (IEEE ISM)*. Dec. 2019, pp. 1–8.

[Goo20] Google. *A new image format for the Web*. 2020. URL: https://developers.google.com/speed/webp/ (visited on 03/07/2020).

[Gör+19] **Steve Göring**, Julian Zebelein, Simon Wedel, Dominik Keller, and Alexander Raake. "Analyze And Predict the Perceptibility of UHD Video Contents". In: *Electronic Imaging, Human Vision Electronic Imaging* 2019.12 (2019).

[Gör+20] **Steve Göring**, Robert Steger, Rakesh Ramachandra Rao Rao, and Alexander Raake. "Automated Genre Classification for Gaming Videos". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6.

[Gör+21a] **Steve Göring**, Rakesh Rao Ramachandra Rao, Bernhard Feiten, and Alexander Raake. "Modular Framework and Instances of Pixel-based Video Quality Models for UHD-1/4K". In: *IEEE Access* 9 (2021), pp. 31842–31864. DOI: 10.1109/ACCESS.2021.3059932. URL: https://ieeexplore.ieee.org/document/9355144.

[Gör+21b] **Steve Göring**, Rakesh Rao Ramachandra Rao, Stephan Fremerey, and Alexander Raake. "AVRate Voyager: an open source online testing platform". In: *2021 IEEE 23st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2021, pp. 1–6.

[GR18] **Steve Göring** and Alexander Raake. "deimeq – A Deep Neural Network Based Hybrid No-reference Image Quality Model". In: *7th European Workshop on Visual Information Processing (EUVIP)*. IEEE. Nov. 2018, pp. 1–6. DOI: 10.1109/EUVIP.2018.8611703.

[GR19] **Steve Göring** and Alexander Raake. "Evaluation of Intra-coding based image compression". In: *8th European Workshop on Visual Information Processing (EUVIP)*. IEEE. 2019, pp. 1–6.

[GR21] **Steve Göring** and Alexander Raake. "Rule of Thirds and Simplicity for Image Aesthetics using Deep Neural Networks". In: *2021 IEEE 23st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2021, pp. 1–6.

[GRF17] **Steve Göring**, Alexander Raake, and Bernhard Feiten. "A framework for QoE analysis of encrypted video streams". In: *Ninth International Conference on Quality of Multimedia Experience (QoMEX)*. May 2017, pp. 1–3. DOI: 10.1109/QoMEX.2017.7965640.

[GRP18] André F. R. Guarda, Nuno M. M. Rodrigues, and Fernando Pereira. "Deep Learning-based Point Cloud Coding: A Behavior And Performance Study". In: *2019 8th European Workshop on Visual Information Processing (EUVIP)*. IEEE. 2018, pp. 1–2.

[GRR19] **Steve Göring**, Rakesh Rao Ramachandra Rao, and Alexander Raake. "nofu - A Lightweight No-Reference Pixel Based Video Quality Model for Gaming Content". In: *Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*. Berlin, Germany, June 2019.

[GRR20] **Steve Göring**, Rakesh Rao Ramachandra Rao, and Alexander Raake. "Prenc – Predict Number Of Video Encoding Passes With Machine Learning". In: *Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. Athlone, Ireland, May 2020.

[GSR18] **Steve Göring**, Janto Skowronek, and Alexander Raake. "DeViQ – A deep no reference video quality model". In: *Electronic Imaging, Human Vision Electronic Imaging* 2018.14 (2018), pp. 1–6.

[Har] Harmonic. *Free 4K Demo Footage - Ultra HD Demo Footage*. URL: https://www.harmonicinc.com/4k-demo-footage-download/ (visited on 08/09/2021).

[He+15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep Residual Learning for Image Recognition". In: *CoRR* (2015).

[Hel+20] Christian R Helmrich, Mischa Siekmann, Sören Becker, Sebastian Bosse, Detlev Marpe, and Thomas Wiegand. "Xpsnr: A Low-Complexity Extension of The Perceptually Weighted Peak Signal-To-Noise Ratio For High-Resolution Video Quality Assessment". In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2020, pp. 2727–2731.

[Hos+13] Tobias Hossfeld, Christian Keimel, Matthias Hirth, Bruno Gardlo, Julian Habigt, Klaus Diepold, and Phuoc Tran-Gia. "Best practices for QoE crowdtesting: QoE assessment with crowdsourcing". In: *IEEE Transactions on Multimedia* 16.2 (2013), pp. 541–558.

[Hos+17a] Vlad Hosu, Franz Hahn, Mohsen Jenadeleh, Hanhe Lin, Hui Men, Tamás Szirányi, Shujun Li, and Dietmar Saupe. *The Konstanz Natural Video Database*. 2017. URL: http://database.mmsp-kn.de.

[Hos+17b] Vlad Hosu, Franz Hahn, Mohsen Jenadeleh, Hanhe Lin, Hui Men, Tamás Szirányi, Shujun Li, and Dietmar Saupe. "The Konstanz natural video database (KoNViD-1k)". In: *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2017, pp. 1–6.

[Hos+20] V. Hosu, H. Lin, T. Sziranyi, and D. Saupe. "KonIQ-10k: An Ecologically Valid Database for Deep Learning of Blind Image Quality Assessment". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 4041–4056.

[Hoß+11] Tobias Hoßfeld, Michael Seufert, Matthias Hirth, Thomas Zinner, Phuoc Tran-Gia, and Raimund Schatz. "Quantification of YouTube QoE via crowdsourcing". In: *2011 IEEE International Symposium on Multimedia*. IEEE. 2011, pp. 494–499.

