Technische Universtität München
Lehrstuhl für Medientechnik

# Towards User Experience-Driven Adaptive Uplink Video Transmission for Automotive Applications

Dipl.-Ing. Christian Lottermann

Vollständiger Abdruck der von der Fakultät Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

| | | |
|---|---|---|
| Vorsitzender: | | Univ.-Prof. Dr.-Ing. Klaus Diepold |
| Prüfer der Dissertation: | 1. | Univ.-Prof. Dr.-Ing. Eckehard Steinbach |
| | 2. | apl. Prof. Dr.-Ing. Walter Stechele |

Die Dissertation wurde am 03.06.2015 bei der Technischen Universität München eingereicht und durch die Fakultät Elektrotechnik und Informationtechnik am 02.11.2015 angenommen.

# Abstract

A large share of connected mobile devices, such as smartphones or connected vehicles, is nowadays equipped with camera systems. As a consequence, the captured videos are increasingly upstreamed to video portals or directly to other devices. This is mainly challenged by high user experience demands, the available computational resources of the mobile devices, and the limited uplink transmission capacity and connectivity. The focus of this thesis is to develop and evaluate means to enable user experience-driven uplink video streaming from mobile video sources with limited computational capacity and to apply these to resource-constraint automotive environments.

The first part of the thesis investigates the perceptual quality-aware encoding of videos. To this end, a video bit rate model is proposed to estimate the bit rate of encoded videos as a function of the quantization parameter, frame rate, group of pictures length, and group of pictures structure encoding settings. Temporal and spatial activity-based estimators of the video content-dependent model parameters for H.264/MPEG-4 AVC encoded videos are developed. A performance assessment shows that the proposed bit rate model is highly accurate. Together with an objective video quality metric, the video bit rate model is used to determine the encoding settings which maximize the perceptual quality for given bit rate constraints.

In the second part of the thesis, the solution from the first part is applied to videos captured with a front-facing camera of a vehicle. In vehicular deployments access to the raw video stream, which is required to determine the temporal and spatial activity parameters, might not be possible. As a remedy, camera context-aware estimators of the temporal and spatial activity parameters are developed, which use information about the status and the dynamics of the vehicle and the vehicles in the field-of-view of the front-facing camera. The developed estimators show a high estimation performance of the measured temporal and spatial activity values.

The last part of the thesis studies the upstreaming of video content from a mobile video source using adaptive HTTP streaming. Due to their limited computational capacities, mobile video sources, such as modern vehicles, are typically not able to simultaneously generate the same number of video levels as commonly employed in adaptive HTTP streaming content delivery network deployments. As a remedy, three context-aware video level selection algorithms, which employ different context information, are proposed in order to select a reduced set of video levels out of a pre-defined static video level set. Experimental results show that the number of video levels at the mobile video source can be reduced significantly while ensuring a high user experience in the streaming sessions.

# Kurzfassung

Ein Großteil der heutigen mobilen Endgeräte, wie etwa Smartphones oder vernetzte Fahrzeuge, sind mit Kameras ausgestattet. Dieser Trend führt dazu, dass vermehrt aufgezeichnete Videos zu Videoportalen oder direkt zu anderen Endgeräten übertragen werden. Die Herausforderungen hierbei liegen vor allem in den hohen Nutzeranforderungen, den zur Verfügung stehenden Berechnungskapazitäten mobiler Endgeräte und den Verbindungskapazitäten heutiger Mobilfunksysteme. Der Schwerpunkt dieser Dissertation liegt daher in der Entwicklung von Mechanismen, die eine Videoübertragung von einem mobilen Endgerät mit begrenzten Berechnungskapazitäten unter Berücksichtigung des Nutzererlebnisses ermöglichen und diese auf ressourcenbeschränkte automobile Szenarien anzuwenden.

Der erste Teil der Dissertation untersucht die Videoencodierung unter Berücksichtigung der Wahrnehmungsqualität. Zu diesem Zweck wird zuerst ein Videobitratenmodell vorgeschlagen, welches eine Abschätzung der Bitrate encodierter Videos als Funktion des Quantisierungsparameters, der Framerate, der Bildgruppenlänge und der Bildgruppenstruktur ermöglicht. Für die videoinhaltsabhängigen Modellparameter H.264/MPEG-4 AVC encodierter Videos werden Schätzer auf Basis räumlicher und zeitlicher Aktivitätsparameter entwickelt. Eine durchgeführte experimentelle Evaluation zeigt, dass das entwickelte Modell eine sehr präzise Abschätzungsgenauigkeit der Bitraten ermöglicht. Das Bitratenmodell wird darüberhinaus in Verbindung mit einer objektiven Videoqualitätsmetrik dazu verwendet, die Videoencodierparameter für vorgegebene Zielbitraten derart zu bestimmen, dass die Wahrnehmungsqualität maximiert wird.

Im zweiten Teil der Dissertation wird die entwickelte Lösung zur Bestimmung der Videoencodierparameter auf ein Fahrzeugszenario angewendet, bei dem der Videoinhalt einer Frontkamera übertragen werden soll. Der direkte Zugang zu den unkomprimierten Videoströmen, der zur Bestimmung der örtlichen und zeitlichen Aktivitätsparameter benötigt wird, ist in modernen Bordnetzarchitekturen oftmals nicht realisierbar. Daher werden Schätzer beider Aktivitätsparameter entwickelt, die Kontextinformationen der Kamera nutzen, wie etwa die Zustandsinformation des Fahrzeugs, sowie fahrdynamische Eigenschaften des Fahrzeugs und anderer Fahrzeuge im Sichtfeld der Frontkamera. Eine durchgeführte Evaluation beider entwickelter Schätzer bestätigt eine hohe Abschätzungsgenauigkeit.

Der letzte Teil der Dissertation untersucht die Videoübertragung von mobilen Endgeräten mittels adaptiver HTTP Übertragung in der Aufwärtsstrecke. Durch die begrenzte Berechnungskapazität heutiger mobiler Endgeräte, wie etwa Fahrzeugen, kann nicht die gleiche Anzahl von Videostufen erzeugt werden, die üblicherweise bei der adaptiven HTTP Videoübertragung von Content-Delivery-Netzwerken eingesetzt wird. Daher werden drei kontextabhängige Algorithmen zur Videostufenauswahl vorgeschlagen, die unterschiedliche Kontextinformationen nutzen, um eine Untermenge von Videostufen aus einer vorher festgelegten Menge von Videostufen auszuwählen. Eine durchgeführte experimentelle Evaluation aller drei Algorithmen zeigt, dass die Anzahl der Videostufen signifikant reduziert werden kann, ohne eine Verschlechterung des Nutzererlebnisses zu erzeugen.

# Acknowledgements

I would like to thank all the people who have supported me on my PhD journey within the last three and a half years. Without them, the progress and the work would not have been possible. At this point, I want to thank all of them, including those not mentioned here by name.

First and foremost, I would like to thank Prof. Dr.-Ing. Eckehard Steinbach for accepting me as an external PhD student and giving me the opportunity to conduct my PhD under his supervision. He opened my eyes in doing research in fascinating and contemporary research areas. I highly acknowledge his continuous support, guidance, and expertise which have been vital in the process of my PhD. I also want to thank Dr.-Ing. Wolfgang Hintermaier who supervised my work at BMW, for building bridges inside the company to a variety of people, for fruitful discussions, and for having always valuable advises for doing research in this setup.

Special thanks also go to the second examiner Prof. Dr.-Ing. Walter Stechele and to Prof. Dr.-Ing. Klaus Diepold for chairing the thesis committee.

Besides, I would like to thank my university colleagues, especially Damien Schroeder, for many fruitful discussions, for his valuable feedback, and for publishing many common ideas together. I also would like to thank my colleagues from different departments of the BMW Group Research and Technology (LT-3) and BMW Group (EI-5), who supported me during my PhD time, especially Dr. techn. Peter Fertl, Dr.-Ing. Felix Klanner, Dr.-Ing. David Gozalvez-Serrano, Dr. techn. Levent Ekiz, Dr.-Ing. Sebastian Zimmermann, and Martin Arend. Special thanks also goes to all my BMW PhD colleagues, particularly Zhe Ren for many fruitful discussions and many common writing days in the library, Christian Ruhhammer and Justus Jordan for their support with the vehicle prototypes. And, of course, to all students I supervised and who have contributed to this thesis due to their master's theses or internships and many productive discussions, especially Alexander Machado and Serhan Gül, who thankfully gave valuable feedback on the thesis.

Last but not least, I also want to thank my friends and family, especially my parents and my brother, for all their love and support throughout the PhD period and my whole life. They helped me through all ups and downs that a PhD period brings with so many things and made the completion of the thesis possible.

Munich, June 2015 *Christian Lottermann.*

# Contents

**Contents** **vii**

**1. Introduction** **1**
1.1. Contributions . . . 3
1.2. Organization . . . 5

**2. Background** **7**
2.1. Video coding and rate control . . . 7
2.1.1. Video coding . . . 7
2.1.2. Video rate control . . . 12
2.1.3. H.264/MPEG-4 AVC . . . 13
2.2. Video streaming technologies . . . 14
2.2.1. Adaptive video coding techniques . . . 14
2.2.2. Video streaming protocols . . . 15
2.3. Video quality . . . 19
2.3.1. Subjective video quality assessment . . . 20
2.3.2. Objective video quality assessment . . . 24
2.3.3. Perceptual video quality modeling . . . 27
2.4. Automotive systems . . . 36
2.4.1. Sensors for advanced driver assistance services . . . 36
2.4.2. Automotive communication technologies . . . 38

**3. Video bit rate model for perceptual quality-aware rate control** **43**
3.1. Introduction . . . 43
3.2. Related work . . . 45
3.3. Proposed video bit rate model . . . 46
3.3.1. Analytical rate factor modeling . . . 47
3.3.2. Content-based model parameter estimation . . . 53
3.3.3. Performance evaluation . . . 54
3.4. Application in perceptual quality-aware rate control . . . 59
3.4.1. Problem definition and solution . . . 60

3.4.2. Application and performance evaluation . . . . . 61
3.5. Chapter summary . . . . . 64

**4. Context-aware estimation of temporal and spatial activities 67**
4.1. Introduction . . . . . 67
4.2. Camera context features . . . . . 69
4.2.1. Experimental settings and context information . . . . . 69
4.2.2. Temporal activity related features . . . . . 70
4.2.3. Spatial activity related features . . . . . 71
4.3. Estimation of temporal and spatial activity values . . . . . 72
4.3.1. Temporal activity estimator development . . . . . 73
4.3.2. Spatial activity estimator development . . . . . 75
4.3.3. Performance evaluation . . . . . 76
4.4. Application in perceptual quality-aware video rate control . . . . . 78
4.4.1. Spatio-temporal video quality metric . . . . . 79
4.4.2. Video bit rate model . . . . . 80
4.4.3. Perceptual quality-aware video rate control . . . . . 81
4.5. Chapter summary . . . . . 83

**5. Dynamic video level encoding for uplink adaptive HTTP streaming 85**
5.1. Introduction . . . . . 85
5.2. Related work . . . . . 87
5.3. Dynamic video level selection . . . . . 89
5.3.1. Video level selection objective . . . . . 89
5.3.2. Network performance-aware dynamic video level selection . . . . . 90
5.3.3. Client request-aware dynamic video level selection . . . . . 93
5.4. Performance evaluation in automotive streaming scenarios . . . . . 95
5.4.1. Simulation model . . . . . 95
5.4.2. Simulation results . . . . . 98
5.5. Chapter summary . . . . . 102

**6. Conclusions and future directions 107**
6.1. Conclusions . . . . . 107
6.2. Future directions . . . . . 108

**A. SAMVIQ guidelines 111**

**B. Thesis website 113**

**List of Abbreviations 115**

**List of Symbols 119**

**List of Figures** **123**

**List of Tables** **127**

**Bibliography** **129**

## Chapter 1

# Introduction

With the development and deployment of high data rate and low delay cellular radio access networks, such as Universal Mobile Telecommunications System (UMTS) in the 2000s and the recent evolution towards 4G standards, such as Long Term Evolution (LTE) and LTE-Advanced, video streaming has evolved to one of the most popular mobile Internet services over the last decade. According to Cisco's traffic forecast [Cis13], video traffic is responsible for half of the overall mobile Internet traffic as of today, and this amount is expected to increase 14-fold until the end of 2018.

Modern consumer electronic devices, such as smartphones, tablet computers, or connected vehicles, are more and more equipped with enhanced signal processing capabilities and camera modules which allow for mobile capturing and processing of high quality video content. Besides the traditional offline storage, videos are increasingly upstreamed from mobile devices to video portals (e.g., YouTube [Webc]) or directly to other consumer electronic devices (e.g., Periscope [Webb]) over wireless networks on demand or in real time. Modern connected vehicles, for example, are equipped with numerous cameras and sensor systems primarily used for on-board advanced driver assistance service (ADAS) applications. Safety, security, and convenience in road situations can be further increased by upstreaming the captured videos of the ADAS cameras directly to other devices outside the vehicle or to a video portal for further on-demand sharing. The video content captured by ADAS cameras, for example, can be used in traffic monitoring systems used for real-time road traffic information and surveillance [BV+14].

Besides the limited computational capacities, video streaming to and from mobile devices is challenging due to the time-varying network throughput performance and frequent inter-radio access network (RAN) handovers, especially at vehicular velocities. In order to realize reliable and interruption-free transmissions of video streams at a high user satisfaction, enhanced encoding and rate adaptation mechanisms are required, which offer support for an adaptation of the bit rate of video streams according to the network performance along the

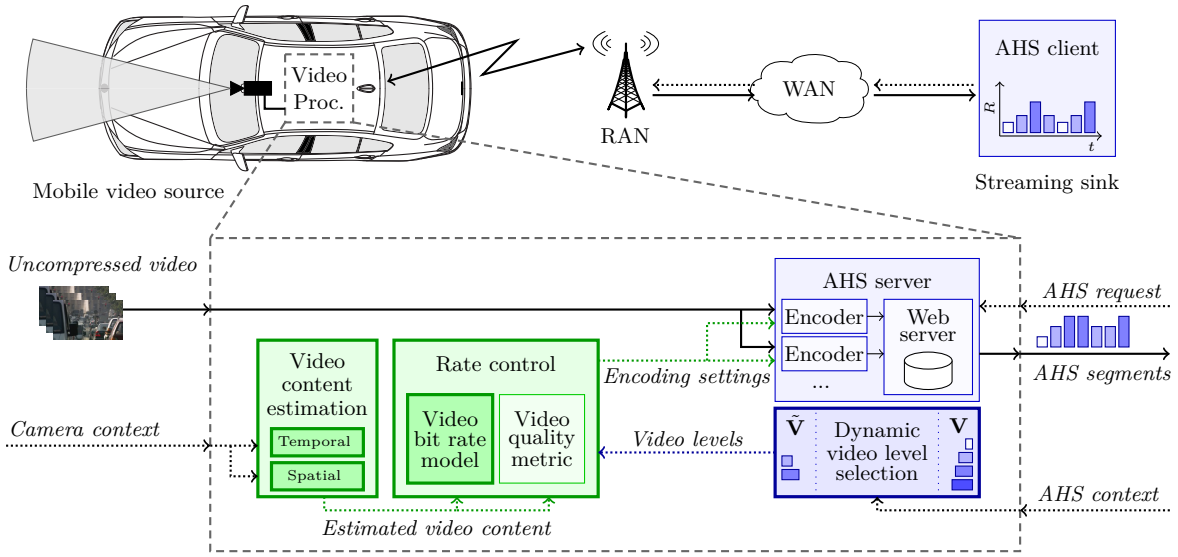



Figure 1.1.: Schematic overview of the considered AHS-based uplink streaming scenario where a source video stream captured with a mobile (vehicular) video source is upstreamed to a remote streaming sink. A perceptual quality-aware rate control entity computes the encoding settings for desired bit rates using video content information determined by camera context-aware estimators (marked in green). The bit rates required for the AHS-based streaming process are selected from a full static video level set using context information (marked in blue). Contributions of this thesis are framed in bold.

transmission path. Over the last years, adaptive HTTP streaming (AHS) gained high popularity, since it offers intra-session rate adaptation and relies on a Hypertext Transfer Protocol (HTTP)/Transmission Control Protocol (TCP) transmission, which is supported by almost all network components deployed in the Internet. AHS follows a pull-based streaming principle, where the streaming source divides the source video into segments of a defined duration, encodes the video segments at different desired bit rates (video levels), and stores the encoded video segments on a standard web server along with a manifest file, which contains references to the different video segments. The AHS client installed at the streaming sink requests the AHS segments at a bit rate which matches the transmission capacity using HTTP requests. So far, AHS has mainly been considered for live and on-demand downlink delivery of video content from content delivery networks (CDNs). This thesis goes one step further and investigates AHS-based streaming for the uplink transmission of videos from mobile video sources with limited computational resources. Figure 1.1 displays the uplink streaming scenario considered within this thesis where a source video stream captured with a mobile (vehicular) video source is upstreamed to a remote streaming sink. The thesis contributes to the perceptual quality-aware, AHS-based adaptive uplink transmission of delay-tolerant video streams in two directions.

First, the determination of encoding settings for desired target video bit rates is investigated, which can be employed as control information at the rate controller of the AHS video source

to realize the desired bit rates. Most of the previously proposed rate controllers for hybrid video coding consider the sole adaptation of spatial encoding settings by a modification of the quantization parameter to adapt the bit rate of the encoded video and employ peak signal-to-noise ratio (PSNR) as the distortion measure in the rate control process. However, it has been shown that by additionally taking modifications of other encoding settings in the rate control process into account, such as adaptations of the temporal resolution, the perceptual quality for desired bit rates might increase significantly [WMO09]. Besides that, it has been shown that PSNR does not correlate well with the human perception, since it does not take the characteristics of the human visual system (HVS) into consideration [Gir93; GHT08]. To overcome these limitations, this thesis considers perceptual quality-aware rate control which suggests the joint modification of spatial quality (i.e., image quality) and temporal resolution encoding settings. For this purpose, a content-dependent video bit rate model and a perceptual video quality metric need to be employed in the rate control process, which consider spatial quality and temporal resolution encoding settings. The determination of the video content-dependent parameters of the bit rate model and the video quality metric requires access to uncompressed source video. This, however, might not be possible, especially in automotive ADAS camera deployments, where typically only restricted access to internal functions and data streams is possible. As a remedy, this thesis investigates an estimation of the video content-dependent parameters based on camera context information.

Second, the uplink streaming of video content from mobile video sources with limited computational capacity using AHS is studied, which is significantly different from the downlink video streaming from CDNs. CDN systems typically generate a comprehensive number of video levels in order to be able to serve all streaming clients at a high user experience in the streaming sessions. In contrast, mobile video sources, such as modern vehicles equipped with ADAS camera systems, offer limited computational capacities and as a consequence allow the parallel generation of only a limited number of video levels. In order to reduce the number of video levels employed in the AHS source process, this thesis investigates algorithms to dynamically select a subset of video levels out of a full static video level set based on different sources of context information.

## 1.1. Contributions

Figure 1.1 displays the contributions of this thesis along the considered uplink streaming scenario. The main contributions are summarized as follows:

1. **Video bit rate model as a function of quantization parameter, frame rate, and group of pictures (GoP) encoding settings:** This thesis proposes a bit rate model for video encoding which considers spatial quality impairments resulting from modifications of the quantization parameter and temporal quality degradations due to

reductions of the frame rate. Besides that, the model captures the influence of the GoP length and GoP structure. Separate factors are defined for each of the four encoding settings, which need to be determined for each source video individually. The model is further applied to H.264/MPEG-4 Advanced Video Codec (AVC) video encoding and estimators of the video content-dependent parameters based on standard video activity measures (temporal and spatial activity) are developed.

2. **Perceptual quality-aware video bit rate control:** A perceptual quality-aware optimization problem to determine quantization parameter and temporal resolution encoding settings for given bit rate constraints of video segments is defined. Based on the aforementioned bit rate model and a video quality metric, which captures the perceptual quality of encoded videos, a solution to the problem for H.264/MPEG-4 AVC video encoding is developed. Since both the bit rate model and the video quality metric use spatial and temporal activity measures to estimate the content-dependent parameters, the determined solution can easily be deployed in automated video processing systems. As an application, the developed solution is applied to an AHS video source, where the optimal encoding settings need to be determined for different desired target bit rates.

3. **Camera context-based estimation of spatial and temporal activity measures:** The calculation of the spatial and temporal activity measures required for the proposed solution of the perceptual quality-aware rate control problem is problematic in automotive deployments since access to the uncompressed source camera stream and to the internal functions of video encoders is typically not possible. This thesis proposes low-complexity estimators of the spatial and temporal activity measures for videos captured with an ADAS front-facing camera of a vehicle based on context information of the vehicle. The developed estimators employ information about the scenario where the video is captured, the dynamics of the vehicle, and the dynamics of other vehicles in the field-of-view of the ADAS front-facing camera.

4. **Dynamic video level encoding for AHS-based uplink video transmission:** An AHS-based streaming system is considered for the uplink transmission of live video content from mobile video sources with limited computational and encoding capacities which only support a limited number of parallel encoding processes. In order to significantly reduce the number of video levels which need to be encoded and to enable the application of the considered AHS system at the mobile video source, three context-aware dynamic video level selection algorithms are proposed. The goal of the algorithms is to select a reduced set of video levels out of a pre-defined static video level set based on context information. To this end, two algorithms employ different statistical information of the measured network performance of the streaming session and one algorithm uses the history of previous segment requests.

## 1.2. Organization

The remainder of this thesis is organized as follows. Chapter 2 provides background material for this thesis. It first reviews the main concepts and recent advances in the area of video coding, provides an overview of adaptive streaming technologies, presents subjective quality assessment methodologies, and compares video quality metrics to estimate the perceptual quality of encoded videos. Besides that, ADAS sensor and communication technologies of modern automotive systems are introduced. Chapter 3 proposes a video bit rate model which captures the influence of spatial, temporal, and GoP encoding settings on the bit rate. The model is applied to H.264/MPEG-4 AVC video coding and corresponding estimators for the video content-dependent model parameters using temporal and spatial activity measures are developed. A perceptual quality-aware rate control problem to determine optimal quantization parameter and temporal resolution encoding settings for desired rate constraints is defined and a solution based on the proposed bit rate model and a video quality metric is developed. Chapter 4 is devoted to the development of camera context-based estimators for spatial and temporal activity measures of videos recorded with an ADAS front-facing camera based on camera context information. The developed estimators are applied to the solution of the perceptual quality-aware rate control problem. In Chapter 5, AHS-based streaming is considered for the uplink transmission of videos from mobile devices. Context-aware dynamic video level selection algorithms are proposed to reduce the number of video levels which need to be encoded on the mobile device. The proposed algorithms are applied to an automotive scenario, where the content of an ADAS front-facing camera is upstreamed to a video portal deployed in the Internet. Finally, Chapter 6 concludes this thesis with a summary of the results and points out some limitations and potential future research directions.

A complementary website of this thesis is available and is introduced in Appendix B.

Parts of this thesis have been published in [LSS; LG+15; LS14; LM+14a; LM+14b].

Chapter 2

# Background

This chapter presents the basic concepts of video coding, gives an overview about adaptive video transmission systems, reviews subjective quality assessment methodologies, and compares objective video quality metrics to estimate the perceptual quality of encoded videos. Finally, it introduces sensor and communication technologies of modern automotive systems.

## 2.1. Video coding and rate control

Video compression is required to reduce the bit rate of a raw digital video in order to enable the transmission of video streams over communication channels with limited transmission capacity or the storage of the compressed video content on a medium with limited storage capacity.

The following section first introduces the concept of video coding, gives a brief overview about the basic building blocks of a modern hybrid video codec, describes the concept of video rate control, and gives a brief overview about the features of H.264/MPEG-4 AVC.

### 2.1.1. Video coding

The compression and decompression of a digital representation of a video is commonly referred to as video coding [Ric03]. Video compression reduces the quantity of bits to digitally represent a video and is measured by the compression ratio which is defined as the number of bits of the uncompressed representation divided by the number of bits after the compression. Lossy compression offers potentially high compression ratios, however, introduces irreversible distortion to the original data, which cannot be reconstructed correctly by the decoder.

Modern video coders, such as block-based hybrid coding systems (displayed in Figure 2.1), typically apply two different coding methodologies: (i) *intra-frame coding*, which comprises

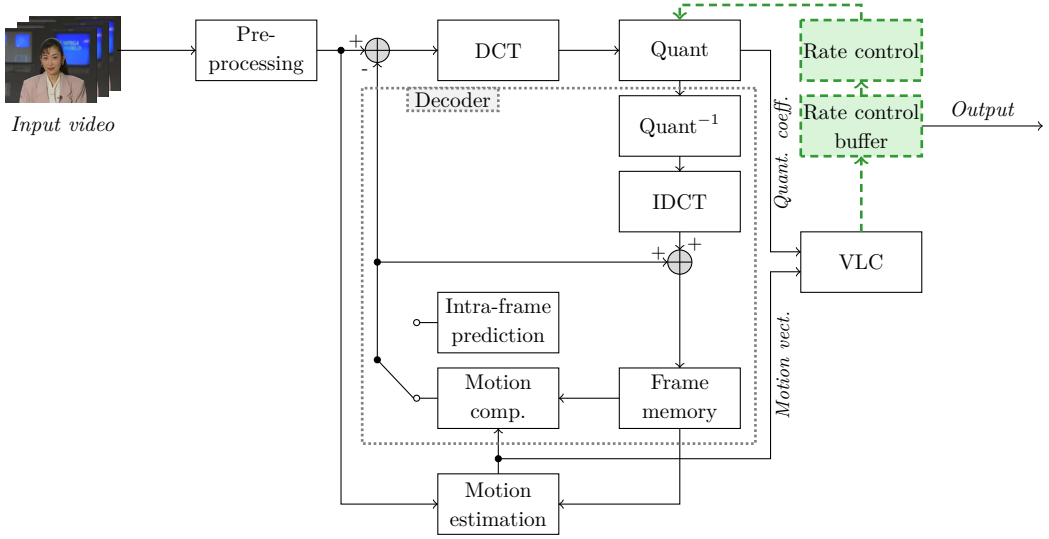



Figure 2.1.: Encoding and decoding (marked with ⋯) process of a block-based hybrid coding system. Rate control blocks are marked in green. Adapted from [WOZ02].

compression techniques for information contained in the single frames, and (ii) *inter-frame coding*, which exploits the temporal redundancies of successive frames to further compress the video content. In the following, all essential mechanisms employed in block-based hybrid coding systems are introduced.

#### 2.1.1.1. Intra-frame coding

In the following, the processing steps used for intra-frame coding in hybrid block-based coding systems are introduced.

**Color space transformation and block partitioning:** Before the actual coding of the raw video content, the video frames are pre-processed (marked as *Pre-processing* in Figure 2.1). First, the video frames need to be transformed to the *YCbCr* color space which is typically applied in the digital representation of video frames. The luminance component ($Y$) and the two chrominance components ($Cb$, $Cr$) can be processed independently of each other. As observed in subjective tests, the HVS is more sensitive to variations in brightness rather than variations in color. To this end, the video representation can be optimized by allocating more bits to the Y component than to Cb and Cr. For example, in the 4:2:0 chroma subsampling scheme, which is applied throughout this work, half of the number of samples for the representation of the content is required compared to a 4:4:4 scheme, where no chroma subsampling is applied. The pixel values of the Y, Cb, and Cr components are partitioned into rectangular blocks of a defined size, commonly referred to as macroblocks [WOZ02]. Macroblocks are further used to exploit the statistical correlation between the blocks spatially (for intra-coding) and temporally (for inter-coding).

(a) Uncompressed ($q = 0$). (b) Medium quantization parameter ($q = 32$). (c) Large quantization parameter ($q = 48$).

Figure 2.2.: Example frame of the *Football* video [Seq] encoded with H.264/MPEG-4 AVC using x264 [Vid] at different quantization parameter settings.

**Transform coding:** In video coding, transform coding is a lossless process which describes the frequency transform of the pixel values of a video frame into transform coefficients. After the transform, the coefficients exhibit reduced statistical dependencies which can be exploited for the further compression. Also, more focus can be set on the coefficients which have a high impact on the human perception. The 0-frequency component is typically known as the *DC* component, whereas the *AC* components denote the higher frequency components. In video coding, typically *discrete cosine transform (DCT)* is used as the transform [WOZ02]. For example, most modern Moving Picture Experts Group (MPEG) video codecs employ a DCT-based transform (marked as *DCT* and *IDCT* as the inverse operation in Figure 2.1).

**Quantization:** One major part of the actual compression in intra-frame coding is performed by quantization (marked as *Quant* with the inverse operation $Quant^{-1}$ in Figure 2.1). To reduce the information of the transform coded representation of the block, the transform coefficients are quantized. Coefficients below a defined quantizer threshold are set to zero [Sun00], whereas coefficients above or equal to the defined quantizer threshold are divided by the quantization step size and set to the integer value with the smallest absolute difference [RR97]. Video encoders employ a quantization matrix which contains quantization step sizes for the different transform coefficients [WOZ02]. The quantization matrix is further multiplied with a quantization factor, which is used to control the quantization step size, and hence, the encoding quality and the compression ratio for the corresponding frames. The quality of the frames decreases as the quantization factor increases. In modern codecs, such as H.264/MPEG-4 AVC, the quantization factor is referred to as the *quantization parameter*, which is defined in values between 0 (uncompressed) and 51 (worst quality) [WS+03]. Quantization is a lossy step in the encoding process and as a consequence, the original transform coefficients can not be reconstructed correctly by the decoder. This might lead to visible distortion artifacts in the video frames. In Figure 2.2 these artifacts are demonstrated for three different quantization parameters using H.264/MPEG-4 AVC video coding. For small values of the quantization parameter (Figure 2.2b) almost no distortion is visible in the video

frames, whereas for larger quantization parameter values, block artifacts can clearly be perceived (Figure 2.2c). Distortion in video frames is commonly measured as the mean square error (MSE) or PSNR[1] between the compressed and the uncompressed (raw) video.

**Variable length coding:** Variable length coding (VLC) is a form of entropy coding and is applied to further decrease the number of allocated bits after the quantization by reducing the statistical redundancies of the quantized transform coefficients. Before applying VLC, the coefficients are arranged in a 1-D array by scanning the coefficients in a specific manner starting with the DC component and followed by the AC components using a zig-zag scan pattern [Gha99]. Since a significant number of coefficients is zero after the quantization, it is not reasonable to specify each of the coefficient values individually. To this end, run-level coding is applied, which represents the coefficients in symbols consisting of the values from non-zero coefficients and the number of zero coefficients [WOZ02]. VLC uses a mapping of symbols to codewords which considers short codewords for symbols with high probabilities [SFR07]. VLC is commonly implemented by arithmetic coding and Huffman coding [HV91; LH87]. Arithmetic coding offers high compression ratios, however, relies on computationally complex operations as opposed to Huffman coding, which is based on less complex operations at the cost of lower compression ratios. Since modern systems offer high computational capacities, typically arithmetic coders are employed. H.264/MPEG-4 AVC and High Efficiency Video Coding (HEVC), for example, use context-adaptive binary arithmetic coding (CABAC) for VLC [WS+03; SO+12].

#### 2.1.1.2. Inter-frame coding

The previously discussed coding steps exploit the spatial redundancies of the single video frames and are employed for intra-frame coding. The content differences of successive frames in videos is typically low. To this end, modern video encoders additionally use inter-frame coding to exploit the temporal redundancies among successive frames in video sequences and to predict a video frame from coded past or future frames. Modern video encoders typically employ motion compensated prediction for inter-frame coding [WOZ02].

**Motion compensated prediction:** Two main concepts are employed for the motion compensated prediction. First, using *motion estimation* an area of pixels in the reference frame for each macroblock of the current frame with the smallest prediction error is identified. To this end, a block-matching algorithm is applied to determine the motion vector that describes the displacement of each macroblock in the reference frames, which have to be decoded and stored in the frame memory [HC+06]. Modern video encoders offer sub-pixel shift accuracies, such as half-pixel or quarter-pixel, to achieve high accuracies for the motion estimation and to realize

[1] A further definition and discussion of PSNR is given in Section 2.3.2.1.

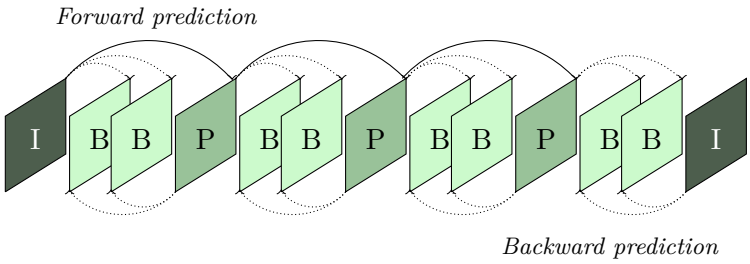



Figure 2.3.: Representative MPEG GoP with I-, P-, and B-frames ($n = 12$, $m = 2$).

an exact encoding of the motion of the videos. Second, the process of *motion compensation* determines the difference between the block of the predicted frame and the reference frame, also commonly referred to as the residual. The residual is transformed, quantized and further compressed using VLC. In the overall encoding process, the motion compensated prediction is the computationally most demanding operation, which consumes roughly 80-90% of the overall computation time of the encoding process in H.264/MPEG-4 AVC video encoding [HH+06].

#### 2.1.1.3. Group of pictures

In modern video coding standards, three different frame types are produced [WOZ02]: $I$ (intra-frame), $P$ (forward predicted inter-frame), and $B$ (bi-directionally predicted inter-frame). I-frames offer random access points for the video playback, however, typically use more bits after the encoding compared to P- or B-frames. These frames are usually grouped into sequences starting with an I-frame and ending before the successive I-frame, also commonly referred to as *GoPs* [WOZ02]. The GoPs are typically continuously repeated in the overall video sequence. A typical MPEG GoP [ISO93], which incorporates all frame types and shows the dependency of the different frame types on each other, is displayed in Figure 2.3. The GoP starts with an I-frame, followed by the predicted P- and B-frames. The structure of GoPs is typically characterized by the GoP length $n$ and the number of consecutive B-frames $m$ [LS14].

P-frames reference preceding frames in display and decoding order. B-frames reference both frames in preceding (referred to as forward prediction) and frames in successive display order (referred to as backward prediction) [ISO93; ISO00]. In modern standards, such as H.264/MPEG-4 AVC [ISO03], multiple previous frames can be used as a reference of P- and B-frames (up to 16). Furthermore, B-frames can also be employed as references for other P- and B-frames [ISO03].

### 2.1.2. Video rate control

As previously discussed, the quantizer settings have a significant impact on the distortion of the encoded video frames. In video coding, two different coding concepts with respect to the distortion and bit rate properties are defined: *variable bit rate (VBR)* and *constant bit rate (CBR)* video coding. In VBR, the quality is kept constant over the encoded video sequence using constant quantizer settings. As a consequence, the video bit rate after the encoding process fluctuates significantly in the order of a decimal magnitude depending on the spatial and temporal characteristics of the video sequence. This, however, might not be reasonable for some video applications where a constant bit rate of the encoded video is required, such as AHS systems, which require the source video encoded at desired target bit rates[2]. To this end, modern video encoders employ rate controllers to dynamically adapt the quantization settings for encoded video sequences, and hence to achieve a constant output bit rate, which is referred to as CBR coding. Unlike in VBR coding, where the quality is kept constant, the quality of the encoded frames in CBR coding might vary significantly, especially in spatially complex scenes [CN07].

The main task of rate controllers is to solve the inherent trade-off between distortion and bit rate of the encoded video [OR98]. As investigated in *R-D* theory [Sha48], distortion $D$ is a decreasing function of the bit rate $R$. This is described by the *R-D* function that defines the theoretical lower bound for the bit rate at a given distortion [CN07]. The general rate control problem to achieve the minimum distortion without exceeding a given bit rate constraint $R_c$ can be formulated as

$$\text{minimize } D \tag{2.1}$$

$$\text{subject to } \quad R \leq R_c.$$

Figure 2.1 shows a video encoder with a rate control unit, which consists of two major elements: (i) a buffer which is set up at the output of the VLC to store the encoded frames, and (ii) the rate control entity which is responsible for adapting the quantizer settings in order to achieve the target bit rate constraint. Rate controllers used in modern encoders, such as H.264/MPEG-4 AVC [ISO03], allow to perform the control on a macroblock-, slice-, and frame-level [CN07]. For all three, the *R-D* relationships are typically realized by *R-D* models, which are developed based on statistical measures of the video sequences and *R-D* theory [HC97]. Throughout this thesis, frame-level rate control is applied. A detailed overview about frame-level *R-D* models which relate the bit rate of encoded videos versus the quantizer settings is given in Section 3.2.

More recently proposed rate controllers employ a multi-dimensional adaptation which additionally consider spatial [BEK03] and temporal resolution [LK05; WMO09] changes. This, however, requires an extension and reformulation of the rate control problem of Eq. (2.1),

[2] An introduction to AHS is given in Section 2.2.2.

which is discussed for quantization parameter and frame rate encoding settings in Section 3.4.

### 2.1.3. H.264/MPEG-4 AVC

Since H.264/MPEG-4 AVC[3] is used as the default video codec throughout this thesis, it is briefly introduced in the following. H.264/AVC was developed by the Joint Video Team (JVT), which is a consortium of ISO/IEC MPEG and International Telecommunication Union (ITU)-T Video Coding Expert Group (VCEG) [ISO03]. It is designed to offer at least twice the compression ratio as compared to H.263 [IT96], MPEG-2 [ISO00], and MPEG-4 Part 2 [ISO04]. Furthermore, it is designed to enable the integration of the encoded video data into different protocols and network architectures, such as high-quality but low bit rate streaming applications or the storage and broadcasting of high-definition video content [WS+03].

The H.264/AVC encoder structure is divided into two parts, the *video coding layer (VCL)* and the *network abstraction layer (NAL)* [Wen03]. The VCL represents the coded video content. Similar as the predecessor standards, H.264/AVC uses block-based motion-compensated coding, which offers improved rate-distortion characteristics. Compared to H.263, the bit rate demands could be halved at roughly the same distortion [OS+12], however, at the cost of higher computational demands of the overall coding process. The NAL, on the other hand, defines a generic format for the application in packet oriented transport systems (e.g., Internet Protocol (IP) packets) [WS+03]. During the encoding process, syntax elements are mapped to NAL units, which consist of the syntax element and the NAL header [WS+03]. A type and importance tag is appended to each syntax element. The importance tag contains information about the influence for the decoding in case the syntax element is corrupted or lost. This information can be exploited by the transmission system to prioritize some syntax elements.

H.264/AVC supports different profiles which incorporate different tools and features. Throughout this thesis, the *Main* profile is used, which supports I-, P-, and B-slices, quarter-pixel motion compensation, different block sizes, intra-prediction, in-loop de-blocking filters, multiple reference frames, CABAC, weighted prediction, and interlaced coding. Further information about H.264/AVC can be found in [WS+03; ISO03] and a detailed overview of the different profiles in [STL04].

HEVC [ISO13] is the successor standard developed by the JVT. Compared to H.264/AVC, major improvements have been achieved on the VCL. At the same level of distortion, the compression ratio could be doubled and very high spatial resolutions of up to 8K are supported [SO+12].