[How+17] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications". In: *CoRR* (2017).

[HR99] Roelof Hamberg and Huib de Ridder. "Time-varying image quality: Modeling the relation between instantaneous and overall quality". In: *SMPTE journal* 108.11 (1999), pp. 802–811.

[HS03] David Hasler and Sabine E Suesstrunk. "Measuring colorfulness in natural images". In: *Human vision and electronic imaging VIII*. Vol. 5007. International Society for Optics and Photonics. 2003, pp. 87–96.

[HSE11] Tobias Hoßfeld, Raimund Schatz, and Sebastian Egger. "SOS: The MOS is not enough!" In: *2011 third international workshop on quality of multimedia experience*. IEEE. 2011, pp. 131–136.

[Hua+18] Ruochen Huang, Xin Wei, Liang Zhou, Chaoping Lv, Hao Meng, and Jiefeng Jin. "A survey of data-driven approach on multimedia QoE evaluation". In: *Frontiers of Computer Science* 12.6 (2018), pp. 1060–1075.

[ISO19] Information technology ISO/IEC 23009-1:2019. *Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats*. ISO/IEC 23009-1:2019, Information technology, 2019.

[ITU06] ITU-T. *Recommendation P. 800.1 Mean Opinion Score (MOS) terminology*. 2006.

[ITU08a] ITU-T. *ITU-T Rec. E.800 (09/08)*. Tech. rep. Int. Telecommunication Union, 2008.

[ITU08b] ITU-T. *Subjective video quality assessment methods for multimedia applications*. Serie P: Telephone Transmission Quality, Telephone Installations, Local Line Networks. Vol. Recommendation ITU-T P.910. International Telecommunication Union. Geneva, 2008.

[ITU14a] ITU-T. *P.1401 : Methods, metrics and procedures for statistical evaluation, qualification and comparison of objective quality prediction models*. Tech. rep. Int. Telecommunication Union, 2014.

[ITU14b] ITU-T. *Recommendation ITU-R BT.500-13 – Methodology for the subjective assessment of the quality of television pictures*. Tech. rep. International Telecommunication Union, 2014.

[ITU15] ITU-R. *Recommendation BT.2020-2: Parameter values for ultra-high definition television systems for production and international programme exchange*. Oct. 2015.

[ITU16a] ITU-T. *ITU-T Rec. G.1022 (07/16)*. Tech. rep. Int. Telecommunication Union, 2016.

[ITU16b] ITU-T. *ITU-T Rec. P.913 (16/03)*. Tech. rep. Int. Telecommunication Union, 2016.

[ITU17] ITU-T. *ITU-T Rec. P.1203 (10/17)*. Tech. rep. Int. Telecommunication Union, 2017.

[ITU19a] ITU-T. *Recommendation P.1204 - Video quality assessment of streaming services over reliable transport for resolutions up to 4K*. Tech. rep. International Telecommunication Union, 2019.

[ITU19b] ITU-T. *Recommendation P.1204.3 - Video quality assessment of streaming services over reliable transport for resolutions up to 4K with access to full bitstream information*. Tech. rep. International Telecommunication Union, 2019.

[ITU19c] ITU-T. *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*. Tech. rep. International Telecommunication Union, 2019.

[ITU19d] ITU-T. *Recommendation P.1204.5 : Video quality assessment of streaming services over reliable transport for resolutions up to 4K with access to transport and received pixel information*. Tech. rep. International Telecommunication Union, 2019.

[ITU20] ITU-T. *Recommendation H.266 (08/20) - Versatile video coding*. Tech. rep. International Telecommunication Union, 2020.

[Kan+14] Le Kang, Peng Ye, Yi Li, and David Doermann. "Convolutional neural networks for no-reference image quality assessment". In: *CVPR*. 2014, pp. 1733–1740.

[Kar+17] Peter A Kara, Werner Robitza, Alexander Raake, and Maria G Martini. "The label knows better: the impact of labeling effects on perceived quality of HD and UHD video streaming". In: *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2017, pp. 1–6.

[Kar+19] Peter A Kara, Werner Robitza, Nikolett Pinter, Maria G Martini, Alexander Raake, and Aniko Simon. "Comparison of HD and UHD video quality with and without the influence of the labeling effect". In: *Quality and User Experience* 4.1 (2019), p. 4.

[KAR15] Ioannis Katsavounidis, Anne Aaron, and David Ronca. "Native resolution detection of video sequences". In: *Annual Technical Conference and Exhibition, SMPTE 2015*. SMPTE. 2015, pp. 1–20.

[Kat18] Ioannis Katsavounidis. *Dynamic Optimizer–A Perceptual Video Encoding Optimization Framework*. 2018. URL: https://netflixtechblog.com/dynamic-optimizer-a-perceptual-video-encoding-optimization-framework-e19f1e3a277f (visited on 03/07/2020).

[KCR18] Shiba Kuanar, Christopher Conly, and KR Rao. "Deep learning based HEVC in-loop filtering for decoder quality enhancement". In: *2018 Picture Coding Symposium (PCS)*. IEEE. 2018, pp. 164–168.

[Kel+21] Dominik Keller, Markus Vaalgamaa, Erkki Paajanen, Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Groovability: Using Groove as a Novel Measure for Audio QoE with the Example of Smartphones". In: *13th International Conference on Quality of Multimedia Experience (QoMEX)*. 2021.

[KN10] Gunnar Kreitz and Fredrik Niemela. "Spotify–large scale, low latency, P2P music-on-demand streaming". In: *2010 IEEE Tenth International Conference on Peer-to-Peer Computing (P2P)*. IEEE. 2010, pp. 1–10.

[KNK12] Sajad Khorsandroo, Rafidah Md Noor, and Sayid Khorsandroo. "A generic quantitative relationship between quality of experience and packet loss in video streaming services". In: *2012 fourth international conference on ubiquitous and future networks (ICUFN)*. IEEE. 2012, pp. 352–356.

[Kra91] Mark A Kramer. "Nonlinear principal component analysis using autoassociative neural networks". In: *AIChE journal* 37.2 (1991), pp. 233–243.

[Kua+19] Shiba Kuanar, KR Rao, Monalisa Bilas, and Jonathan Bredow. "Adaptive CU mode selection in HEVC intra prediction: A deep learning approach". In: *Circuits, Systems, and Signal Processing* 38.11 (2019), pp. 5081–5102.

[Lai+16] Jani Lainema, Miska M Hannuksela, Vinod K Malamal Vadakital, and Emre B Aksu. "HEVC still image coding and high efficiency image file format". In: *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2016, pp. 71–75.

[Lap+16] Valero Laparra, Johannes Ballé, Alexander Berardino, and Eero P Simoncelli. "Perceptual image quality assessment using a normalized Laplacian pyramid". In: *Electronic Imaging* 2016.16 (2016), pp. 1–6.

[LC10] Eric C Larson and Damon M Chandler. "Most apparent distortion: full-reference image quality assessment and the role of strategy". In: *Journal of Electronic Imaging* 19.1 (2010), pp. 011006–011006.

[Led+04] Helmut Leder, Benno Belke, Andries Oeberst, and Dorothee Augustin. "A model of aesthetic appreciation and aesthetic judgments". In: *British journal of psychology* 95.4 (2004), pp. 489–508.

[Lee+20] Wei-Cheng Lee, Chih-Peng Chang, Wen-Hsiao Peng, and Hsueh-Ming Hang. "A Hybrid Layered Image Compressor with Deep-Learning Technique". In: *2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6.

[Lee07] Chae-Sub Lee. "IPTV over next generation networks in ITU-T". In: *2007 2nd IEEE/IFIP International Workshop on Broadband Convergence Networks*. IEEE. 2007, pp. 1–18.

[Lev44] Kenneth Levenberg. "A method for the solution of certain non-linear problems in least squares". In: *Quarterly of applied mathematics* 2.2 (1944), pp. 164–168.

[LH08] Maaten LJPvd and GE Hinton. "Visualizing high-dimensional data using t-SNE". In: *J Mach Learn Res* 9.2579-2605 (2008), p. 5.

[LHS19] Hanhe Lin, Vlad Hosu, and Dietmar Saupe. "KADID-10k: A Large-scale Artificially Distorted IQA Database". In: *2019 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2019, pp. 1–3.

[LHS20] Hanhe Lin, Vlad Hosu, and Dietmar Saupe. "DeepFL-IQA: Weak Supervision for Deep IQA Feature Learning". In: *arXiv preprint arXiv:2001.08113* (2020).

[Li+11] Songnan Li, Fan Zhang, Lin Ma, and King Ngi Ngan. "Image quality assessment by separately evaluating detail losses and additive impairments". In: *IEEE Trans. on Multimedia* 13.5 (2011), pp. 935–949.

[Li+14] Jing Li, Yao Koudota, Marcus Barkowsky, Hélène Primon, and Patrick Le Callet. *Comparing upscaling algorithms from HD to Ultra HD by evaluating preference of experience*. LUNAM Université, Université de Nantes, 2014.

[Lin+14] J. Y. Lin, T. J. Liu, E. C. H. Wu, and C. C. J. Kuo. "A fusion-based video quality assessment (fvqa) index". In: *APSIPA, 2014 Asia-Pacific*. Dec. 2014, pp. 1–5. DOI: 10.1109/APSIPA.2014.7041705.

[Lin+15] Kevin Lin, Huei-Fang Yang, Jen-Hao Hsiao, and Chu-Song Chen. "Deep learning of binary hash codes for fast image retrieval". In: *Proc. of the IEEE conf. on computer vision and pattern recognition workshops*. 2015, pp. 27–35.

[Lin+20] Jianping Lin, Mohammad Akbari, Haisheng Fu, Qian Zhang, Shang Wang, Jie Liang, Dong Liu, Feng Liang, Guohe Zhang, and Chengjie Tu. "Learned Variable-Rate Multi-Frequency Image Compression using Modulated Generalized Octave Convolution". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6.

[Lin10] Nam Ling. "Expectations and challenges for next generation video compression". In: *2010 5th IEEE Conference on Industrial Electronics and Applications*. IEEE. 2010, pp. 2339–2344.

[Liu+16] Lixiong Liu, Yi Hua, Qingjie Zhao, Hua Huang, and Alan Conrad Bovik. "Blind image quality assessment by relative gradient statistics and adaboosting neural network". In: *Signal Processing: Image Communication* 40 (2016), pp. 1–15.

[Liu+20] Dong Liu, Yue Li, Jianping Lin, Houqiang Li, and Feng Wu. "Deep learning-based video coding: A review and a case study". In: *ACM Computing Surveys (CSUR)* 53.1 (2020), pp. 1–35.

[LMP+12] Patrick Le Callet, Sebastian Möller, Andrew Perkis, et al. "Qualinet white paper on definitions of quality of experience". In: *European network on quality of experience in multimedia systems and services (COST Action IC 1003)* 3.2012 (2012).