[3] In the remainder of this thesis, H.264/MPEG-4 AVC is referred to as H.264/AVC.

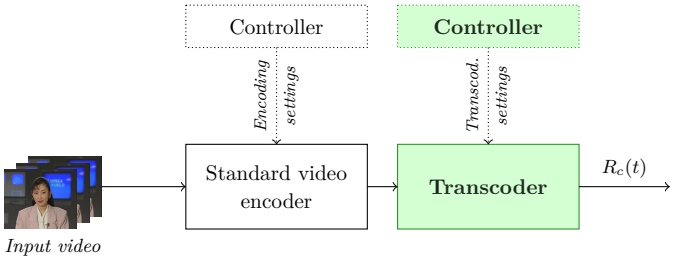



Figure 2.4.: Transcoding based adaptation of a video stream. Adapted from [DCMP11].

## 2.2. Video streaming technologies

Video streaming applications, such as on-demand or live streaming, gained substantial popularity over the last decade. In order to ensure an interruption-free transmission of video streams over wide area networks (WANs) and RANs at a high user satisfaction, adaptive coding and streaming technologies are employed in modern streaming systems.

The following section first introduces adaptive video coding techniques which enable the adaptation of the video bit rate according to the available network transmission capacity. Second, an overview of the most commonly used transport layer protocols designed for video streaming and their corresponding application layer implementations is given.

### 2.2.1. Adaptive video coding techniques

The techniques to adapt the source video stream to transmission capacity constraints can be classified into three main types [DCMP11]: *transcoding*, *scalable video coding (SVC)*, and *multiple bit rate coding (MBR)*.

**Transcoding:** Transcoding describes the digital transformation of an encoded representation of a video to a different one. To adapt the video bit rate according to the available transmission capacity, the source video is first encoded by a standard video encoder (cf., Figure 2.4). A separate transcoder is used to further encode the video at a target bit rate according to the available transmission capacity. The transcoding operation can be performed both in the temporal and spatial domain, e.g., by an adaptation of the frame rate, the spatial resolution, the picture quality, or combinations of those [MFW13]. Transcoding algorithms are able to realize a fine-granular resolution of bit rates of the encoded video. The transcoding process, however, introduces additional complexity and additional processing delays since the video needs to be decoded and re-encoded at a desired bit rate.

**Scalable video codecs:** SVC considers the encoding of a raw source video into one or more separate bitstreams [SMW07]. To this end, in SVC a layered coding structure is considered,

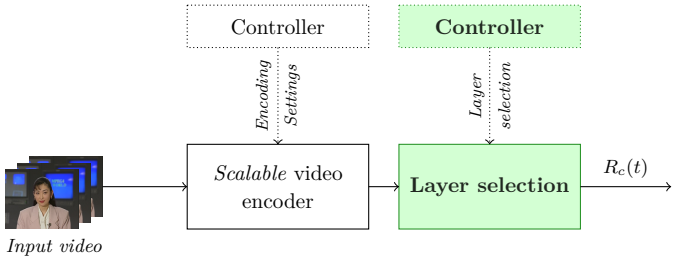



Figure 2.5.: Adaptation of a video stream using scalable video coding. Adapted from [DCMP11].

which incorporates a base layer and enhancement layers [SMW07]. The base layer, which is encoded in a way to be decoded independently, offers a minimal perceptual quality. Additional enhancement layers can be added in order to enhance the perceptual quality of the encoded video. They offer temporal scalability (frame rate), spatial scalability (spatial resolution), SNR scalability (picture quality), and combinations of the three [WOZ02]. To achieve a desired bit rate of the video bitstream, an adaptation algorithm selects video layers according to the transmission capacity (cf., Figure 2.5). In typical deployments, three to five layers are produced, since more layers typically lead to substantial inefficiencies in the rate-distortion performance. An SVC-based video transmission approach requires an advanced deployment strategy since an adaptation logic is required at the video source and intermediate proxies. Another limitation of SVC is that the usage of scalable codecs has not been adopted widely by industry as of today. Popular on-demand streaming Internet streaming platforms, such as *YouTube* [Webc] or *Netflix* [Weba], use single-layered representations of the encoded videos for AHS.

**Multiple bit rate coding:** In MBR coding (cf., Figure 2.6), a video coding entity encodes the raw source video at $N_L$ desired bit rates (video levels). The simultaneously encoding of the $N_L$ different video levels can, for example, be realized by $N_L$ separate video encoders or a multi-rate video encoding entity [FS+11; SRS15]. A separate adaptation algorithm selects the video levels dynamically according to the transmission capacity between the video source and the sink. In contrast to the transcoding-based approach, no additional transcoding step of the encoded video is required after the encoding to adapt the video to the network performance. Besides that, MBR is codec-agnostic since it does not rely on any advanced codec features [DCMP11]. MBR is applied in AHS, which has been realized in several commercial and open source implementations and is further discussed in Section 2.2.2.2.

### 2.2.2. Video streaming protocols

The transmission of video streams between a streaming server, which acts as a video source, and the streaming clients can be performed in different ways, depending on the type of video

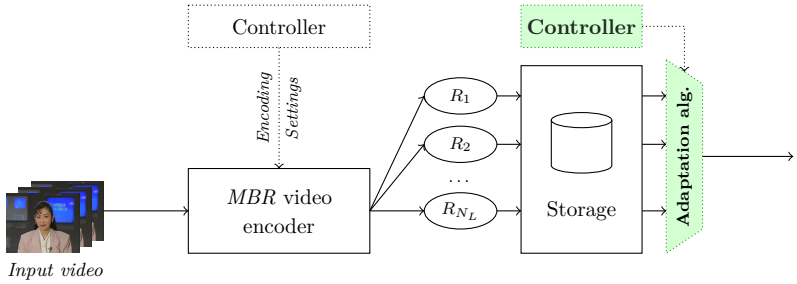



Figure 2.6.: Adaptation of the video stream using multiple bit rate coding. Adapted from [DCMP11].

content (live or on-demand) and the network conditions. To ensure a reliable delivery of the video content over lossy networks, streaming protocols are required which support retransmissions of lost packets. For real-time requirements, low latency protocols are required which might admit occasional packet losses. The major video streaming protocols can be grouped into two major classes [BAB11]: *push-based* and *pull-based* streaming protocols.

#### 2.2.2.1. Push-based streaming protocols

In push-based streaming, a media streaming session is established between the streaming source and the client, which is used for the media transmission until the client terminates the session. The server employs a session state to control the streaming session, which is updated by the client using session-state updates [BAB11]. These, for example, can be used at the server to adapt the bit rate of the encoded video stream according to the transmission capacity using a transcoder for the bitstream adaptation.

Real-Time Transport Protocol (RTP) [SC+03] is the most commonly used push-based streaming application protocol which provides end-to-end transport mechanisms developed for the transmission of multimedia content. RTP usually employs User Datagram Protocol (UDP) [Pos80] as a transport layer protocol, which does not offer internal rate control processes [BAB11]. For session control, such as the start-up or termination of a streaming session, Real-Time Streaming Protocol (RTSP) [SRL98] is applied. In typical RTP/UDP-based streaming sessions, the media bit rate is adapted according to the transmission capacity between the server and the client. For this purpose, the client constantly tracks the network statistics, such as throughput, round-trip-time, and jitter. The measurement reports are provided to the server using Real-Time Control Protocol (RTCP) [SC+03], which are used at the server to adapt the bit rate of the encoded video stream accordingly.

Despite its efficient bandwidth usage and its built-in adaptation mechanisms, RTP/UDP-based streaming has several disadvantages. The RTP-based media transmission might suffer

significantly due to the usage of UDP as a transport protocol. The inherent unreliability and likeliness of filterings of UDP packets at firewalls might lead to significant packet losses [Sto11] and as a consequence to degradations of the perceptual quality at the client. Therefore, flow congestion control and error control mechanisms are up to the application layer, since UDP does not provide these mechanisms [Sto11]. Besides that, for the transmission of the video contents a specialized RTSP server is required, which introduces additional complexity at the streaming server [BAB11].

#### 2.2.2.2. Pull-based streaming protocols

Unlike in push-based streaming, in pull-based streaming the client is responsible to command and request the video stream from the server. The delivered media bit rate from the server depends on the client requests. Most pull-based streaming protocols employ HTTP [FG+99] as an application protocol and TCP [Pos81] as the underlying transport protocol. In modern deployments, typically two different implementations of pull based streaming are used: non-adaptive *progressive download over HTTP* and *AHS*.

**Progressive download over HTTP:** In progressive download, the client requests a download of a pre-encoded media file stored on a conventional web server using a HTTP *GET* request. In response to the HTTP request, the download is started and performed at the maximum possible download rate [ABD11]. The playback of the video is performed in parallel to the download and started once the client buffer is filled above a certain threshold. If the download rate is equal to or larger than the playback rate, the client buffer is always filled above a critical level which ensures an interruption-free media playback. If, however, the download rate is lower than the playback bit rate of the video, buffer underflow events might occur which lead to stalling during the playback.

Progressive download over HTTP typically depends on TCP as the transport protocol. Due to TCP's retransmission mechanisms for lost packets and its successful traversal at firewalls, a reliable transmission of the media stream can be achieved. Besides that, TCP offers congestion control algorithms in order to realize an overall stable network [DCMP11]. This, however, can lead to significant TCP rate fluctuations which in turn causes stalling in the media playback due to an empty playout buffer. A sufficiently large buffer size and an initial playout delay are typically required to guarantee that the client playout buffer is filled above the critical threshold and consequently to ensure an uninterrupted playback.

Although progressive download is the most commonly used streaming system for on-demand media transmission as of today, it exhibits some major limitations. Progressive download does not offer live streaming capabilities and adaptation mechanisms [Sto11]. During a streaming session, the download rate cannot be adapted according to the available transmission capacity without re-initiating a new download session. This becomes problematic in the streaming

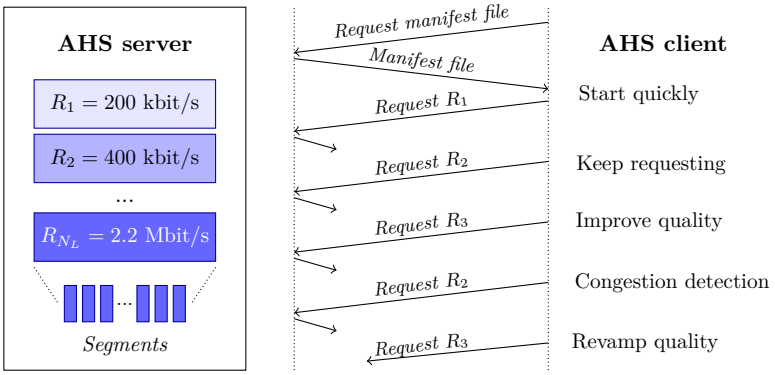



Figure 2.7.: Representative AHS rate adaptation scenario with adaptations of the client as a response to network performance changes. Adapted from [BAB11].

sessions of clients with significant network performance changes during the streaming sessions [ABD11]. Furthermore, transmitted media content in the buffer might be wasted in case of an abortive program stop [Sto11].

**Adaptive HTTP streaming:** AHS is a pull-based streaming technology which relies on HTTP/TCP transport mechanisms, analogous to progressive download over HTTP. AHS is designed to solve the major weaknesses of progressive download over HTTP by additionally employing rate adaptations and support for live streaming. In AHS, the source video stream is split into segments of a defined duration (typically 2-10 s [EK+14]). A MBR encoder is used to encode the video segments at multiple bit rates, which represent different video quality levels. Information about the available video levels, i.e., their bit rate, resolution, timing information, etc., are merged into a manifest file, which needs to be transmitted to the client prior to the streaming session. During a streaming session, the client continuously monitors its playout buffer fullness and the throughput to the streaming server. In return the client requests video levels at bit rates that best match the current network performance. The media stream can be fully reconstructed at the client, if all downloaded segments are played back consecutively [BAB11]. Figure 2.7 displays a typical AHS adaptation scenario, where a client reacts dynamically to changing network conditions.

Similar to progressive download over HTTP, AHS inherits the deployment and transmission properties of the underlying HTTP/TCP transport. Therefore, the encapsulated media packets do not suffer from firewall filterings on the transport layer. Furthermore, standard web servers can be used for the media streaming, which makes special streaming servers, such as RTSP servers required in RTP/UDP-based streaming, superfluous. Unlike progressive download over HTTP, AHS offers support for live-streaming, which makes it suitable for the transmission of live video content, such as live camera streams [Sto11]. In case of live streaming, the manifest file is updated on-the-fly once new video segments become available [Sod11].

AHS has initially been developed for the downlink delivery of videos from CDN networks. In this context, several proprietary AHS systems have been developed and deployed over the last years, such as Apple's *HTTP Live Streaming (HLS)* [PM14], Microsoft's *Smooth Streaming* [Zam09], and Adobe's *HTTP Dynamic Streaming* [Ado10]. Besides that, 3rd Generation Partnership Project (3GPP) and MPEG have developed Dynamic Adaptive Streaming over HTTP (DASH) as a common AHS standard, which is standardized as MPEG-DASH [ISO14] and 3GP-DASH [3GP14].

## 2.3. Video quality

Lossy compression is required to adapt the bit rate of encoded videos, depending on the application, to the storage or transmission channel capacities. This can be realized by an adaptation of the spatial quality as well as temporal and spatial resolutions, which all introduce different kinds of distortions to the encoded video. The artifacts, such as blurring, blocking, and jerkiness, appear as visually annoying distortions to the viewer [YW98]. In order to attain the desired bit rate constraints and to select the encoding settings that maximize the perceptual quality, it is necessary to evaluate the impact of the different impairments on the perceived quality of the viewer. To this end, two assessment methodologies are primarily used: *subjective* and *objective* quality assessments. In subjective quality assessments, human subjects evaluate the perceptual quality of the displayed videos. The assessments are conducted to investigate how the video quality is perceived for various impairments. While subjective video quality assessments are able to offer reliable information about the actually perceived quality, they are time-consuming and expensive since a large number of test subjects is required. Objective video quality assessments, on the other hand, employ video features which can be computed directly from the video frames and mathematical models of the HVS to determine the video quality. Since no direct human interaction is required, objective quality metrics are inexpensive and therefore suitable for video processing systems to automatically measure or estimate the video quality. However, objective video quality metrics are not able to capture various factors which affect the perceptual quality, such as the user experience and expectations, display settings, or specific tasks and thus might not be able to offer accurate estimations of the subjective video quality [Ric03].

In the remainder of this section, subjective video quality assessment methodologies are reviewed in Section 2.3.1. Objective video quality assessment methodologies are introduced in Section 2.3.2. Finally, major state-of-the-art objective video quality metrics for spatial quality and temporal resolution impairments are discussed and compared in Section 2.3.3.

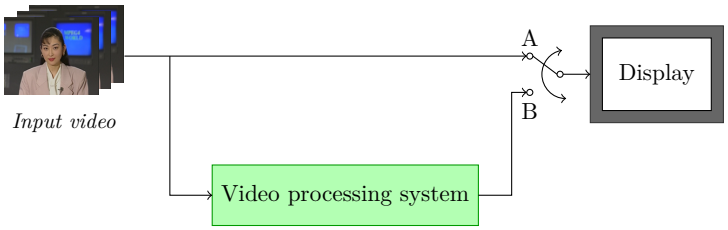



Figure 2.8.: Double-stimulus testing system. Adapted from [Ric03].

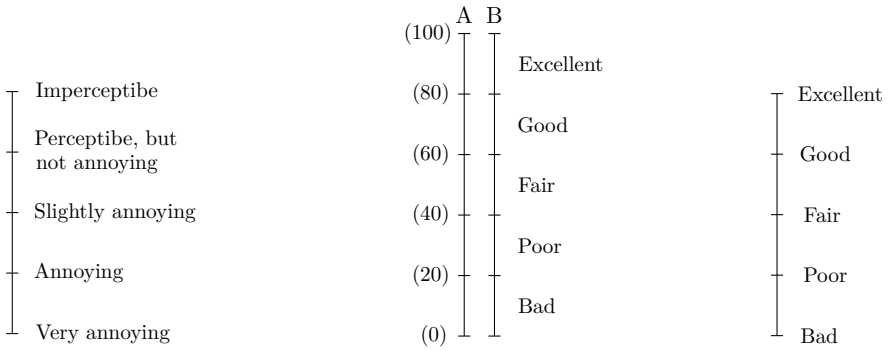
### 2.3.1. Subjective video quality assessment

The quality feeling of a perceived video is highly subjective as it is influenced by different factors of the human perception, such as the eye, the brain and their interactions [Ric03]. To capture the opinions of the subjective quality observed by the humans, subjective video quality assessment tests are applied, which present a set of video sequences to the human subjects and record the corresponding ratings. Two different kinds of subjective tests are commonly used [Zha14]. In *pairwise tests*, two test videos are displayed side-by-side in a test case and the subject has to evaluate which video has the higher perceptual quality as opposed to *mean opinion score (MOS)-based tests*, where the test videos are presented solely and the subject needs to assess the quality of each test video on a defined quality scale. The MOS value has originally been designed for the subjective assessment of the voice quality in telephone applications [IR98], however, is commonly applied to capture the subjective perception of video sequences. It is calculated as the mean of the individual quality ratings of the test subjects.

#### 2.3.1.1. Assessment methodologies

To design and conduct subjective tests and to ensure reproducibility of the subjective votes realized in equal test conditions, ITU has defined a set of guidelines to conduct the subjective tests [IR08; IR09; IR12]. In the following, three commonly used types of tests are briefly reviewed.

**Double-stimulus methods:** Two different double-stimulus methods, which have originally been developed for television applications, are widely accepted as test methods (cf., Figure 2.8): double-stimulus impairment scale (DSIS), and double-stimulus continuous quality scale (DSCQS) [IR12]. In both methods, the unimpaired and processed sequences are displayed consecutively twice, and the subjects rate the perceptual quality of the processed sequence during the second playback. Depending on the test method, different display orders of the reference and the processed video sequence are performed and different rating scales are applied. In DSIS, which uses a five-point impairment scale to capture the user ratings (cf., Figure 2.9a), the reference sequence is always presented first and the processed sequence sec-




(a) Five-point impairment scale. (b) Double-stimulus continuous quality scale. (c) Five-point rating scale.

Figure 2.9.: Rating scales of the different video quality assessment methods. Adapted from [IR12].

ond [IR12]. DSCQS considers a randomized display order of the reference and the processed representation of the video. The difference between the subjective quality ratings of both representations, which is captured by a continuous quality scale (cf., Figure 2.9b), is used to compute the perceptual quality of the processed representation of the video [PW03].

**Single-stimulus methods:** Single-stimulus continuous quality evaluation (SSCQE) [IR12] and absolute category rating (ACR) [IR08], originally defined for television applications, are two widely used single-stimulus test methods. In SSCQE the observers rate the displayed video continuously throughout the test using the same quality scale as in the DSCQS assessment method (Figure 2.9b), sampled at a frequency of 2 Hz [IR12]. This allows to capture the continuous change of impairments over time perceived by the observers. In the ACR method, similar as in DSCQS and DSIS, the subjects rate the displayed video representations after the playback using a five point scale (Figure 2.9c). In comparison to the double-stimulus assessment methods, ACR enables to analyze more video representations in the same test duration, which, however, might lead to contextual effects. Contextual effects occur when subject ratings are influenced by the degree and ordering of the impairments in the test session [PW03]. It has been shown that by using additional hidden reference sequences among the test sequence set, the contextual effects could be reduced significantly at roughly the same reliability of DSCQS [ITU08].