[Loh+11] Thorsten Lohmar, Torbjörn Einarsson, Per Fröjdh, Frédéric Gabin, and Markus Kampmann. "Dynamic adaptive HTTP streaming of live content". In: *2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*. IEEE. 2011, pp. 1–8.

[LSD15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation". In: *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*. 2015, pp. 3431–3440.

[Lu+15] Xin Lu, Zhe Lin, Hailin Jin, Jianchao Yang, and James Z Wang. "Rating image aesthetics using deep learning". In: *IEEE Trans. on Multimedia* 17.11 (2015), pp. 2021–2034.

[Mac+67] James MacQueen et al. "Some methods for classification and analysis of multivariate observations". In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[MB10] Anush K Moorthy and Alan C Bovik. "A two-stage framework for blind image quality assessment". In: *2010 IEEE International Conference on Image Processing*. IEEE. 2010, pp. 2481–2484.

[Mic19] Franck Michel. *Flickr may have lost 63% of its photos after being acquired by SmugMug*. 2019. URL: https://www.flickr.com/photos/franckmichel/6855169886 (visited on 03/07/2020).

[MLS18] Hui Men, Hanhe Lin, and Dietmar Saupe. "Spatiotemporal Feature Combination Model for No-Reference Video Quality Assessment". In: *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2018, pp. 1–3.

[MMB12] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. “No-reference image quality assessment in the spatial domain”. In: *IEEE Transactions on Image Processing* 21.12 (2012), pp. 4695–4708.

[MR14] Sebastian Möller and Alexander Raake. *Quality of experience: advanced concepts, applications and methods*. Springer, 2014.

[MRT18] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[MSB13] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. “Making a "completely blind" image quality analyzer”. In: *IEEE Signal Processing Letters* 20.3 (2013), pp. 209–212.

[Nad+20] Babak Naderi, Rafael Zequeira Jiménez, Matthias Hirth, Sebastian Möller, Florian Metzger, and Tobias Hoßfeld. “Towards speech quality assessment using a crowdsourcing approach: evaluation of standardized methods”. In: *Quality and User Experience* 6.1 (2020), pp. 1–21.

[Nas] Nasa. *Bennu's Journey*. URL: https://svs.gsfc.nasa.gov/20220.

[NAS17] NASA. *Ultra high definition video gallery*. 2017. URL: https://www.nasa.gov/content/ultra-high-definition-video-gallery (visited on 08/09/2021).

[Nee20] Much Needed. *Netflix by the Numbers: Statistics, Demographics, and Fun Facts*. 2020. URL: https://muchneeded.com/netflix-statistics/ (visited on 03/07/2020).

[Net] Netflix. *Netflix VMAF, https://github.com/Netflix/vmaf; 2019-07-03*.

[Net18a] Netflix. *4K Support*. 2018. URL: https://help.netflix.com/en/node/13444 (visited on 03/07/2020).

[Net18b] Netflix. *VMAF 4K included*. 2018. URL: https://github.com/Netflix/vmaf (visited on 03/07/2020).

[Net21] Netflix. *VMAF subsample*. 2021. URL: https://github.com/Netflix/vmaf/blob/master/resource/doc/vmafossexec.md (visited on 10/03/2021).

[NHK20] NHK. *NHK 8K/4K Broadcast*. 2020. URL: https://www.nhk.or.jp/bs4k8k (visited on 03/07/2020).

[NM14] Tung Nguyen and Detlev Marpe. "Objective performance evaluation of the HEVC main still picture profile". In: *IEEE Transactions on circuits and systems for video technology* 25.5 (2014), pp. 790–797.

[Nok20] Nokiatech. *About HEIF and MIAF*. 2020. URL: https://nokiatech.github.io/heif/ (visited on 03/07/2020).

[Ope19] The Alliance for Open Media. *AV1 Image File Format (AVIF)*. 2019. URL: https://aomediacodec.github.io/av1-avif/ (visited on 03/07/2020).

[Ope20] The Alliance for Open Media. *AV1 Features*. 2020. URL: https://aomedia.org/av1-features/ (visited on 03/07/2020).

[Ord+19] Marta Orduna, César Díaz, Lara Muñoz, Pablo Pérez, Ignacio Benito, and Narciso García. "Video Multimethod Assessment Fusion (VMAF) on 360VR contents". In: *arXiv preprint arXiv:1901.06279* (2019).

[OS20] Irena Orsolic and Lea Skorin-Kapov. "A framework for in-network qoe monitoring of encrypted video streaming". In: *IEEE Access* 8 (2020), pp. 74691–74706.

[OT94] Kiyotaka Otsuji and Yoshinobu Tonomura. "Projection-detecting filter for video cut detection". In: *Multimedia Systems* 1.5 (1994), pp. 205–210.

[Pan] Jurre Pannekeet. *Tournaments for the West's Four Biggest Esports Games*. URL: https://newzoo.com/insights/articles/tournaments-for-%20the-wests-four-biggest-esports-games-generated-190-1-million-%20hours-of-viewership/ (visited on 06/05/2020).

[Ped+11] 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". In: *JMLR* 12 (2011), pp. 2825–2830.

[Per14] Cristian Perra. "A low computational complexity blockiness estimation based on spatial analysis". In: *Telecommunications Forum Telfor (TELFOR), 2014 22nd*. IEEE. 2014, pp. 1130–1133.

[Pet+19] Stefano Petrangeli, Gwendal Simon, Haoliang Wang, and Vishy Swaminathan. "Dynamic Adaptive Streaming for Augmented Reality Applications". In: *21st IEEE International Symposium on Multimedia (2019 IEEE ISM)*. Dec. 2019.

[PM11] R. Pantos and W.M. May. *HTTP Live Streaming*. 2011. URL: https://tools.ietf.org/html/draft-pantos-http-live-streaming-13.

[Pon+15] Nikolay Ponomarenko, Lina Jin, Oleg Ieremeiev, Vladimir Lukin, Karen Egiazarian, Jaakko Astola, Benoit Vozel, Kacem Chehdi, Marco Carli, Federica Battisti, and C. C. Jay Kuo. "Image database TID2013: Peculiarities, results and perspectives". In: *Signal Processing: Image Communication* 30 (Jan. 2015), pp. 57–77. ISSN: 0923-5965. DOI: 10.1016/j.image.2014.10.009.

[PVZ15] Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman. "Deep face recognition". In: *BMVC*. Vol. 1. 3. 2015, p. 6.

[PW03] Margaret H Pinson and Stephen Wolf. "Comparing subjective video quality testing methodologies". In: *Visual Communications and Image Processing 2003*. Vol. 5150. International Society for Optics and Photonics. 2003, pp. 573–582.

[QTG10] MT Qadri, KT Tan, and M Ghanbari. "Frequency domain blockiness measurement for image quality assessment". In: *Computer Technology and Development (ICCTD), 2010 2nd International Conference on*. IEEE. 2010, pp. 282–285.

[Qui86] J. Ross Quinlan. "Induction of decision trees". In: *Machine learning* 1.1 (1986), pp. 81–106.

[Raa+17] Alexander Raake, Marie-Neige Garcia, Werner Robitza, Peter List, **Steve Göring**, and Bernhard Feiten. "A bitstream-based, scalable video-quality model for HTTP adaptive streaming: ITU-T P.1203.1". In: *Ninth International Conference on Quality of Multimedia Experience (QoMEX)*. May 2017, pp. 1–6. DOI: 10.1109/QoMEX.2017.7965631.

[Raa+20] Alexander Raake, Silvio Borer, Shahid Satti, Jörgen Gustafsson, Rakesh Rao Ramachandra Rao, Stefano Medagli, Peter List, **Steve Göring**, David Lindero, Werner Robitza, Gunnar Heikkilä, Simon Broom, Christian Schmidmer, Bernhard Feiten, Ulf Wüstenhagen, Thomas Wittmann, Matthias Obermann, and Roland Bitto. "Multi-model standard for bitstream-, pixel-based and hybrid video quality assessment of UHD/4K: ITU-T P.1204". In: *IEEE Access* 8 (2020), pp. 193020–193049. DOI: 10.1109/ACCESS.2020.3032080. URL: https://ieeexplore.ieee.org/document/9234526?source=authoralert.

[Rao+19a] Rakesh Rao Ramachandra Rao, **Steve Göring**, Werner Robitza, Bernhard Feiten, and Alexander Raake. "AVT-VQDB-UHD-1: A Large Scale Video Quality Database for UHD-1". In: *21st IEEE International Symposium on Multimedia (IEEE ISM)*. Dec. 2019, pp. 1–8.

[Rao+19b] Rakesh Rao Ramachandra Rao, **Steve Göring**, Patrick Vogel, Nicolas Pachatz, Juan Jose Villamar Villarreal, Werner Robitza, Peter List, Bernhard Feiten, and Alexander Raake. "Adaptive video streaming with current codecs and formats: Extensions to parametric video quality model ITU-T P.1203". In: *Electronic Imaging* (2019).

[Rao+20a] Rakesh Ramachandra Rao Rao, **Steve Göring**, Robert Steger, Saman Zadtootaghaj, Nabajeet Barman, Stephan Fremerey, Sebastian Möller, and Alexander Raake. "A Large-scale Evaluation of the bitstream-based video-quality model ITU-T P.1204.3 on Gaming Content". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6.

[Rao+20b] Rakesh Rao Ramachandra Rao, **Steve Göring**, Peter List, Werner Robitza, Bernhard Feiten, Ulf Wüstenhagen, and Alexander Raake. "Bitstream-based Model Standard for 4K/UHD: ITU-T P.1204.3 – Model Details, Evaluation, Analysis and Open Source Implementation". In: *Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. Athlone, Ireland, May 2020.

[RE14] Alexander Raake and Sebastian Egger. "Quality and quality of experience". In: *Quality of experience*. Springer, 2014, pp. 11–33.

[ŘE14] Martin Řeřábek and Touradj Ebrahimi. "Comparison of compression efficiency between HEVC/H. 265 and VP9 based on subjective assessments". In: *Applications of Digital Image Processing XXXVII*. Vol. 9217. International Society for Optics and Photonics. 2014, 92170U.

[Rec08] ITUT Recommendation. "P. 910,Subjective video quality assessment methods for multimedia applications," in: *International Telecommunication Union, Tech. Rep* (2008).

[Rei+10] Peter Reichl, Sebastian Egger, Raimund Schatz, and Alessandro D'Alconzo. "The logarithmic nature of QoE and the role of the Weber-Fechner law in QoE assessment". In: *2010 IEEE International Conference on Communications*. IEEE. 2010, pp. 1–5.

[Rei+14] Ulrich Reiter, Kjell Brunnström, Katrien De Moor, Mohamed-Chaker Larabi, Manuela Pereira, Antonio Pinheiro, Junyong You, and Andrej Zgank. "Factors influencing quality of experience". In: *Quality of experience*. Springer, 2014, pp. 55–72.