**Subjective Assessment of Multimedia VIdeo Quality (SAMVIQ):** Unlike the previously presented single- and double-stimulus methods, SAMVIQ [IR09] has been developed for quality assessments of multimedia content and applications. The test is performed scene-wise (cf., Figure 2.10) with a number of processed video sequences and a reference sequence each. To avoid contextual effects throughout the test, the processed video sequences are ordered randomly and a hidden reference sequence is added to the sequence set. Analogous to DSCQS,

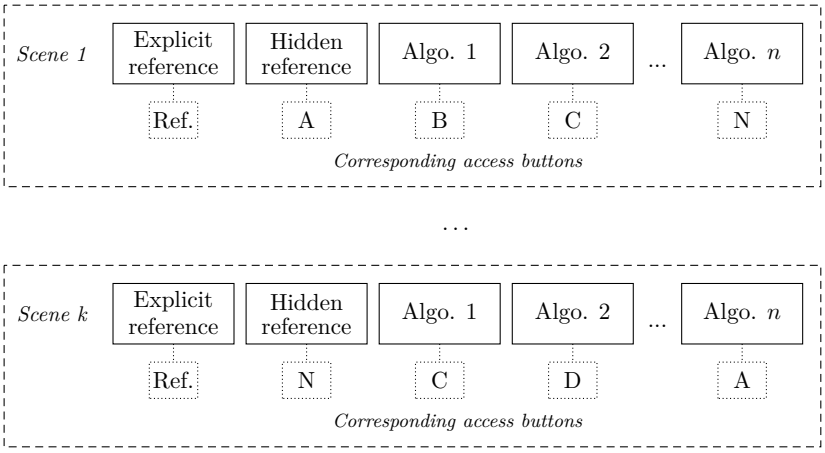



Figure 2.10.: Example of the test organization in SAMVIQ. Adapted from [IR09].

SAMVIQ uses a continuous quality scale for the quality ratings (cf., Figure 2.9b). In SAMVIQ, no strict timing for the rating is specified which allows the subject to make arbitrary comparisons between the processed video sequences with the reference or other processed video sequences at his/her desired pace. This makes it possible for the subject to replay and re-rate the different representations, and as a consequence to reduce erroneous ratings. It has been shown that SAMVIQ enables to capture the subjective ratings more reliably as compared to DSCQS [Bli06] and with a similar reliability as compared to ACR but with a lower number of required test subjects [RP+10].

Because of the higher reliability of the subjective ratings of SAMVIQ in comparison with the other subjective quality assessment methods, SAMVIQ is used to assess the perceptual quality in the further course of this chapter.

#### 2.3.1.2. Video content selection

To ensure generality of the determined results of the subjective quality assessment, it is necessary to select the video sequences for the test carefully. According to ITU [IR08; IR09] it is recommended to use temporal perceptual information (TI) and spatial perceptual information (SI) values, which quantify the spatial and temporal properties of videos, in order to select a representative set of test video sequences. The test videos should contain all extreme conditions, and should "be 'critical but not unduly so' for the system under test" (quoted from [IR12]).

The TI value of a video sequence with $N_f$ frames indicates the amount of temporal change and is calculated by

$$TI = \max_{N_f}[\sigma_{N_x,N_y}\{P_k(x,y) - P_{k-1}(x,y)\}], \tag{2.2}$$

where $\sigma_{N_x,N_y}\{P_k(x,y) - P_{k-1}(x,y)\}$ is the standard deviation of the pixel differences between two consecutive frames ($k$ and $k-1$) with $N_x$ horizontal and $N_y$ vertical pixels. The maximum value over all $N_f$ frames of a video sequence ($\max_{N_f}[.]$) is used to compute the single valued TI measure. In general, video sequences with a high amount of motion lead to large TI values [IR08].

Besides that, the SI value indicates the amount of spatial detail of a video sequence. The calculation of the SI value for a video sequence with $N_f$ frames is

$$SI = \max_{N_f}[\sigma_{N_x,N_y}\{\text{Sobel}(P_k(x,y))\}], \tag{2.3}$$

where the luminance plain of the $k$th frame is processed by the Sobel-filter ($\text{Sobel}(P_k(x,y))$) [SF68]. Similar as for the TI value, first the standard deviation over all pixel values for all frames are computed ($\sigma_{N_x,N_y}\{\text{Sobel}(P_k(x,y))\}$). The maximum value over all $N_f$ frames of a video sequence is used to generate the single valued SI measure. SI values are larger for spatially more complex video scenes [IR08].

One main limitation of the defined TI and SI measures is that they rely on the maximum values of the temporal change of consecutive frames and the spatial information of a sequence, respectively. This, however, is problematic for long video sequences with a significant content change, such as sport sequences (cf., Figure 2.2a). In order to quantify the temporal and spatial properties of the whole video sequence more representatively and reliably, slightly modified versions of TI and SI values have been proposed in [PS11] as temporal activity (TA) and spatial activity (SA) values:

$$TA = \mu_{N_f}[\sigma_{N_x,N_y}\{P_k(x,y) - P_{k-1}(x,y)\}] \tag{2.4}$$

$$SA = \mu_{N_f}[\sigma_{N_x,N_y}\{\text{Sobel}(P_k(x,y))\}]. \tag{2.5}$$

Instead of the maximum value over the $N_f$ frames ($\max_{N_f}[.]$), the mean value ($\mu_{N_f}[.]$) is employed to compute TA and SA.

Throughout this work, TA and SA are used to quantify the temporal and spatial properties of video sequences.

#### 2.3.1.3. Assessment preliminaries and procedure

Besides the thorough selection of the test video set, the selection and number of the test subjects is essential for the reliability of the results of the subjective assessment. Test subjects can be classified as experts and non-experts. Expert subjects are people who are familiar with the intricacies of image and video processing and the different kinds of visual impairments. They have an experienced way of looking at the displayed video sequences and tend to conduct the test too hastily. In order to ensure generality and to capture representative subject votes,

only non-experts should be selected for the assessments [IR12]. Most test recommendations suggest to use at least 15 subjects with normal color vision and visual acuity in the test in order to ensure statistical reliability [IR08; IR09; IR12]. Besides that, a playback duration of 10 to 15 s for the displayed video sequence is commonly recommended in order to get stable and reliable results [IR09].

The actual test procedure is commonly divided into the three main phases *preparation*, *test assessment*, and *post-processing*. In the preparation phase, the test room should be prepared according to the test recommendations [IR08; IR09; IR12]. The test should be explained to the subjects in an oral and written form. A training session prior to the actual test should be performed in order to clarify questions and to accustom the subjects to the test environment and impairments inquired in the test. The ratings recorded during the training session should not be included in the final results.

In the post-processing, the test subjects should be evaluated regarding the reliability of their ratings, and outliers should be removed from the set of subjects. The test standards define rejection criteria which evaluate the correlation between the votes of the individual subjects with the mean values of all subjects of the test [IR08; IR09; IR12].

### 2.3.2. Objective video quality assessment

Objective video quality assessment methodologies are employed in order to estimate the perceptual quality of a video sequence in an algorithmic way without direct human interactions. The metrics are typically applied in a number of different video processing and video transport applications, such as rate controllers installed at the video encoder or for perceptual quality-aware adaptations along the video transport path. In general, objective video quality metrics can be divided, depending on the employed information of the original and compressed representation, into three main classes: *full-reference*, *no-reference*, and *reduced-reference* quality metrics [IR00]. Figure 2.11 displays a system view of the three different objective quality assessment concepts and their deployments in video processing systems.

In a full-reference (FR) metric, the video quality is determined by comparing the reference frames with the processed frames in a pixel-by-pixel manner. Although it has been shown that full-reference assessments offer the highest estimation accuracy of the perceptual quality, the practical applicability of these assessments systems is limited to scenarios where the reference frames are available. To this end, FR assessments are typically applied in source-based video processing, such as rate controllers at the video source, or off-line applications. In Section 3.4 of this thesis, for example, a FR metric has been used in perceptual-quality aware rate control to determine encoding settings for given rate constraints.

In no-reference (NR) quality metrics, the video quality of the processed video is directly assessed without the involvement of the original video source, which offers a higher usability

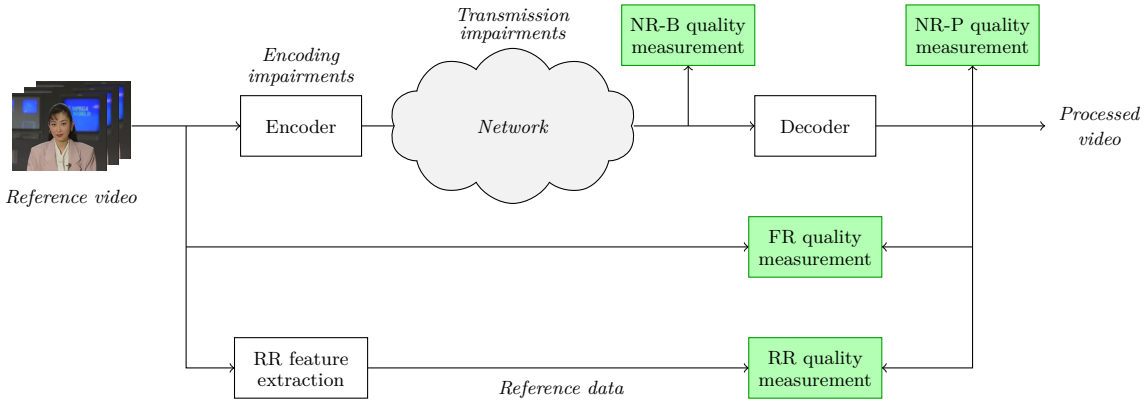



Figure 2.11.: Block diagram of no-reference, full-reference and reduced-reference video quality assessment systems. Adapted from [Pen12].

range compared to FR metrics. In general, there are two types of NR metrics. NR-B metrics consider the bitstream of the video before decoding and NR-P analyze the decoded video [KC+06]. One major drawback of NR metrics is that it is sometimes complicated to differentiate between content and encoding artifacts in the video frames, such as straight borders, which might be either caused by blocking artifacts or an actual part of the video frame [Pen12].

Finally, reduced-reference (RR) metrics employ features which are determined from the original and the processed video. These features need to be transmitted over a separate channel to the video quality measurement point, where the video quality is estimated by comparing the features from both measurement points. In general, RR metrics are able to provide higher accuracies of the video quality estimations as compared to NR models, however, at the cost of a larger transmission overhead caused by the features which need to be transmitted to the measurement point.

#### 2.3.2.1. Objective video quality metrics

In the following, the most commonly used objective video quality metrics are briefly reviewed.

**Peak signal-to-noise ratio:** The most commonly applied FR objective video quality metric in image and video processing is PSNR, which describes the ratio of the maximum power of a source signal and the power of the noise of the distorted image or video frame [WOZ02]. The PSNR is computed by

$$PSNR = 10 \cdot \log_{10} \frac{(2^b - 1)^2}{MSE} \text{dB}, \tag{2.6}$$

where $b$ is the number of bits per pixel, which is set to $b = 8\,\text{bit/px}$ throughout the thesis. The MSE of a compressed image or video frame is computed by a pixel-to-pixel comparison

(a) Original (b) 37.2 dB (c) 35.2 dB (d) 35.1 dB

Figure 2.12.: Original and distorted versions of an example frame of the *Foreman* video [Seq] at different PSNR values.

between the original image ($P(x, y)$) and the processed image ($\hat{P}(x, y)$) as

$$MSE = \frac{1}{N_x \cdot N_y} \cdot \sum_{x=1}^{N_x} \sum_{y=1}^{N_y} (P(x, y) - \hat{P}(x, y))^2, \tag{2.7}$$

where $N_x$ and $N_y$ are the number of horizontal and vertical pixels of a frame, respectively. In video processing the PSNR value of a video sequence is typically computed as the mean over the PSNR values of the luminance component (referred to as $PSNR_Y$) of all $N_f$ video frames of a video sequence, which is computed by

$$\mu_{PSNR_Y} = \frac{1}{N_f} \cdot \sum_{i=1}^{N_f} PSNR_{Y,i}. \tag{2.8}$$

PSNR[4] is widely used to evaluate the distortion of images and video frames due to its simple calculation. However, one main limitation of PSNR is that it does not take the characteristics of the HVS into account and thus offers a low estimation accuracy of the human perception [Gir93; VQE03]. A PSNR value does not necessarily match the actual subjective evaluation of a human observer. To give an example, Figure 2.12 displays different distorted versions of an example frame (Figure 2.12a). Figures 2.12(b, c) display blurred representations of the original image at different PSNR levels (37.2 dB and 35.2 dB), where the distortion is applied to the whole frame. Besides that, in Figure 2.12d the face is almost undistorted and the distortion has primarily been applied to the background of the image. Although the resulting image has approximately the same PSNR value compared to Figure 2.12c, the perceptual quality of Figure 2.12d is commonly rated higher [Ric03]. These findings reveal that the actual perceived quality might vary from the quality ratings of the PSNR values. A second limitation of the PSNR is its content-dependency, since it offers unreliable quality estimations when the quality of videos with diverse content is compared [GHT08].

[4]Throughout this thesis, $\mu_{PSNR_Y}$ is referred to as PSNR.

**Advanced objective video quality metrics:** Two different types of quality metrics have been developed, which aim to solve the limitations of the PSNR metric [Win15]: *HVS-based metrics* and *engineering metrics*. HVS-based metrics take different psycho-visual characteristics of the HVS into consideration, such as contrast sensitivity and multichannel decomposition [WSB03]. They incorporate the error sensitivity of the HVS between the original and processed images by modeling the HVS as a sequential process [WSB03]. HVS-based metrics are typically FR metrics which offer accurate estimations of the perceptual quality, however, are computationally complex. Besides that, HVS-based metrics do not account for other aspects of the visual perception, such as the processing of the results in the brain, which also have a significant impact on the overall visual perception. Further information on HVS-based metrics can be found in [WSB03; WM08; CS+11].

Engineering metrics, on the other hand, assume the HVS as a black box and estimate the overall quality using some content features of the video. Unlike the HVS-based metrics, engineering metrics are typically developed for a specific quality task and employ video features which can be determined from the video frames using computationally efficient algorithms. Various metrics depend on PSNR or MSE values and further extend these by incorporating some video content features to reduce the content-dependency effects of the PSNR [PW02; ODZ07; BRK09]. Structural similarity index (SSIM) offers higher correlations with the human perception than the PSNR-based metrics by taking the structural information of the frames into account, which covers the inter-dependencies between spatially close pixels [WB+04]. SSIM is a single valued measure which expresses the contrast, luminance, and structure of two images after the mean subtraction and variance normalization [WB+04].

The major limitation of the previously discussed SSIM and the PSNR-based engineering metrics is that they do not capture other visual impairments than the spatial quality of the video frames, such as spatial and temporal resolution impairments. To this end, numerous engineering metrics have been developed which consider the joint influence of spatial quality as well as spatial and temporal resolution impacts on the perceptual quality [KD+07; MX+12; FS+07; PS11].

### 2.3.3. Perceptual video quality modeling

In the following subsection, PSNR-based objective video (engineering) quality metrics are reviewed, which estimate the perceptual quality (referred to as $Q$ in the following) of processed representations of a video for both, spatial quality (i.e., image quality) and temporal resolution impairments. To this end, first, three major spatio-temporal video quality metrics are introduced. Second, a subjective test to assess the perceptual quality for spatio-temporal quality impairments is conducted for a representative video set. The results of the subjective assessment are used to train the model parameters of the different metrics and used to

compare the estimation performance of the three metrics[5].

#### 2.3.3.1. Spatio-temporal video quality metrics

In the following, three objective video quality metrics are introduced which estimate the influence of the spatial quality and the temporal resolution on the perceptual quality.

**QM [FS+07]:** The quality metric proposed in [FS+07] (referred to as *QM* in the following) depends linearly on the PSNR value. It additionally takes the temporal estimation weakness of PSNR into account, which underrates the perceptual quality for temporally downsampled processed video sequences [Pen12]. For this purpose, *QM* introduces a video content-dependent temporal compensation factor which depends on the frame rate difference of the original video and the downsampled representation of the video as well as a content-dependent motion factor $\kappa$. The motion factor is computed as the average of the highest 25% of the motion vector magnitudes divided by the frame width.

The perceptual quality using *QM* is estimated by

$$Q_{QM} = a_1 \cdot (PSNR + a_2 \cdot \kappa^{a_3} \cdot (f_{max} - f)) + a_4, \tag{2.9}$$

where $a_1$, $a_2$, $a_3$, and $a_4$ are content-dependent model parameters, which are determined by least squares non-linear fitting, and $f_{max}$ is the highest frame rate considered in the metric. $Q_{QM}$ of Eq. (2.9) has been slightly modified compared to its originally proposed version in [FS+07] by additionally introducing $a_1$ and $a_4$ as scaling factors in order to realize other quality scales than the ones used in the originally proposed version, such as five-point rating scales (cf., Fig 2.9c).

Although *QM* is able to improve the estimation performance of the perceptual quality for temporally downsampled processed video sequences compared to PSNR, it suffers from its linear dependency on the PSNR and its content-dependency issues. This, however, might lead to an overall low estimation performance of the perceptual quality.

**VQMTQ [WMO09; MX+12]:** Unlike *QM*, the quality metric proposed in [WMO09] (referred to as *Wang* in the following) depends directly on the quantization parameter of the encoded video for spatial quality impairments. The metric proposes a separation of the spatial and temporal encoding settings as separate factors. The perceptual quality using *Wang* is estimted by

$$Q_{Wang} = Q_{max} \cdot Q_q(q, f_{max}) \cdot Q_f(f, q_{min}), \tag{2.10}$$

[5] Parts of this study appeared in preliminary form in [LSS].

where $Q_{max}$ is the highest rating of the video quality, $Q_q$ is the factor which captures the influence of the quantization encoding settings, and $Q_f$ is the factor which takes the impact of temporal resolution changes into account. $Q_q$ and $Q_f$ are computed by

$$Q_q(q) = e^{b_1} \cdot e^{-b_1 \cdot \frac{q}{q_{min}}}$$
$$Q_f(f) = \frac{1 - e^{-b_2 \cdot \frac{f}{f_{max}}}}{1 - e^{-b_2}},$$

where $b_1$ and $b_2$ are content-dependent model parameters, $f_{max}$ is the highest frame rate and $q_{min}$ the lowest quantization parameter considered in the metric. *Wang* takes the content-dependency of the spatial quality into consideration, which improves the metric accuracy significantly. The authors extend their metric in [MX+12], where they develop video codec-specific estimators for the content-dependent parameters (referred to as *VQMTQ* in the following). The estimators depend on the motion vectors and video contrast-dependent measures which need to be computed by a separate pre-processor entity from the original video. This process, however, is computationally demanding and requires direct access to the video and internal mechanisms of the video codec.