[RGR21a] Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Enhancement of Pixel-based Video Quality Models using Meta-data". In: *Electronic Imaging, Human Vision Electronic Imaging*. 2021.

[RGR21b] Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Towards High Resolution Video Quality Assessment in the Crowd". In: *13th International Conference on Quality of Multimedia Experience (QoMEX)*. 2021.

[Rob+17] Werner Robitza, Arslan Ahmad, Peter A. Kara, Luigi Atzori, Maria G. Martini, Alexander Raake, and Lingfen Sun. "Challenges of future multimedia QoE monitoring for internet service providers". In: *Multimedia Tools and Applications* 76.21 (Nov. 2017), pp. 22243–22266. DOI: 10.1007/s11042-017-4870-z. URL: https://doi.org/10.1007/s11042-017-4870-z.

[Rob+18a] Werner Robitza, **Steve Göring**, Alexander Raake, David Lindegren, Gunnar Heikkilä, Jörgen Gustafsson, Peter List, Bernhard Feiten, Ulf Wüstenhagen, Marie-Neige Garcia, Kazuhisa Yamagishi, and Simon Broom. "HTTP Adaptive Streaming QoE Estimation with ITU-T Rec. P.1203 – Open Databases and Software". In: *9th ACM Multimedia Systems Conference*. Amsterdam, 2018. ISBN: 9781450351928. DOI: 10.1145/3204949.3208124.

[Rob+18b] Werner Robitza, Dhananjaya G Kittur, Alexander M Dethof, **Steve Göring**, Bernhard Feiten, and Alexander Raake. "Measuring YouTube QoE with ITU-T P. 1203 under Constrained Bandwidth Conditions". In: *Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2018, pp. 1–6.

[Rob+20] Werner Robitza, Alexander M. Dethof, **Steve Göring**, Alexander Raake, Tim Polzehl, and Andre Beyer. "Are You Still Watching? Streaming Video Quality and Engagement Assessment in the Crowd". In: *Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. Athlone, Ireland, May 2020.

[Rob+21] Werner Robitza, Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Impact of Spatial and Temporal Information on Video Quality and Compressibility". In: *13th International Conference on Quality of Multimedia Experience (QoMEX)*. 2021.

[Rus+15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3 (2015), pp. 211–252. DOI: 10.1007/s11263-015-0816-y.

[RZM09] Snjezana Rimac-Drlje, Drago Zagar, and Goran Martinovic. “Spatial masking and perceived video quality in multimedia applications”. In: *2009 16th International Conference on Systems, Signals and Image Processing*. IEEE. 2009, pp. 1–4.

[Sac+12] Andreas Sackl, Sebastian Egger, Patrick Zwickl, and Peter Reichl. “The QoE alchemy: Turning quality into money. Experiences with a refined methodology for the evaluation of willingness-to-pay for service quality”. In: *2012 Fourth International Workshop on Quality of Multimedia Experience*. IEEE. 2012, pp. 170–175.

[Sam18] Samsung. *Future of display*. [22.10.2019]. 2018. URL: https://news.samsung.com/global/ifa-docent-series-part-1-tv-as-the-lifestyle-screen-the-future-of-display.

[Sam20] Samsung. *Samsung S20 Specification*. 2020. URL: https://www.samsung.com/us/mobile/galaxy-s20-5g/camera/ (visited on 03/07/2020).

[San+18] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. “Mobilenetv2: Inverted residuals and linear bottlenecks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 4510–4520.

[SB06] Hamid R Sheikh and Alan C Bovik. “Image information and visual quality”. In: *IEEE Trans. Image Process.* 15.2 (2006), pp. 430–444.

[SB12] Rajiv Soundararajan and Alan C Bovik. “Video quality assessment by reduced reference spatio-temporal entropic differencing”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 23.4 (2012), pp. 684–694.

[Sch+18] Anika Schwind, Florian Wamser, Thomas Gensler, Phuoc Tran-Gia, Michael Seufert, and Pedro Casas. “Streaming characteristics of spotify sessions”. In: *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2018, pp. 1–6.

[Sha+14] Muhammad Shahid, Andreas Rossholm, Benny Lövström, and Hans-Jürgen Zepernick. “No-reference image and video quality assessment: a classification and review of recent approaches”. In: *EURASIP Journal on Image and Video Processing* 2014.1 (Aug. 14, 2014), p. 40. ISSN: 1687-5281. DOI: 10.1186/1687-5281-2014-40. URL: https://doi.org/10.1186/1687-5281-2014-40.

[Sha17] Aadil Shadman. *WhatsApp Has Started Ruining Your Images*. 2017. URL: https://propakistani.pk/2017/04/17/whatsapp-started-ruining-images/ (visited on 03/07/2020).

[She+16] Hamid R Sheikh, Zhou Wang, Lawrence Cormack, and Alan C Bovik. *LIVE image quality assessment database release 2 (2005)*. 2016.

[Sin+19a] Ashutosh Singla, **Steve Göring**, Alexander Raake, Rob Koenen, Britta Meixner, and Thomas Buchholz. "Subjective Quality Evaluation of Tile-based Streaming for Omnidirectional Videos". In: *10th ACM Multimedia Systems Conference*. Amherst, MA, USA, 2019.

[Sin+19b] Ashutosh Singla, Rakesh Rao Ramachandra Rao, **Steve Göring**, and Alexander Raake. "Assessing Media QoE, Simulator Sickness and Presence for Omnidirectional Videos with Different Test Protocols". In: *26th IEEE Conference on Virtual Reality and 3D User Interfaces*. Osaka, Japan, Mar. 2019.