**STVQM [PS11]:** Similar to *Wang* and *VQMTQ*, *STVQM* models the impact of spatial and temporal encoding settings as separate factors. To this end, in *STVQM* a spatial factor ($SVQM$) that depends on PSNR and a temporal factor ($TVQM$) which depends on the frame rate of the encoded video are introduced. $SVQM$ is modeled as a logistic function

$$SVQM = \frac{Q_{max} - Q_{min}}{1 + e^{-(PSNR + w_s \cdot SA + w_t \cdot TA - \mu)/s}} + Q_{min}, \tag{2.11}$$

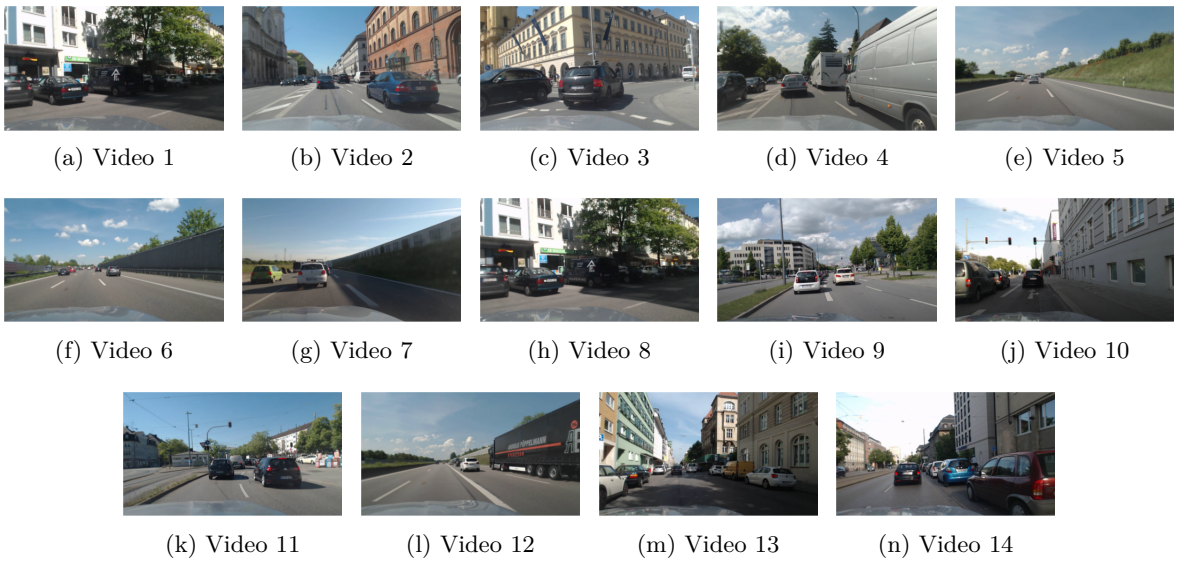
and $TVQM$ as

$$TVQM = \frac{Q_{max} - Q_{min}}{Q_{max}} \cdot \frac{1 + t_a \cdot TA^{t_b}}{1 + t_a \cdot TA^{t_b} \cdot \frac{f_{max}}{f}} + \frac{Q_{min}}{Q_{max}}, \tag{2.12}$$

where $Q_{min}$ is the lowest rating of the video quality, $f_{max}$ is the highest considered frame rate, and $w_s$, $w_t$, $t_a$, and $t_b$ are model factors. The perceptual quality using $STVQM$ is estimated by

$$Q_{STVQM} = SVQM \cdot TVQM. \tag{2.13}$$

*STVQM* employs TA and SA as video content parameters for the spatial and temporal components. This makes it possible to compensate for the contextual issues of PSNR and hence to achieve high estimation accuracies of $SVQM$ [Pen12]. *STVQM* achieves a similar estimation performance as *VQMTQ*, however, with the advantage that less computational demanding estimators for the video content-dependent parameters are required, which makes *STVQM* suitable for real-time video processing systems [PS11].



(a) Video 1 (b) Video 2 (c) Video 3 (d) Video 4 (e) Video 5

(f) Video 6 (g) Video 7 (h) Video 8 (i) Video 9 (j) Video 10

(k) Video 11 (l) Video 12 (m) Video 13 (n) Video 14

Figure 2.13.: Example frames of the *Road* video set recorded with an ADAS front-facing camera.

#### 2.3.3.2. Subjective study

In the following, the estimation performance of the perceptual quality of *QM*, *VQMTQ*, and *STVQM* is determined and compared. To this end, first the setup and the results of the conducted subjective test is introduced, followed by an assessment of the estimation performance of the three video quality metrics.

**Test sequences:** For the investigation, 14 videos from road scenes are selected, which have been recorded with a resolution of 1280x720 at 30 frames per second[6]. The videos are recorded with a prototypical ADAS front-facing camera while driving in urban and highway environments. All sequences have a length of 300 frames. Example frames of the selected videos are displayed in Figure 2.13 and the TA/SA values of the full video sequences are displayed in Figure 2.14. For the further analysis, the video pool is separated into two datasets: a *training set* (video 1-10), which is used to train the model parameters and a *validation set* (video 11-14), which is used to assess the performance for videos outside the training set.

For each uncompressed source video, 12 processed video sequences (PVSs) at four different frame rates (30 fps, 15 fps, 10 fps, 5 fps) are created. For each frame rate level, the videos are encoded at three different PSNR values (42 dB, 38 dB, 34 dB) with H.264/AVC (Main profile) using x264 [Vid]. All videos are encoded with a constant quantization parameter using I-frame only GoPs. In order to realize the same video duration, frame repetitions are generated for the temporally downsampled representations of the videos.

[6]In the remainder of this thesis, the video set is referred to as *Road* video set.

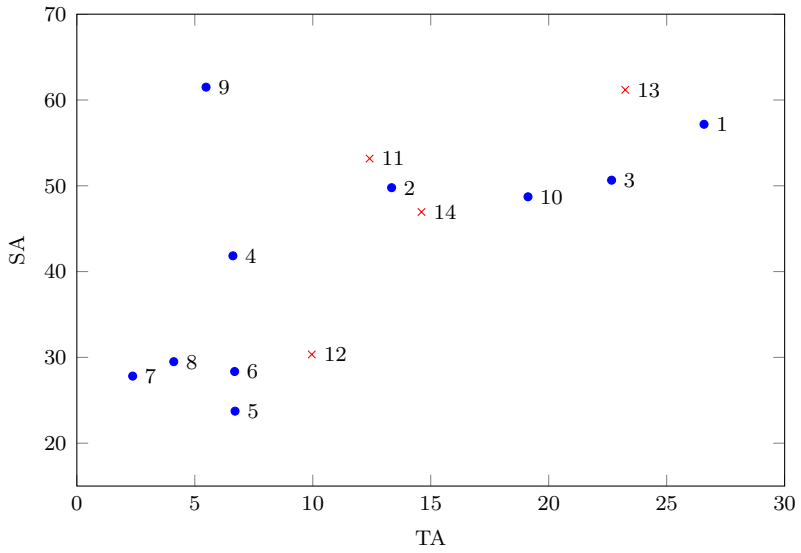



Figure 2.14.: TA and SA values of the *Road* training (•) and validation (×) sets (full video sequences).

**Test setup and procedure:** The perceptual quality of all uncompressed and compressed video sequences is assessed using SAMVIQ [IR09]. For the test development and assessment, the instructions of [IR09] are followed. A graphical user interface according to the guidelines of [IR09] is implemented, where the video is displayed in full resolution in the center of the screen with a gray background. Further information on the implemented interface is given in Appendix A. A five-point rating Likert scale (cf., Figure 2.9c) with values from 1 (worst) to 5 (best) with intermediate steps of 0.5 is used to capture the user ratings. The subjects are able to access the uncompressed reference sequence (referred to as *Ref*), the hidden reference, and the different PVSs of one video scene by clicking on the buttons placed below the video. In order to perform the ratings, the subjects have to watch the full video representations when played the first time but are able to jump back to the previously rated video representations and change the ratings afterwards. Once all representations of one video are rated, a *Next* button is unlocked to proceed to the next video. An *End* button appears if the subjects rated the representations of all video scenes.

Before the actual test, the test setup and process is introduced to the subjects using a printed version of the test (cf., Appendix A) as well as a presentation of the test with a demonstration of the different types of impairments that are assessed during the test. Besides that, the subjects are able to familiarize themselves with the test interface using a mock-up model of the test and to ask questions.

The properties of the hardware setup of the conducted test are displayed in Table 2.1. To avoid fatigue of the subjects, two separate experiments with two groups of persons were performed. In the first experiment, the subjects evaluated videos 1-7, and in the second experiment the subjects rated videos 8-14. In both experiments, 17 non-expert subjects with an average

| Parameter | Settings |
|---|---|
| Type of display | LCD (CCFL, TN-panel) |
| Display size | 15.4” |
| Display resolution | WUXGA (1920x1200) |
| Model | LG.Philips LP154WU1-TLB1 |

Table 2.1.: Subjective test hardware configuration.

age of 31 participated and each of the subjects rated in total 91 PVSs. The duration of both experiments was on average 25 minutes each. To remove the bias in the quality ratings effected by the subjects' feeling regarding the video content, differential mean opinion score (DMOS) values for each PVS are computed as

$$DMOS = \frac{1}{N_s} \cdot \sum_{i=1}^{N_s} (r_i(PVS) - r_i(SRC)) + Q_{max}, \tag{2.14}$$

where $r_i(PVS)$ and $r_i(SRC)$ are the ratings of the $i$th subject for the PVS and uncompressed video sequence (SRC), respectively, and $N_s$ is the number of subjects which rated the sequences for the corresponding video.

**Subjective data post processing:** The screening procedure defined in [IR09] is used to exclude outliers from the ratings. For this purpose, the Spearman rank correlation $SC_i$ and the Pearson correlation $PC_i$ are computed for each subject $i$ versus the mean ratings of all subjects. For each subject $i$ the following rejection criterion is applied:

$$\textbf{If } C_i \leq U_R \textbf{ Then} \text{ reject subject } i,$$

where $U_R$ is the rejection threshold[7] and $C_i = \min(SC_i, PC_i)$. After the screening procedure, 15 subjects in the first experiment and 16 subjects in the second experiment are identified as valid. Figure 2.15 displays the mean DMOS values of the subjective test results for the videos of the training set versus the frame rate for the different PSNR values along with the 95% confidence interval (CI), which is determined using the Student's t-distribution.

#### 2.3.3.3. Performance assessment

In the following, the performance in estimating the perceptual video quality is determined for the three objective video quality metrics.

**Evaluation metrics:** In order to investigate the estimation performance of the objective video quality metrics, the Pearson correlation (PC) and the root mean square error (RMSE) are

[7] $U_R$ has been set to 0.85, which is the default value suggested in SAMVIQ [IR09].

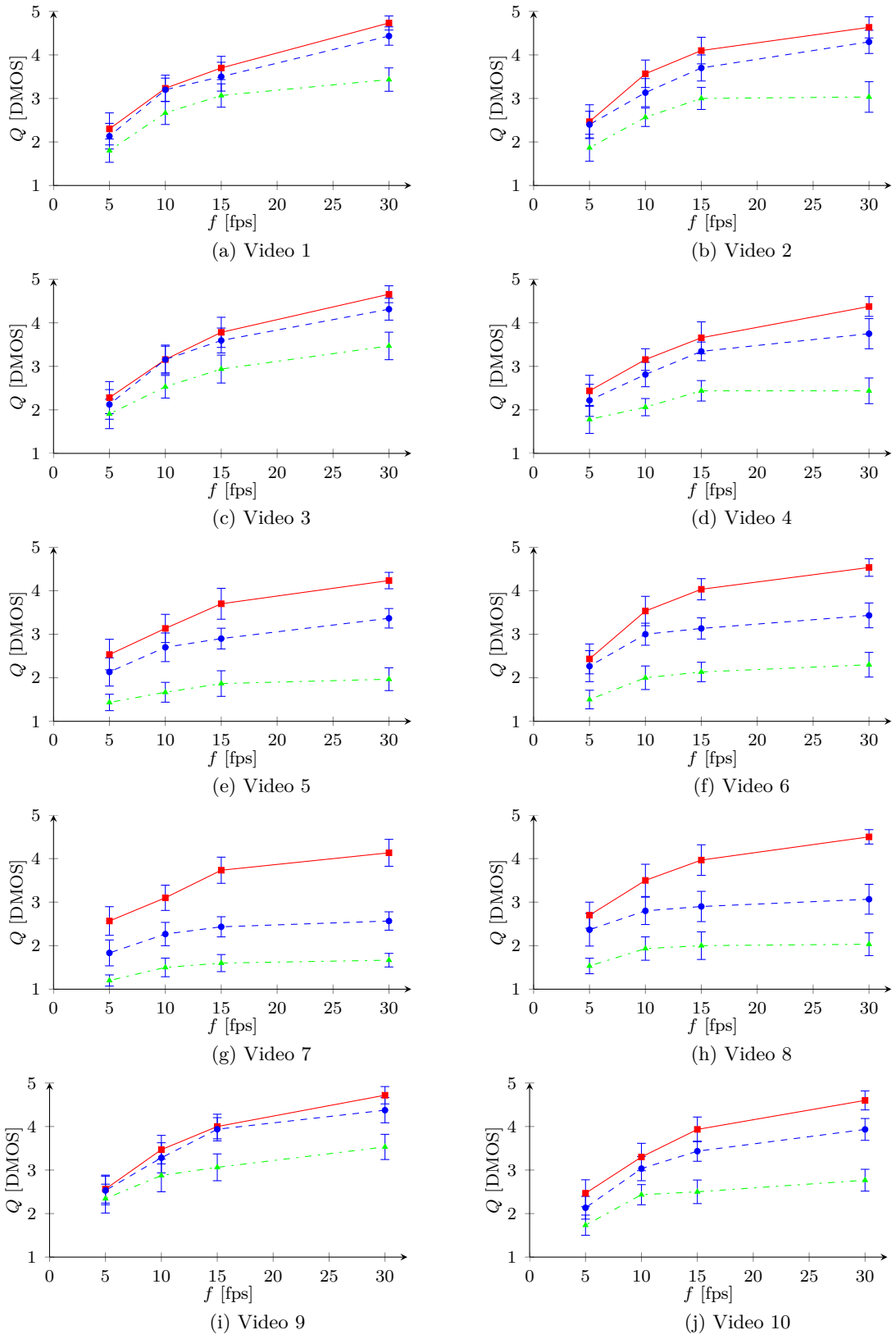



Figure 2.15.: Perceptual quality $Q$ versus $f$ for different PSNR values determined using the subjective test for videos of the *Road* training set: 42 dB (—■—), 38 dB (- -●- -), 34 dB (· -▲- ·) with 95% CI (——).

used as performance measures, which are commonly used for the evaluation of subjective quality metrics [VQE08].

The PC is a statistical metric which indicates the linearity between two random variables [PF98]. Applied to objective video quality metrics, the PC describes the linearity between the metric estimations and the subject ratings of the test assessment. The PC is calculated as

$$PC = \frac{\sum_{i=1}^{N_v} (r_i - r_{mean}) \cdot (\hat{r}_i - \hat{r}_{mean})}{\sqrt{\sum_{i=1}^{N_v} (r_i - r_{mean})^2} \cdot \sqrt{\sum_{i=1}^{N_v} (\hat{r}_i - \hat{r}_{mean})^2}}, \tag{2.15}$$

where $N_v$ denotes the number of videos considered in the investigation. The estimations from the metrics for the $i$th video and the mean over all videos are denoted by $\hat{r}_i$ and $\hat{r}_{mean}$, respectively, whereas $r_i$ and $r_{mean}$ are the corresponding subjective ratings and the mean value over all subjects. The PC is a unitless measure which is defined between 0 and 1. Values close to 1 indicate a high metric estimation performance.

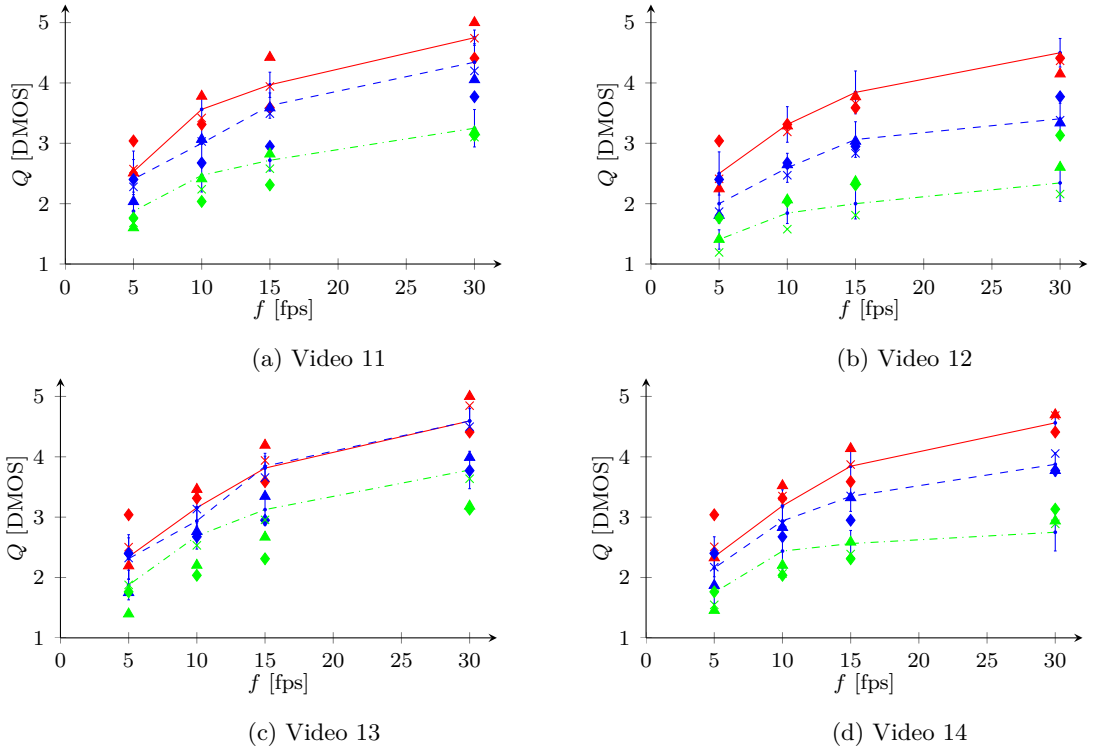
RMSE is an error measure that describes the accuracy of the model estimations. The RMSE is calculated as

$$RMSE = \sqrt{\frac{1}{N_v - u} \sum_{i=1}^{N_v} (r_i - \hat{r}_i)^2}, \tag{2.16}$$

where $u$ is the number of subjective quality-dependent model parameters. The inclusion of $u$ in the error determination is considered to take the model complexity into account [Pen12]. The smaller the RMSE value, the better is the estimation performance.

**Performance comparison:** The model parameters of $QM$, $VQMTQ$, and $STVQM$ are trained with the subject ratings from the subjective test with the videos of the training set using least squares non-linear fitting. In order to evaluate the performance of the models and to investigate the robustness for videos outside the training set, the estimation performance is also assessed for the videos of the validation set.

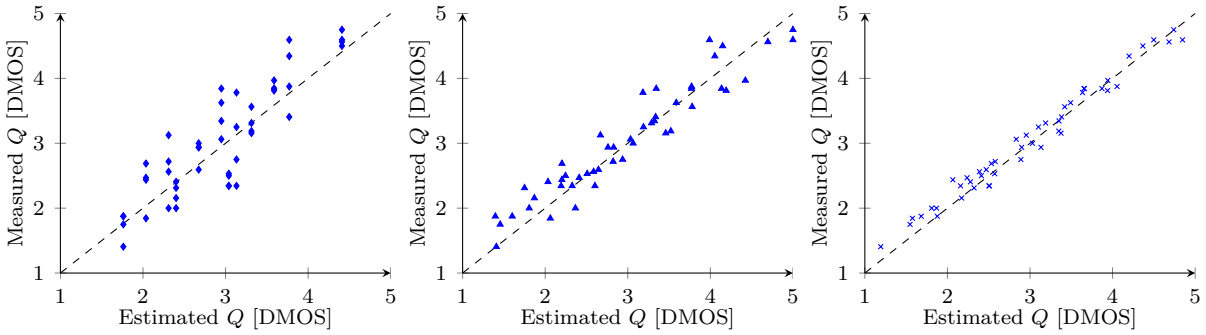
Figure 2.16 displays the measured $Q$ along with the estimated $Q$ values for videos of the validation set. Furthermore, Figure 2.17 displays the monotonicity characteristics of all three metrics for the validation videos. The results show that $STVQM$ and $VQMTQ$ offer a similar estimation performance with only small inaccuracies and a high correlation with the measured $Q$ values from the subjective assessment. The estimation performance of the $QM$ metric offers the worst estimation performance, for both the correlation and the deviation from the measured $Q$ values. The numerical results listed in Table 2.2 underline these findings, where additionally the PC and RMSE values for the videos of the training and validation set are listed.




(a) Video 11 (b) Video 12

(c) Video 13 (d) Video 14

Figure 2.16.: Performance evaluation of *QM* [FS+07] (◇), *VQMTQ* [MX+12] (△), *STVQM* [PS11] (×) for videos of the *Road* validation set; measured $Q$ obtained from subjective test for 42 dB (——), 38 dB (- - -), 34 dB (-·-·-) with 95% CI (——).




(a) *QM* (b) *VQMTQ* (c) *STVQM*

Figure 2.17.: Performance evaluation of *QM* [FS+07], *VQMTQ* [MX+12], and *STVQM* [PS11] for videos of the *Road* validation set: measured $Q$ vs. estimated $Q$.

| VQM | Training set | | Validation set | |
|---|---|---|---|---|
| | PC | RMSE [DMOS] | PC | RMSE [DMOS] |
| *QM* [FS+07] | 0.843 | 0.47 | 0.877 | 0.42 |
| *VQMTQ* [MX+12] | 0.950 | 0.30 | 0.955 | 0.29 |
| *STVQM* [PS11] | 0.983 | 0.25 | 0.991 | 0.17 |

Table 2.2.: *QM* [FS+07], *VQMTQ* [MX+12], and *STVQM* [PS11] estimation performance: PC and absolute RMSE values for videos of the *Road* training and validation set.

## 2.4. Automotive systems

Over the last decade, automobiles evolved from simple vehicles for personal traveling and transportation to elements in the Internet-of-things which actively exchange information with other nodes in the Internet [FS+14]. This has been primarily driven by the increased computational resources of the electronic control units (ECUs)[8], and the advances of radio access networks. Larger capacities and the enhanced mobility support of contemporary cellular radio access networks of the third and fourth generation enable the uplink and downlink of data traffic even at high velocities. Besides that, vehicles are nowadays able to drive autonomously in a multitude of driving situations enabled by different ADAS systems. The recent advances in the ADAS domain have mainly been driven by developments in the near-field and far-field sensor technology domain and the evolution of advanced ADAS algorithms.

The following section builds the background for the automotive related topics of this thesis. In Section 2.4.1 an overview of ADAS sensor systems employed in modern vehicles is given, followed by an introduction to automotive communication systems in Section 2.4.2.

### 2.4.1. Sensors for advanced driver assistance services

ADAS systems installed at modern vehicles aim at supporting the driver during various driving tasks and thus increase the safety and convenience on the road. Typical ADAS applications are pedestrian detection [DW+12], collision avoidance [VE03], congestion assistance [Kra08], and adaptive cruise control (ACC) [WDS09]. The different ADAS systems use information about the status, the dynamics, and the surroundings of the vehicle determined by different sensor systems. In the following, the major sensor technologies to capture the status of the vehicle and information about the surroundings are briefly introduced.

**Vehicle dynamics and positioning:** In modern vehicles, several sensor systems are employed to gather information about the dynamics of the vehicle [Sch09]. The velocity of the vehicle is determined by an induction-based wheel-speed sensor installed at the front wheels. The lateral and longitudinal accelerations are computed from the displacement of a spring-mounted mass using a capacitive sensor. In order to determine the yaw rate, which describes the vehicle's angular velocity around the vertical axis, the tilting movement of a swinging impeller is measured by a capacitive sensor. Further details about the sensor systems to determine the dynamics of the vehicle can be found in [Fle01; Rei11].

Information about the location of the vehicle is determined using a satellite-based global navigation satellite system (GNSS). Most modern vehicles typically employ NAVSTAR Global

[8] In vehicular deployments, ECUs constitute embedded computational entities which are responsible for one or more computational tasks.

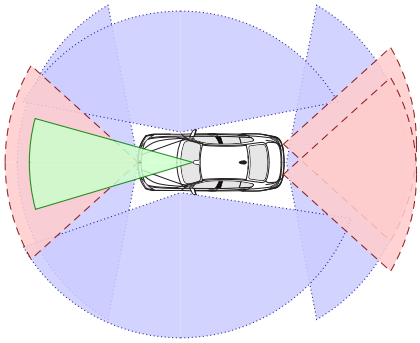


Figure 2.18.: Surround sensor configuration of a typical modern vehicle: LIDAR-scanner ( ), RADAR ( ), mono ADAS front-facing camera ( ). Adapted from [AS+12].

Positioning System (GPS) and GLONASS to determine the absolute latitudinal and longitudinal position [Gro13].

**Environment perception:** Modern vehicles are typically equipped with numerous sensor technologies to capture objects in the vicinity of the vehicle, which differ in their suitability for different ADAS systems and their accuracy. In Figure 2.18 the most commonly used near-field and far-field sensor technologies are displayed.

*Radio detection and ranging (RADAR)* sensors are employed in both near-field (24 GHz) and far-field sensing (77 GHz) [Sch05]. In order to capture the distance to other objects, the RADAR sensor emits electro-magnetic pulses and measures the power of the reflected impulse in emission direction, which mainly depends on the diameter of the reflected pulse. The emitter and receiver of the RADAR sensor are typically installed at the same unit. RADAR systems are especially interesting for automotive ADAS applications, since they are almost insensitive to weather and atmospheric influences. Besides that, the velocities of objects can be directly measured based on Doppler measurements. RADAR sensors, however, offer only a limited lateral resolution [Det89]. In modern vehicles, RADAR sensors are typically employed for ACC applications.

*Light detection and ranging (LIDAR)* sensors exploit the principle of time measurement of emitted light impulses [RG05]. To this end, the distance to other objects is determined by the runtime of a reflected light impulse with a wave length of around 900 nm and the speed-of-light. The light impulses can be either emitted by a laser diode with flexible optics or by a sensor which is constructed by multiple rows of diodes with fixed optics. In comparison to RADAR sensors, LIDAR sensors are more sensitive to atmospheric and weather influences, however, offer a higher lateral resolution which can be used to determine the dimensions of objects [FD+02]. LIDAR sensors are employed in various safety related ADAS applications, such as collision avoidance and pedestrian detection.

Modern vehicles are increasingly equipped with camera systems for *direct* and *indirect* camera-based ADAS applications [Rah09]. Direct camera-based ADAS applications present the frames captured by the cameras directly to the driver. Indirect camera-based ADAS systems, on the other hand, use advanced image processing algorithms to automatically detect objects or situations in the field-of-view. Objects and situations are typically detected in the single frames using edge detectors and are further processed through advanced filters [GKL05]. Due to the usage of advanced image processing algorithms, indirect camera-based ADAS systems are able to support a multitude of ADAS applications as opposed to LIDAR and RADAR systems, which are mainly used to determine the distance and the velocity to other objects. Direct camera-based ADAS applications are typically employed in parking situations (e.g., top-view ADAS systems, rear-view ADAS systems), whereas indirect camera-based ADAS applications are employed to support the driver in various driving tasks (e.g., lane departure warnings or pedestrian detection). Similar as for LIDAR systems, the major drawback of camera-based ADAS systems is that they suffer from the unreliability in unfavorable light conditions (e.g., low light or extreme weather conditions).

Further information on other sensor systems employed in modern automotive systems (such as ultrasonic) can be found in [Fle01; WDS09].

In order to enable highly automated driving (HAD) systems with strict reliability constraints, a sensor setup is required which is able to reliably capture objects and critical situations in the surroundings of the vehicle. Single-sensor perception systems might not be able to provide reliable and robust information due to weaknesses inherited by their sensing technology. As a remedy, modern HAD deployments use a heterogeneous sensor setup with redundant and complementary sensing technologies (LIDAR, RADAR, and indirect camera-based ADAS systems) and employ further sensor fusion algorithms to combine the information gathered by the different sensor technologies [AS+12].

### 2.4.2. Automotive communication technologies

In automotive deployments, functions and data are typically distributed over several ECUs and sensor systems. In modern premium vehicles, for example, up to 70 different on-board[9] ECUs are employed which exchange thousands of variables and signals among each other [Alb04]. These numbers are even higher if off-board applications deployed at backend servers in the Internet are considered.

In order to exchange the data among the different on-board ECUs and off-board entities, different wired and wireless communication technologies are used in modern vehicles. An

[9] *On-board* refers to the communication of ECUs installed locally at the vehicle and *off-board* refers to the communication of the vehicle's ECUs with entities outside the vehicle, e.g., other vehicles or servers deployed in the Internet.

overview of both communication technologies and an introduction to the ECU architecture of modern vehicles is given in the following.

**Wired on-board communication technologies:** Until the 1990s, ECUs were initially interconnected by point-to-point links for inter-ECU communication. However, as the number of ECUs increased significantly over the years, different fieldbuses have been introduced, which allowed to replace several dedicated cables by serial buses and as a consequence to decrease the amount of required cables substantially [NHB05]. In modern vehicles, five major bus systems are employed:

- The *controller area network (CAN)* bus system, originally developed by *Bosch GmbH* in the 1980s, is the most commonly employed inter-ECU communication technology in modern vehicles. Typically, two different types of CAN networks are considered [NS+05]: (i) real-time control CANs for power-train and chassis functions with a data rate of 250 or 500 kbit/s and (ii) CANs for body-domain functions with a data rate of 125 kbit/s.
- *Local Interconnect Network (LIN)* is a low-cost in-vehicle network standard, which offers low network speeds of up to 20 kbit/s and is typically employed for comfort applications, such as climate control, or light sensors [NHB05].
- *Media Oriented Systems Transport (MOST)* is a network standard for inter-ECU communication of multimedia applications. In modern vehicles, MOST is typically used for the interconnection of infotainment applications, such as video displays, active speakers, or digital radios [NHB05].
- *FlexRay* has been developed as a major progress towards the requirements of *x-by-wire* (e.g., break-by-wire, steer-by-wire) and high data rate in-vehicle applications [NS+05]. It is capable of data rates of up to 10 Mbit/s and is considered as a potential replacement for the existing buses and increasingly deployed in modern vehicles [NS+05].
- *Automotive Ethernet* offers even higher data rates of up to 100 Mbit/s. It is considered as the upcoming standard for the in-vehicle transmission of bandwidth-demanding video streams from ADAS cameras and high-definition entertainment multimedia traffic [Rah09; LVH11].

More information on wired automotive communication technologies can be found in [NHB05; NS+05; SD14].

**Wireless off-board communication technologies:** Automotive systems increasingly employ data from off-board systems which introduces the need for wireless connectivity to the Internet. The potential off-board applications mainly cover three domains: (i) off-board ADAS applications which exchange road traffic relevant information between the vehicles using vehicle-

to-infrastructure communications (e.g., real-time traffic information), (ii) telematic domain applications (e.g., off-board navigation), and (iii) multimedia applications (e.g., online video and audio entertainment applications) [LB+15]. Modern vehicles are increasingly equipped with a heterogeneous RAN modem [HF+09] and multi-standard antennas [Eki14] which offer connectivity to a multitude of wireless communication standards. In the following, an overview of the major RAN technologies employed in modern European vehicles is given.

Over the last decade, *dedicated short range communication (DSRC)* has been developed as an automotive wireless communication technology to offer both, ad-hoc based vehicle-to-vehicle and infrastructure-based vehicle-to-infrastructure communication. For vehicle-to-infrastructure communication, *road-site units* located along the road are required, which offer the connectivity to backend servers and the Internet. *ITS-G5* [ETS12] represents a DSRC implementation of the European Telecommunications Standards Institute (ETSI) intelligent transportation systems and is able to support peak data rates of up to 27 Mbit/s [ITU14]. However, such an automotive-specific technology suffers from the typical chicken-and-egg deployment problem, since a certain penetration of DSRC-equipped vehicles is required before road operators install required road site units [LB+12b]. Besides that, the high throughput demands of future off-board automotive applications might exceed the capacities of DSRC systems [LB+15].

*UMTS* is the cellular communication standard of the third generation [HT00] and has been developed by 3GPP. Compared to its predecessor, *Global System for Mobile Communications (GSM)* with its extensions *General Packet Radio Service (GPRS)* and *Enhanced Data rates for GSM Evolution (EDGE)*, it offers higher data rates, lower latencies, and a higher spectral efficiency, which is mainly achieved by the usage of code-multiplex mechanisms [HT00]. With the introduction of the UMTS extensions *High Speed Packet Access (HSPA)* and *HSPA+*, theoretical uplink and downlink data rates of up to 168 Mbit/s (downlink) and 22 Mbit/s (uplink) can be achieved [JB+09]. Despite the high peak throughput, it has been shown that both GSM- and UMTS-based cellular networks are not able to satisfy the stringent quality-of-service (QoS) requirements of future automotive off-board applications [BG+09].

To meet the increasing demand in mobile data traffic and the advanced QoS requirements of new mobile and automotive applications, *LTE* has been developed by 3GPP [HT09]. Compared to UMTS and HSPA, it offers even higher data rates, lower latencies and a higher network capacity. In the downlink, orthogonal frequency-division multiple access (OFDMA) is applied, which separates the overall downlink stream in small sub-bands. This makes it possible to reduce the influence of frequency-selective fading and to enable the usage of multiple-input and multiple-output (MIMO) transmissions which offer peak data rates of 300 Mbit/s in the downlink. In the uplink, peak data rates of 85 Mbit/s with a single antenna system [DPS11] can be achieved. Although LTE has been developed to offer connectivity at vehicular velocities, it has been shown that bandwidth-demanding video streaming applications might not be supported in high velocity scenarios [LB+12b].

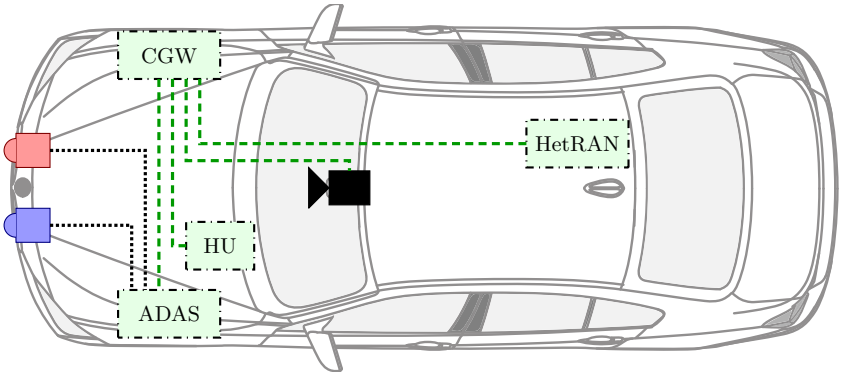



Figure 2.19.: Exemplary ECU architecture (ECUs marked with ▭) with sensor systems (LIDAR marked with ▭, RADAR marked with ▭ and ADAS front-facing camera marked with ▬) and interconnecting bus systems (CAN marked with ····· and Ethernet marked with - - - -).

LTE-Advanced [HT11] radio access networks will provide even higher data rates and an advanced support for high velocity (vehicular) terminals. Unlike GSM, UMTS, and LTE, LTE-Advanced is not yet commonly deployed by mobile network operators as of today.

Further information about wireless automotive connectivity can be found in [EL+13; SD14; LB+12b; LB+15].

**ECU architecture of modern vehicles:** Modern vehicles are equipped with a mixture of different ECUs for different application domains. The ECUs are typically developed as black boxes by automotive suppliers. Access to settings and functionalities of the ECUs, such as video encoders and data streams, is only granted through dedicated interfaces with restricted access. As a consequence, a direct access to raw data streams, such as uncompressed video streams from ADAS cameras and processing functions of ECUs, is typically not possible. In accordance with low production costs, the ECUs are typically purpose-built with limited computational capacities.

Figure 2.19 displays an example ECU architecture and the corresponding interconnection of a typical modern vehicle with a selection of the ECUs which are employed for the automotive applications in the course of this thesis:

- The *central gateway (CGW) ECU* acts as a central relay and transmits traffic between the different ECUs and wired on-board bus systems [Rah09].
- The *head unit (HU) ECU* is a central processing entity within the vehicle which performs all infotainment related processes, such as on-board and remote human-machine interface (HMI) functions and multimedia applications [WT06; EPS10].
- The *heterogeneous RAN (HetRAN) ECU* offers connectivity to IP-based cellular networks, such as GPRS/EDGE, UMTS/HSPA, LTE, and DSRC. It is connected via Ether-

net to the CGW and offers Internet connectivity to a multitude of ECUs in the vehicle [HF+09].

- The *ADAS ECU* performs all ADAS related computations, such as the filtering and fusion of the data from different sensor systems [Hin11]. It is connected to the CGW with an Ethernet connection and acts as a sink for the sensor data of the LIDAR and RADAR sensors which are directly connected to the ECU via the CAN bus. The computed information can be transmitted to the HU for on-board ADAS services or to off-board entities via the HetRAN ECU.

# Chapter 3

# Video bit rate model for perceptual quality-aware rate control

In this chapter, the impact of the temporal resolution, spatial quality, and GoP settings on the bit rate of encoded videos is investigated. Based on analytical evaluations, a video bit rate model[1] is developed which makes it possible to estimate the bit rate of encoded videos for different quantization parameter, frame rate, GoP length, and GoP structure settings. Estimators of the content-dependent model parameters for H.264/AVC encoded videos are developed as functions of the temporal and spatial activity which can be determined from the uncompressed source video. Furthermore, the proposed video bit rate model is applied to the solution of a perceptual quality-aware rate control problem to determine quantization parameter and frame rate encoding settings for given rate constraints.

## 3.1. Introduction

Enhanced features and the high computational capacity of modern consumer electronic devices as well as the evolution of modern cellular radio access networks enable wireless video streaming of live and on-demand video content to and from consumer electric devices, such as smartphones, tablet computers, or connected vehicles. To adapt the video stream to the dynamically changing performance of wireless channels, different adaptive streaming systems have been proposed, which have been discussed earlier in Section 2.2. Traditional RTP/UDP-based video streaming systems and AHS systems use single layer representations of the same source video and require the video encoded at certain desired bit rates. To achieve the bit rates, rate controllers at video encoders are required to adapt the encoding settings accordingly. Most state-of-the-art rate controllers employ rate models that only consider modifications of the spatial quality by an adaptation of the quantization settings or a joint adaptation

[1] The bit rate model presented in this chapter has previously been proposed in [LS14].

of quantization and temporal resolution encoding settings by additionally taking frame rate modifications into account. Typically, these rate models consider a fixed GoP length and structure for all temporal resolutions with an IPP...P GoP.

While these GoPs are suitable for traditional RTP/UDP-based streaming, AHS based streaming systems require rate controllers that additionally consider the adaptation of the GoP length for temporally downsampled representations of the encoded video in order to realize video segments of a specified duration. Besides that, it has been shown that further bit rate savings of 8% on average can be realized while additionally considering B-frames in the GoPs and hence taking the influence of the GoP structure on the bit rate into account [LS14]. In order to realize advanced rate control for video segments of a defined duration, it is necessary to investigate the impact of each encoding setting on the video bit rate, and thus, to be able to estimate the bit rate accurately when the encoding settings are adjusted.

Based on these motivations, this chapter proposes a bit rate model which captures the influence of the spatial quality resulting from adaptations of the quantization parameter and the impact of the temporal resolution resulting from a variation of the frame rate. In addition, the bit rate model captures the influence of the GoP length and GoP structure which makes it possible to estimate the bit rate of encoded video segments of a defined duration. The model has structural similarity with the models proposed in [MX+12] and [LM+14a] which consider separate factors for the spatial and temporal encoding settings. It further introduces two additional factors which take the GoP length and GoP structure into account. The proposed bit rate model depends on constant parameters and content-dependent factors that need to be calculated directly from the source video. However, the calculation of the content-dependent parameters for each source video is computationally complex. To reduce the computational complexity and to make the rate model applicable in rate controllers, content-dependent estimators based on temporal and spatial activity values are developed. The proposed bit rate model is applied to H.264/AVC video coding and shows a high estimation performance with the measured bit rates of the encoded videos. Finally, a perceptual quality-aware rate control problem is defined to determine quantization parameter and frame rate encoding settings for given rate constraints of video segments of a specified duration. A solution of the problem is developed which takes the proposed bit rate model and an objective video quality metric into consideration.