[Sin+21] Ashutosh Singla, **Steve Göring**, Dominik Keller, Rakesh Rao Ramachandra Rao, Stephan Fremerey, and Alexander Raake. "Assessment of the Simulator Sickness Questionnaire for Omnidirectional Videos". In: *IEEE Virtual Reality and 3D User Interfaces (VR)*. 2021. URL: https://conferences.computer.org/vrpub/pdfs/VR2021-2AyvgnPUHcYon9QQHz6BPD/255600a198/255600a198.pdf.

[SIV16] Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke. "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning". In: *CoRR* abs/1602.07261 (2016).

[Sku+17] Robert Skupin, Yago Sanchez, Y-K Wang, Miska M Hannuksela, J Boyce, and Mathias Wien. "Standardization status of 360 degree video coding and delivery". In: *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE. 2017, pp. 1–4.

[Spa16] Todd Spangler. *Amazon Prime Video Has 4 Times Netflix's Movie Lineup, But Size Isn't Everything*. 2016. URL: https://variety.com/2016/digital/news/netflix-amazon-prime-video-movies-tv-comparison-1201759030/ (visited on 03/07/2020).

[SRL98] H. Schulzrinne, A. Rao, and R. Lanphier. *Real Time Streaming Protocol (RTSP)*. 1998. URL: https://tools.ietf.org/html/rfc2326.

[SS08] Yun Q Shi and Huifang Sun. *Image and video compression for multimedia engineering: Fundamentals, algorithms, and standards*. CRC press, 2008.

[SSB06] Hamid R Sheikh, Muhammad F Sabir, and Alan C Bovik. "A statistical evaluation of recent full reference image quality assessment algorithms". In: *IEEE Transactions on image processing* 15.11 (2006), pp. 3440–3451.

[SSG18] Vida Fakour Sevom, Sebastian Schwarz, and Moncef Gabbouj. "Geometry-Guided 3D Data Interpolation for Projection-Based Dynamic Point Cloud Coding". In: *2018 7th European Workshop on Visual Information Processing (EUVIP)*. IEEE. 2018, pp. 1–6.

[Sto+11] Thomas Stockhammer et al. "Dynamic adaptive streaming over HTTP-design principles and standards". In: *Qualcomm Incorporated* (2011), pp. 1–3.

[SV08] Alex Smola and SVN Vishwanathan. "Introduction to machine learning". In: *Cambridge University, UK* 32.34 (2008), p. 2008.

[SZ14] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *CoRR* (2014).

[Sze+15] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. "Rethinking the Inception Architecture for Computer Vision". In: *CoRR* (2015).

[Sze+16] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2818–2826.

[SZM20] Michael D Smith, Michael Zink, and Aansh Malik. "Tested Perceptual Difference Between UHD-1/4K and UHD-2/8K". In: *SMPTE Motion Imaging Journal* 129.6 (2020), pp. 43–51.

[TE93] Robert J Tibshirani and Bradley Efron. "An introduction to the bootstrap". In: *Monographs on statistics and applied probability* 57 (1993), pp. 1–436.

[Tho+15] Emmanuel Thomas, MO van Deventer, Thomas Stockhammer, Ali C Begen, and Jeroen Famaey. "Enhancing MPEG DASH performance via server and network assistance". In: (2015).

[Tho+16] Emmanuel Thomas, MO van Deventer, Thomas Stockhammer, Ali C Begen, M-L Champel, and Ozgur Oyman. "Applications and deployments of server and network assisted DASH (SAND)". In: (2016).

[TM18] Hossein Talebi and Peyman Milanfar. "NIMA: Neural image assessment". In: *IEEE Trans. on Image Processing* 27.8 (2018), pp. 3998–4011.

[TS10] Lisa Torrey and Jude Shavlik. "Transfer learning". In: *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. IGI global, 2010, pp. 242–264.

[Tur16] Rasty Turek. *What YouTube Looks Like In A Day [Infographic]*. 2016. URL: https://medium.com/@synopsi/what-youtube-looks-like-in-a-day-infographic-d23f8156e599 (visited on 03/07/2020).

[Twi] TwitchTracker. *Twitch Viewers Statistics*. URL: https://twitchtracker.com/ (visited on 11/26/2019).

[Ul +20] Raza Ul Mustafa, Simone Ferlin, Christian Esteve Rothenberg, Darijo Raca, and Jason J. Quinlan. "A Supervised Machine Learning Approach for DASH Video QoE Prediction in 5G Networks". In: *Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks*. 2020, pp. 57–64.

[Uni12] International Telecommunication Union. *Press Release: Ultra High Definition Television: Threshold of a new age*. 2012. URL: https://www.itu.int/net/pressoffice/press_releases/2012/31.aspx#.UOB2JsxZL6p (visited on 03/07/2020).

[Utk+20] Markus Utke, Saman Zadtootaghaj, Steven Schmidt, Sebastian Bosse, and Sebastian Möller. "NDNetGaming-development of a no-reference deep CNN for gaming video quality prediction". In: *Multimedia Tools and Applications* (2020), pp. 1–23.

[Van+16] Glenn Van Wallendael, Paulien Coppensy, Tom Paridaens, Niels Van Kets, Wendy Van den Broecky, and Peter Lambert. *Perceptual Quality of 4K-resolution video content compared to HD*. Ghent University - iMinds - Data Science Lab, Belgium and Free University Brussels - iMinds - SMIT, Belgium, 2016.

[vau19] vau.net. *Media Usage 2018: Germans are using audiovisual media for more than nine hours each day*. 2019. URL: https://www.vau.net/pressemitteilungen/content/media-usage-2018-germans-using-audiovisual-media-than-nine-hours-day (visited on 03/07/2020).