The remainder of this chapter is organized as follows. Section 3.2 reviews the related work on video bit rate models. Section 3.3 presents the proposed bit rate model and the developed TA- and SA-based estimators for the content-dependent model parameters. In Section 3.4 the proposed bit rate model is applied to perceptual quality-aware rate control in order to determine optimal quantization parameter and frame rate encoding settings for given rate constraints. Finally, Section 3.5 summarizes this chapter.

## 3.2. Related work

Rate control algorithms at video encoders for hybrid video encoding typically employ video bit rate models and video quality metrics[2] to determine the encoding parameters for achieving desired video bit rates. In the following, rate models which consider modifications of the spatial encoding settings and joint modifications of the spatio-temporal encoding settings[3] are surveyed.

**Spatial rate models:** In most state-of-the-art rate control algorithms for hybrid video coding, the spatial resolution and the frame rate are fixed, and the encoder varies the quantization encoding settings to achieve desired target bit rates.

The rate control algorithms proposed for MPEG-4 Part 2 [LCZ00] and H.264/AVC [LSW05] directly depend on the quantization parameter $q$ and use quadratic rate models of the form

$$R(q) = \frac{a_1}{q} + \frac{a_2}{q^2} + a_3, \tag{3.1}$$

where $a_1$, $a_2$, and $a_3$ are content dependent model parameters which rely on the mean absolute difference (MAD) of the video frames.

A different class of rate models employ $\rho$, which is the percentage of zero quantized transform coefficients for a specific quantization parameter. [KHM01] proposes a linear $\rho$-based rate model for typical video coding systems as

$$R(q) = c \cdot (1 - \rho(q)), \tag{3.2}$$

where $c$ is a content-dependent constant. $\rho$-based models typically offer a high accuracy in estimating the bit rates of encoded videos, however, with the requirement to first encode several or even all blocks before the rate allocation can be done.

**Spatio-temporal rate models:** Advanced video bit rate models have been developed which, besides spatial encoding settings, also take the influence of frame rate adaptations into consideration.

The video bit rate model for H.264/AVC encoding proposed in [WMO09] takes both the impact of quantization parameter and frame rate modifications on the bit rate into account and introduces separate factors for both encoding settings. The overall model is defined as

$$R(q, f) = R_0 \cdot R_q(q) \cdot R_f(f), \tag{3.3}$$

[2] A survey of video quality metrics is given in Section 2.3.

[3] In the course of this chapter, *spatial encoding settings* refer to quantization parameter settings and *temporal encoding settings* refer to frame rate settings.

where $R_q(q)$ accounts for the quantization parameter $q$, $R_f(f)$ incorporates the influence of the frame rate $f$, and $R_0$ is a video-specific bit rate factor. All three factors rely on content-dependent model parameters which need to be determined for each video separately. The rate model considers a static IPP...P GoP and does not take influences of the GoP length and structure into consideration. As a consequence, the rate model does not allow for estimating the bit rate of video segments of a specified duration while changing the temporal resolution, which, however, is typically applied in AHS deployments. The authors extend their metric in [MX+12] by estimating the video-specific model parameters by content features, which are determined from the motion-estimation scheme and frame difference information of the underlying codec. The authors propose a video-codec dependent pre-processor to extract the features, which, however, is computationally demanding [PS11; LM+14a].

In [LM+14a], a video bit rate model for video segments of a fixed duration for H.264/AVC encoded videos is proposed, which considers frame rate and quantization parameter modifications. The model follows a similar approach as [MX+12], however, employs features based on temporal and spatial activity values to estimate the content-dependent model parameters which can be computed from the source video. In comparison to [MX+12], the computational complexity could be reduced and the proposed bit rate model is able to capture downsampled temporal resolutions for video segments of a defined duration. However, the model has only been trained for one specific segment duration and is only able to consider IPP...P GoPs.

## 3.3. Proposed video bit rate model

The bit rate model follows a similar approach as [MX+12] and [LM+14a] and considers the influence of each encoding setting separately. To this end, an impact factor for each individual encoding setting is introduced. The spatial factor $R_s$ captures the influence of quantization parameter modifications. Besides that, the temporal factor $R_t$ takes the influence of temporal resolution variations on the bit rate of encoded videos into account. The influence of the GoP settings are captured by a GoP length factor $R_{GoPL}$, which considers the length of a GoP ($n$) and a GoP structure factor $R_{GoPS}$, which additionally depends on the number of consecutive B-frames in a GoP ($m$). Furthermore, a maximum bit rate factor $R_{max,I}$ is introduced which is the bit rate at full frame rate ($f_{max}$), the minimum quantization parameter ($q_{min}$), and an I-frame only GoP structure ($n = 1$, $m = 0$). The overall bit rate model can be formulated as

$$R(q, f, n, m) = R_{max,I} \cdot R_s(q) \cdot R_t(f) \cdot R_{GoPL}(n) \cdot R_{GoPS}(m, n). \tag{3.4}$$

The modeling of the individual factors for the different encoding settings is performed in two steps. First, the impact of each individual encoding setting on the bit rate of the encoded videos is studied analytically in Section 3.3.1. Second, estimators for the video content-dependent model parameters based on temporal and spatial activity values are developed in

(a) Video 1 (b) Video 2 (c) Video 3 (d) Video 4 (e) Video 5

(f) Video 6 (g) Video 7 (h) Video 8 (i) Video 9 (j) Video 10

Figure 3.1.: Example frames of the videos from the *CIF* video set.

Section 3.3.2. Finally, the estimation performance of the model is assessed in Section 3.3.3.

### 3.3.1. Analytical rate factor modeling

To investigate the impact of the individual parameters on the video bit rate, two different video sets are introduced:

- *CIF*: The video set consists of 10 multimedia domain video sequences in Common Intermediate Format (CIF) (352x288 pixels) with a frame rate of 30 fps and a total length of 300 frames [Seq]: *Akiyo* (1), *Container* (2), *Football* (3), *Foreman* (4), *Hall* (5), *Mobile* (6), *Mother & Daughter* (7), *Paris* (8), *Highway* (9), and *Deadline* (10). Figure 3.1 shows example frames of the video sequences and the corresponding TA and SA values of the full video sequences, computed from the raw videos using Eqs. (2.4) and (2.5), are displayed in Figure 3.2.

- *Road*: The video set consists of 14 road-view video sequences recorded with a front-facing camera of a vehicle with a spatial resolution of 1280x720 pixels, a frame rate of 30 fps and a sequence length of 300 frames. The video set including example frames and an overview of the TA and SA properties has been introduced earlier in Section 2.3.3.2.

Similar to the investigation in Section 2.3.3.2, both video sets are separated into two datasets each. To this end, *training sets* (*CIF*: video 1-6; *Road*: video 1-10) are introduced which are used to train the model parameters and the corresponding estimators. Separate *validation sets* (*CIF*: video 7-10; *Road*: video 11-14) are used to verify the robustness of the developed bit rate model and the estimators for videos outside the training sets. The videos of the training sets are selected to cover the extremes of the TA and SA values and to consider the characteristics of typical videos of both sets.

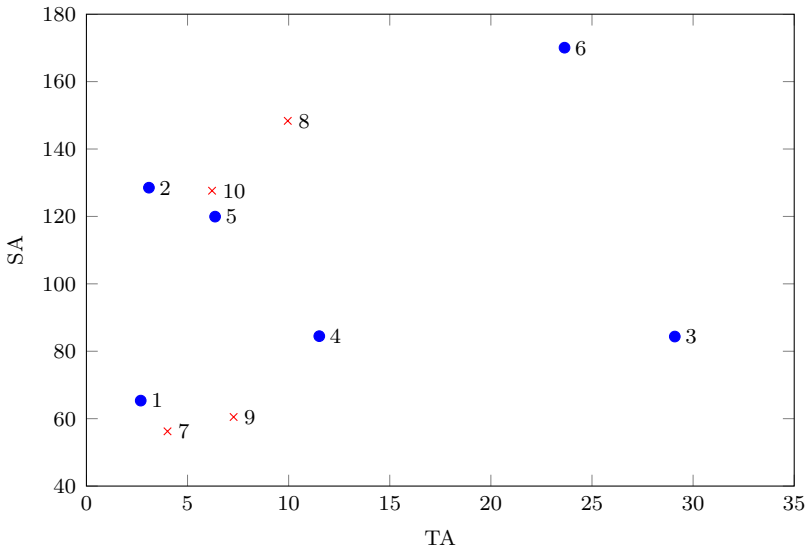



Figure 3.2.: TA and SA values of the *CIF* training (●) and validation (×) sets (full video sequences).

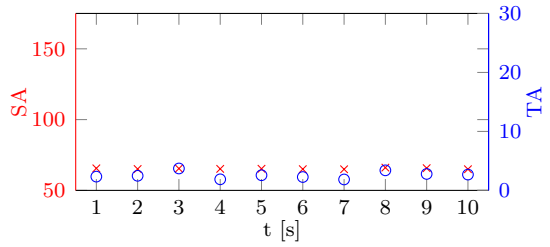
For the further investigation, each video sequence is split into ten sub-sequences of 30 frames length, which corresponds to a duration of $\tau = 1\,\mathrm{s}$ per segment at full frame rate. For each video segment, the TA and SA values are computed separately. Figure 3.3 displays the TA and SA values of all video segments of the *CIF* video set. It can be observed that the variation of the TA and SA values for the segments of the same sequence is high for some video sequences. For example, video 3 (*CIF* video set) shows a TA value change of 61% and a SA value change of 66% over the whole video sequence.

For the analysis, different PVSs are created for each video segment with the encoding settings listed in Table 3.1. The encoding settings are selected such that all GoPs of one video sequence offer the same number of frames depending on the $n$ and $f$ settings. All PVSs are encoded in H.264/AVC Main profile using x264 [Vid].

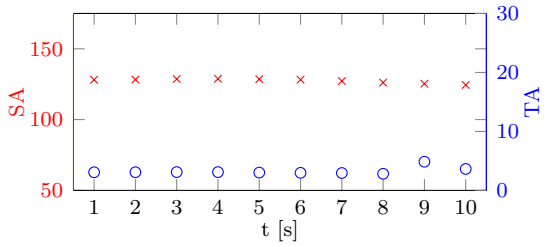
In the following, the influencing factors are analyzed separately and analytical models for the different factors are developed.

| Encoding setting | min | max | Steps |
|---|---|---|---|
| $q$ | 24 | 45 | stepsize: 1 |
| $f$ [fps] | 5 | 30 | 5, 10, 15, 30 |
| $m$ | 0 | 4 | stepsize: 1 |
| $n$ | 1 | 30 | 1, 2, 3, 5, 6, 10, 15, 30 |

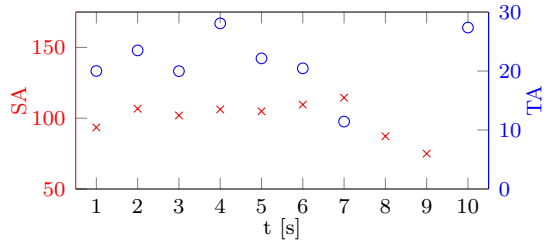
Table 3.1.: Considered $q$, $f$, $m$, and $n$ encoding settings in the video bit rate model.



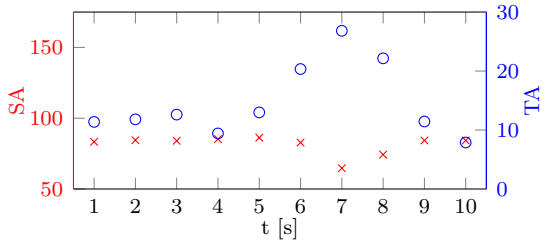

(a) Video 1



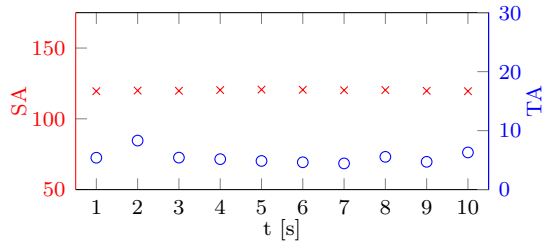

(b) Video 2



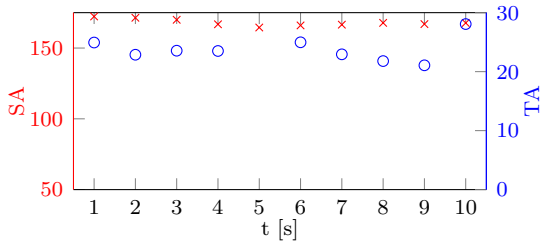

(c) Video 3



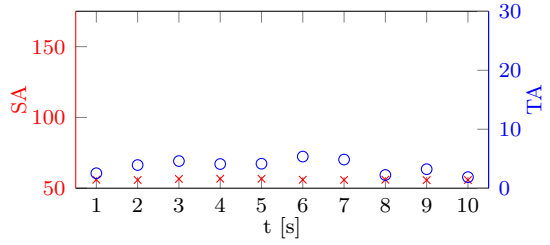

(d) Video 4



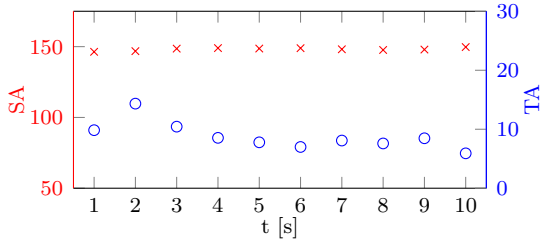

(e) Video 5



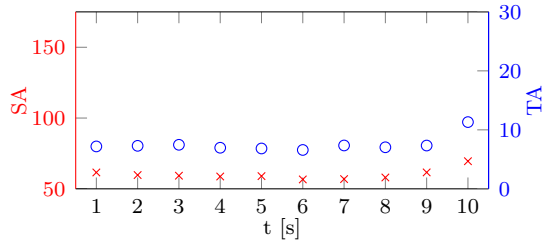

(f) Video 6



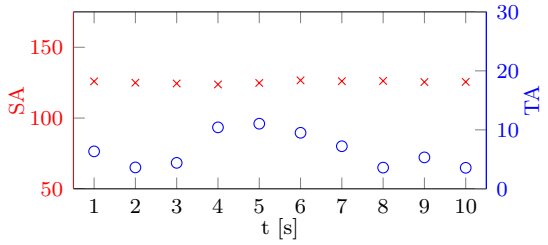

(g) Video 7




(h) Video 8

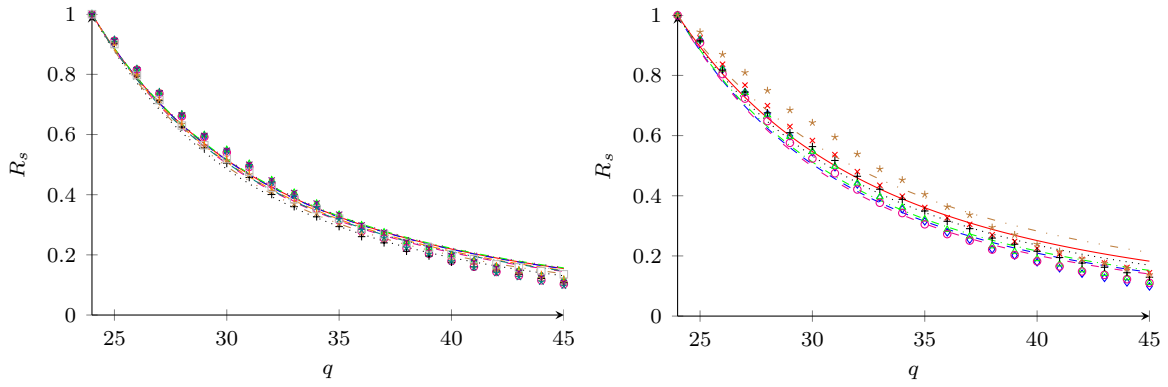



(i) Video 9




(j) Video 10

Figure 3.3.: SA (×) and TA (○) values of video segments of the *CIF* video set ($\tau = 1\,\mathrm{s}$).




(a) Measured $R_s$ (dots) and estimated $R_s$ (lines) of Eq. 3.5 for the first video segment of the *Road* training set: 1 (×, ——), 2 (◇, - - -), 3 (△, -·-·-), 4 (○, - - ), 5 (+, ·····), 6 (⋆, - · · -), 7 (□, ·····), 8 (, - - - -), 9 (⋆, -·-·-), 10 (▽, · · · · ).

(b) Measured $R_s$ (dots) and estimated $R_s$ (lines) of Eq. 3.5 for the first video segment of the *CIF* training set: 1 (×, ——), 2 (◇, - - -), 3 (△, -·-·-), 4 (○, - - ), 5 (+, ·····), 6 (⋆, - · · -).

Figure 3.4.: $R_s$ versus $q$ for videos of the *Road* and *CIF* training sets.

#### 3.3.1.1. Spatial factor

The spatial factor $R_s(q)$ describes the influence of the quantization parameter $q$ on the video bit rate. To capture the decrease of the bit rate for increasing $q$ values, PVSs at full frame rate ($f = f_{max}$) using an I-frame only GoP ($n = 1$, $m = 0$) are created and normalized by $R_{max,I}$. Figure 3.4 shows the relation between $R_s$ and the quantization parameter for training videos of the *Road* and *CIF* video set. It can be derived that $R_s$ is 1 at the smallest considered $q$ value ($q_{min}$) and reduces down to 0 for large $q$ values. Similar as in [MX+12] and [LM+14a] an inverse power function is used to model $R_s$:

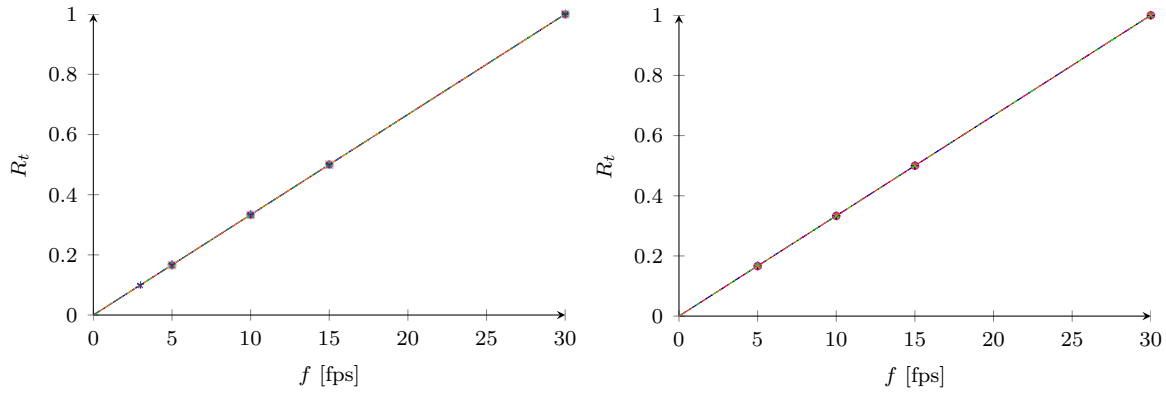
$$R_s(q) = \left(\frac{q}{q_{min}}\right)^{-s}, \tag{3.5}$$

where $s$ is a content-dependent parameter to define how fast $R_s$ is decreasing for increasing $q$ values.