[Wan+04] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. "Image quality assessment: from error visibility to structural similarity". In: *IEEE Trans. Image Process.* 13.4 (2004), pp. 600–612.

[Wan06] Yubing Wang. *Survey of objective video quality measurements*. 2006.

[Wei+14] Benjamin Weiss, Dennis Guse, Sebastian Möller, Alexander Raake, Adam Borowiak, and Ulrich Reiter. "Temporal development of quality of experience". In: *Quality of experience*. Springer, 2014, pp. 133–147.

[WIA19] Yilin Wang, Sasi Inguva, and Balu Adsumilli. "YouTube UGC Dataset for Video Compression Research". In: *arXiv preprint arXiv:1904.06457* (2019).

[Wie+18] Oliver Wiedemann, Vlad Hosu, Hanhe Lin, and Dietmar Saupe. "Disregarding the big picture: Towards local image quality assessment". In: *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE. 2018, pp. 1–6.

[Wik20] Wikipedia. *Wikipedia Database Download*. 2020. URL: https://en.wikipedia.org/wiki/Wikipedia:Database%5C_download (visited on 12/25/2019).

[WM08] Stefan Winkler and Praveen Mohandas. "The evolution of video quality measurement: From PSNR to hybrid metrics". In: *IEEE transactions on Broadcasting* 54.3 (2008), pp. 660–668.

[WSB03] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. "Multiscale structural similarity for image quality assessment". In: *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on*. Vol. 2. IEEE. 2003, pp. 1398–1402.

[WWC19] Sarah Wassermann, Nikolas Wehner, and Pedro Casas. "Machine learning models for YouTube QoE and user engagement prediction in smartphones". In: *ACM SIGMETRICS Performance Evaluation Review* 46.3 (2019), pp. 155–158.

[Yam+21] Kazuhisa Yamagishi, Noritssugu Egi, Noriko Yoshimura, and Pierre Lebreton. "Derivation Procedure of Coefficients of Metadata-based Model for Adaptive Bitrate Streaming Services". In: *IEICE Transactions on Communications* (2021), 2020CQP0002.

[YE13] Yildiray Yalman and Ismail Ertürk. "A new color image quality measure based on YUV transformation and PSNR for human vision system". In: *Turkish Journal of Electrical Engineering & Computer Sciences* 21.2 (2013), pp. 603–612.

[YKH09] Kazuhisa Yamagishi, Taichi Kawano, and Takanori Hayashi. "Hybrid video-quality-estimation model for IPTV services". In: *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*. IEEE. 2009, pp. 1–5.

[You15] YouTube. *Ten years of YouTube video tech in ten videos*. 2015. URL: https://youtube-eng.googleblog.com/2015/05/ten-years-of-youtube-video-tech-in-ten.html (visited on 03/07/2020).

[Zad+20a] Saman Zadtootaghaj, Nabajeet Barman, Rakesh Ramachandra Rao Rao, **Steve Göring**, Maria G. Martini, Alexander Raake, and Sebastian Möller. "DEMI: Deep Video Quality Estimation Model using Perceptual Video Quality Dimensions". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6.

[Zad+20b] Saman Zadtootaghaj, Steven Schmidt, Saeed Shafiee Sabet, Sebastian Moeller, and Carsten Griwodz. "Quality Estimation Models for Gaming Video Streaming Services Using Perceptual Video Quality Dimensions". In: *Proceedings of the 11th International Conference on Multimedia Systems*. ACM. 2020.

[Ziv04] Zoran Zivkovic. "Improved adaptive Gaussian mixture model for background subtraction". In: *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*. Vol. 2. IEEE. 2004, pp. 28–31.

[ZMM95] Ramin Zabih, Justin Miller, and Kevin Mai. *Feature-based algorithms for detecting and classifying scene breaks*. Tech. rep. Cornell University, 1995.

[Zou+20] Nannan Zou, Honglei Zhang, Francesco Cricri, Hamed Tavakoli, Jani Lainema, Miska Hannuksela, Emre Aksu, and Esa Rahtu. "L2C – Learning to Learn to Compress". In: *2020 IEEE 22st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2020, pp. 1–6.

[ZRB08] Slawomir Zielinski, Francis Rumsey, and Søren Bech. "On some biases encountered in modern audio quality listening tests-a review". In: *Journal of the Audio Engineering Society* 56.6 (2008), pp. 427–451.

[ZV06] Zoran Zivkovic and Ferdinand Van Der Heijden. "Efficient adaptive density estimation per image pixel for the task of background subtraction". In: *Pattern recognition letters* 27.7 (2006), pp. 773–780.

List of Figures

List of Tables

List of Acronyms

In the following table all used acronyms are listed.

ACR	Absolute Category Rating
CI	confidence interval
CNNs	convolutional neural networks
DASH	dynamic adaptive streaming using HTTP
DNN	deep neural network
FR	full-reference
GBC	gradient boosting classifier
GBR	gradient boosting regression
HAS	HTTP based adaptive streaming
HRCs	hypothetical reference circuits
HVS	Human Visual System
KNN	k-nearest neighbors algorithm
MOS	mean opinion score
MP	megapixels
NR	no-reference
PVSs	processed video sequences
QoE	Quality of Experience
QoS	Quality of Service
RF	Random forest models
RMSE	root mean square error
RR	reduced-reference
SVM	Support Vector Machine
SVR	Support Vector Regressor
bpp	bits-per-pixel