#### 3.3.1.2. Temporal factor

$R_t(f)$ describes the influence of the frame rate $f$ on the bit rate of the encoded video. In order to analyze the decrease of the bit rate for a reduction of the temporal resolution, PVSs at the best considered quantization parameter setting ($q = q_{min}$) with an I-frame only GoP ($n = 1$, $m = 0$) are generated. In Figure 3.5 the measured bit rate relative to $R_{max,I}$ are displayed for the training videos of the *Road* and *CIF* video sets. $R_t$ can be modeled as a linear function with a constant slope which is equal to 0 for still images and increases linearly to 1 at $f_{max}$:

$$R_t(f) = \left(\frac{f}{f_{max}}\right). \tag{3.6}$$




(a) Measured $R_t$ (dots) and estimated $R_t$ (lines) of Eq. 3.6 for the first video segment of the *Road* training set: 1 (×, ——), 2 (◇, - - -), 3 (△, - · - · -), 4 (○, – – ), 5 (+, ·····), 6 (⋆, – · · · -), 7 (□, ·····), 8 (|, – – – -), 9 (⋆, – · – · ), 10 (ϒ, · · · · ).

(b) Measured $R_t$ (dots) and estimated $R_t$ (lines) of Eq. 3.6 for the first video segment of the *CIF* training set: 1 (×, ——), 2 (◇, - - -), 3 (△, - · - · -), 4 (○, – – ), 5 (+, ·····), 6 (⋆, – · · · -).

Figure 3.5.: $R_t$ versus $f$ for videos of the *Road* and *CIF* training sets.

Unlike $R_s$, $R_t$ is independent of the video content and does not rely on any content-dependent parameters.

#### 3.3.1.3. GoP length factor

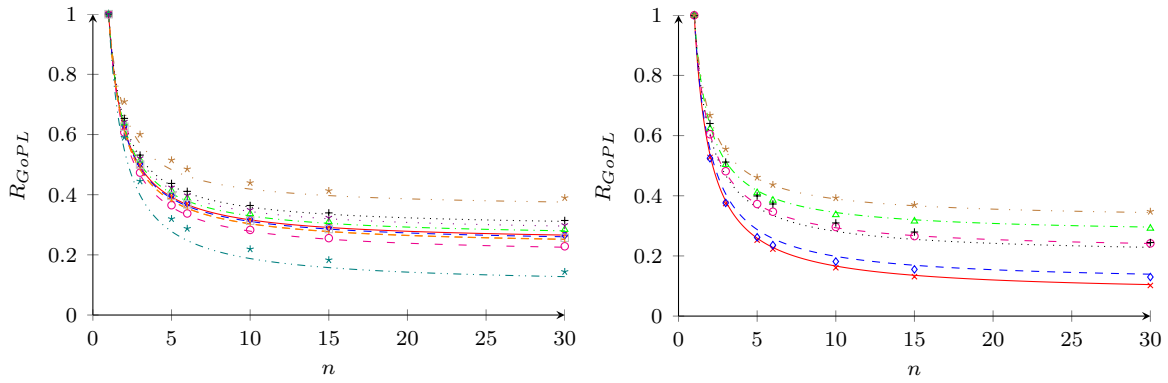
$R_{GoPL}(n)$ considers the impact of P-frames and the GoP length on the video bit rate. In order to model the impact of the GoP length, the uncompressed video sequences are encoded at the best spatio-temporal encoding settings ($q = q_{min}$, $f = f_{max}$) using the previously defined GoP lengths with an IPP…P GoP structure ($m = 0$). Figure 3.6 displays the measured rate of the encoded videos relative to $R_{max,I}$ for the training videos of both video sets. $R_{GoPL}$ is 1 at $n = 0$ (I-frame only GoP) and converges to a video content-specific offset for large $n$ values. Similar as $R_s$, $R_{GoPL}$ can be modeled as an inverse power function with a content-dependent offset:

$$R_{GoPL}(n) = l_1 \cdot \left(\frac{1}{n}\right) + l_2, \tag{3.7}$$

where $l_1$ is the content-dependent weight for the inverse power function, and $l_2$ is the factor that describes the content-dependent relative offset for large $n$ values.

#### 3.3.1.4. GoP structure factor

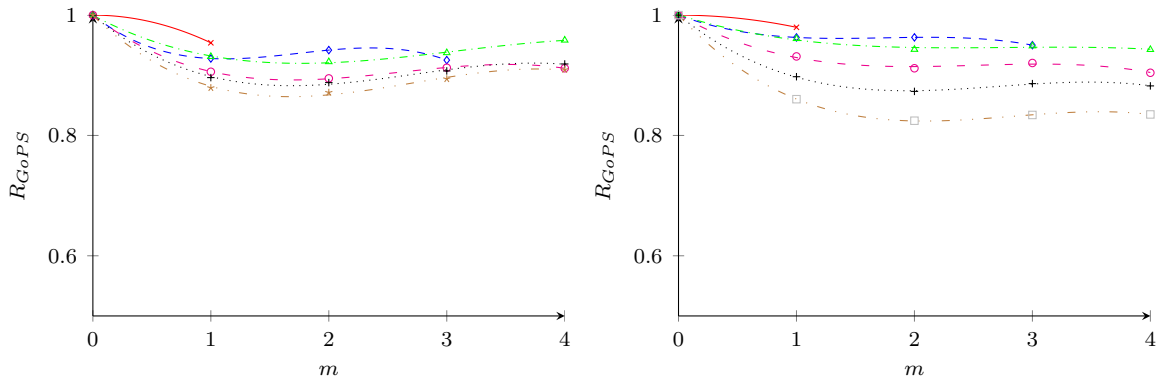
The GoP structure factor $R_{GoPS}(m, n)$ additionally accounts for the influence of B-frames and describes the influence of the number of consecutive B-frames $m$ for a given GoP length $n$ on the video bit rate. $R_{GoPS}$ considers $m \leq 4$, since it has been shown that the bit rate increases and the average PSNR of the encoded video decreases significantly for $m > 4$ [WCK06], which is not convenient for video compression. Furthermore, the GoP structure is limited to GoPs




(a) Measured $R_{GoPL}$ (dots) and estimated $R_{GoPL}$ (lines) of Eq. 3.7 for the first video segment of the *Road* training set: 1 (×, ——), 2 (◇, - - -), 3 (△, - · - ·), 4 (○, - - ), 5 (+, ·····), 6 (⋆, - · · -), 7 (□, - - - -), 8 (, - - - -), 9 (⋆, - · - ·), 10 (×, · · · ·).

(b) Measured $R_{GoPL}$ (dots) and estimated $R_{GoPL}$ (lines) of Eq. 3.7 for the first video segment of the *CIF* training set: 1 (×, ——), 2 (◇, - - -), 3 (△, - · - ·), 4 (○, - - ), 5 (+, ·····), 6 (⋆, - · · -).

Figure 3.6.: $R_{GoPL}$ versus $n$ for videos of the *Road* and *CIF* training sets.




(a) Measured $R_{GoPS}$ (dots) and estimated $R_{GoPS}$ (lines) of Eq. 3.8 for the first segment of video 1 (*Road* video set) for $n$: 3 (×, ——), 5 (◇, - - -), 6 (△, - · - ·), 10 (○, - - ), 15 (+, ·····), 30 (□, - · · -).

(b) Measured $R_{GoPS}$ (dots) and estimated $R_{GoPS}$ (lines) of Eq. 3.8 for the first segment of video 1 (*CIF* video set) for $n$: 3 (×, ——), 5 (◇, - - -), 6 (△, - · - ·), 10 (○, - - ), 15 (+, ·····), 30 (□, - · · -).

Figure 3.7.: $R_{GoPS}$ versus $m$ for the considered $n$ values for video 1 of the *Road* and *CIF* video sets each.

where B-frames can only rely on P-frames and not on other B-frames, i.e., $m \leq n-2$. $R_{GoPS}$ is modeled using a third order polynomial:

$$R_{GoPS}(m,n) = g_1(n) \cdot m^3 + g_2(n) \cdot m^2 + g_3(n) \cdot m + 1, \tag{3.8}$$

where $g_1(n)$, $g_2(n)$, and $g_3(n)$ are the factors of the polynomial, which depend on the GoP length $n$. Figure 3.7 displays $R_{GoPS}$ for different $n$ values for one video segment of both video sets each.

| Factor | Value | Feature |
|---|---|---|
| $s$, $l_1$, $l_2$, $g_1$, $g_2$, $g_3$ | $TA$ | $TA^{\pm 1}$, $\log(TA)$, $e^{\pm TA}$ |
| $s$, $l_1$, $l_2$, $g_1$, $g_2$, $g_3$ | $SA$ | $SA^{\pm 1}$, $\log(SA)$, $e^{\pm SA}$ |
| $g_1$, $g_2$, $g_3$ | $n$ | $n^{\pm 1}$, $\log(n)$, $e^{\pm n}$ |

Table 3.2.: Set of TA- and SA-based features used for the estimator of $R_{max,I}$, and the model parameter estimators of $R_s(q)$ and $R_{GoPL}(n)$. Set of $n$-based features are additionally employed for the model parameter estimators of $R_{GoPS}(m, n)$.

### 3.3.2. Content-based model parameter estimation

The proposed rate model considers a separation of the impact of spatial, temporal, and GoP encoding settings as individual factors which all rely on content-dependent model parameters. In the following section, estimators of the content-dependent model parameters based on TA and SA features are developed which can be computed from the uncompressed video.

#### 3.3.2.1. Content features

Similar as in [LM+14a], a set of TA- and SA-based features is used for the development of the estimators of the content-dependent model parameters. Besides TA and SA, different elementary functions[4] of TA and SA are introduced. In a pre-investigation, the single features which show a high correlation with the content-dependent parameters are selected as candidates for the further estimator development process. Table 3.2 lists the considered TA- and SA-based feature set used for the development of the estimators of $R_{max,I}$, and the estimators of the model parameters of $R_s(q)$ and $R_{GoPL}(n)$. The interaction terms of the different factors are calculated as the products of the single features. The estimators of the model parameters of $R_{GoPS}(m, n)$ additionally depend on features which employ the GoP length $n$.

#### 3.3.2.2. Model parameter estimation

It has been shown in [LM+14a] that the single TA- and SA-based features might not be able to estimate the model parameters accurately. However, by linearly combining several of the previously introduced features, the estimation performance of the model parameters might improve significantly. Therefore, the generalized linear regression methodology (GLM) proposed in [MN90] is used to combine different features in an iterative process. For example, the estimator of a parameter $y$ (referred to as $\widehat{y}$) is developed as

$$\widehat{y} = \sum_{i=1}^{L} \alpha_i \cdot f(x_i) + \alpha_0, \tag{3.9}$$

[4] *Elementary functions* cover logarithms, exponentials, trigonometric and hyperbolic functions, and their inverses.

where $f(x_i)$ are the considered features in the estimators with $\alpha_i$ as the corresponding weighting factors. To realize generic results that are also valid for samples outside the training set, the leave-out-one cross-validation error (CVE) is used as a measure to iteratively combine the different features for each estimator. To compute the CVE, $K-1$ out of $K$ samples are used to train and determine the weighting factors of the investigated estimator. The remaining video is used as a test sample and used to compute the squared fitting error in this round. This procedure is performed $K$ times using a different testing sample in each round. The CVE for the feature set using the training samples is calculated as the mean of the squared fitting errors of the testing sample of all $K$ rounds. To develop the estimators for the model parameters, the features are selected and combined using the iterative stepwise feature selection approach proposed in [MX+12]. At the first iteration step, the single feature that offers the lowest CVE with respect to the real model parameter is determined. In a second iteration round, a second feature is investigated that in combination with the first feature offers a lower CVE. This iteration process is repeated until no further CVE reductions can be achieved while adding more features to the feature set. In each iteration round, the model parameters $\alpha_i$ are determined by using least squares non-linear fitting.

It has been shown in [Ma11] that the model parameter estimators developed for a video set of one spatial resolution lead to an overall poor estimation performance for videos of a different spatial resolution. To this end, separate estimators for the model factors of the *Road* and *CIF* video sets are developed using GLM. Table 3.9 and Table 3.10 list the feature weights of each content-dependent parameter estimator for the *Road* and for the *CIF* video set using the features and feature combinations defined in Table 3.2.

#### 3.3.2.3. Temporal and spatial activity dependent rate model

The developed model parameter estimators are now integrated into the rate factors[5] of $R_s(q)$, $R_{GoPL}(n)$, and $R_{GoPS}(m,n)$, and further integrated into the overall bit rate model of Eq. (3.4). The resulting spatio-temporal rate model which considers the influence of the GoP length and structure ($STRM^+$) is

$$R_{STRM^+}(q,f,m,n) = \widehat{R}_{max,I} \cdot \widehat{R}_s(q) \cdot R_t(f) \cdot \widehat{R}_{GoPL}(n) \cdot \widehat{R}_{GoPS}(m,n). \tag{3.10}$$

### 3.3.3. Performance evaluation

The estimation performance of the rate factors which depend on the proposed TA- and SA-based estimators is assessed individually. Furthermore, the estimation performance of $STRM^+$ with the measured bit rates of the encoded videos is determined and compared to other bit

[5]The TA- and SA- dependent rate factors are referred to as $\widehat{R}$, while $R$ represents the individual rate factors.

| Perf. metric | *Road* | *CIF* |
|---|---|---|
| PC | 0.934 | 0.944 |
| RMSE [kbits/s] | 1589.12 | 830.99 |

Table 3.3.: $\widehat{R}_{max,I}$ estimation performance for the videos of the *Road* and *CIF* training sets: PC and absolute RMSE.

| Factor | Perf. metric | *Road* | *CIF* |
|---|---|---|---|
| $\widehat{R}_s(q)$ | PC | 0.998 | 0.996 |
| | RMSE | 0.0238 | 0.0365 |
| $R_t(f)$ | PC | 0.999 | 0.999 |
| | RMSE | 0.0005 | 0.0004 |
| $\widehat{R}_{GoPL}(n)$ | PC | 0.999 | 0.999 |
| | RMSE | 0.0294 | 0.0444 |
| $\widehat{R}_{GoPS}(m,n)$ | PC | 0.924 | 0.947 |
| | RMSE | 0.0864 | 0.0613 |

Table 3.4.: $\widehat{R}_s(q)$, $R_t(f)$, $\widehat{R}_{GoPL}(n)$, $\widehat{R}_{GoPS}(m,n)$ estimation performance for the videos of the *Road* and *CIF* training sets: PC and absolute RMSE.

rate models for one fixed GoP length and different GoP structures. In some video processing systems, the TA and SA values might not be available for each individual video segment and instead only be determined for a longer video window length. To this end, the influence of the TA and SA window length on the bit rate estimation accuracy is investigated.

#### 3.3.3.1. Bit rate estimation performance

In Table 3.3 the PC and the absolute RMSE of the TA- and SA-based $\widehat{R}_{max,I}$ with the measured $R_{max,I}$ values are listed for the training videos of the *Road* and *CIF* video set. The results show that the estimation performance is high with a PC of roughly 0.94 for both video sets, a RMSE of roughly 830 kbit/s for the *CIF* video set, and a RMSE of approximately 1600 kbit/s for the *Road* video set. The measured $R_{max,I}$ values are in the range of $R_{max,I} \in [1.8, 9.7]$ Mbit/s for the *CIF* and in the range of $R_{max,I} \in [4.8, 20.9]$ Mbit/s for the *Road* video set.

Table 3.4 lists the estimation performance of the individual rate factors using the developed TA- and SA-based estimators. Similar to $\widehat{R}_{max,I}$, a high estimation performance can be achieved with a PC of larger than 0.92 and an absolute RMSE of less than 0.09 taking all factors into consideration.

Finally, the overall model performance of the TA- and SA-based bit rate model is determined for all considered $q$, $f$, $m$, and $n$ encoding settings listed in Table 3.1. Table 3.5 lists the PC and relative RMSE (normalized by $R_{max,I}$) of the measured bit rate versus the estimated bit rate calculated using $STRM^+$ for the *Road* and the *CIF* video sets. For both video sets, the proposed TA- and SA-based rate model is able to achieve a high estimation performance with a

| Video set | Perf. metric | Training set | Validation set |
|---|---|---|---|
| *Road* | PC | 0.975 | 0.966 |
| | % RMSE | 3.72 | 3.84 |
| *CIF* | PC | 0.978 | 0.979 |
| | % RMSE | 3.14 | 2.69 |

Table 3.5.: $STRM^+$ estimation performance for the videos of the *Road* and *CIF* training and validation sets: PC and RMSE (normalized by $R_{max,I}$).

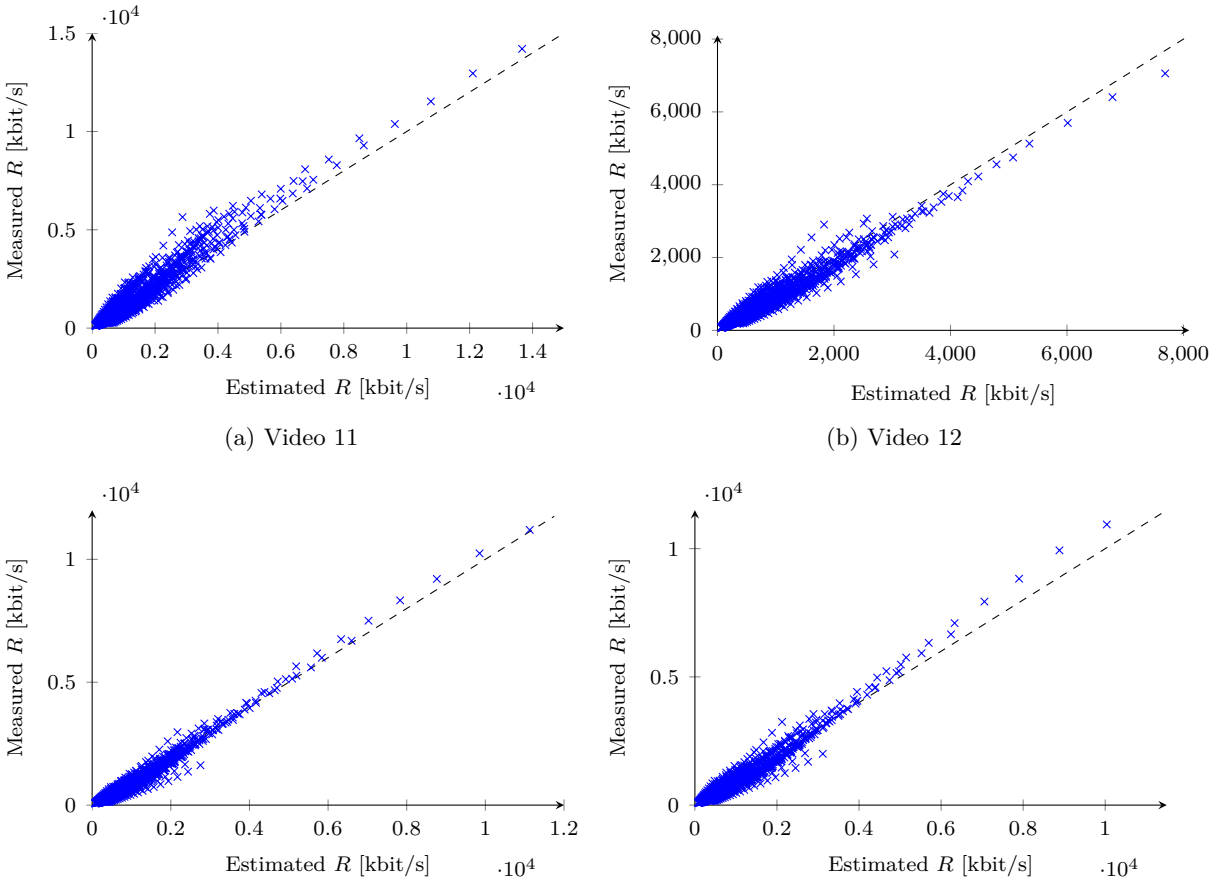
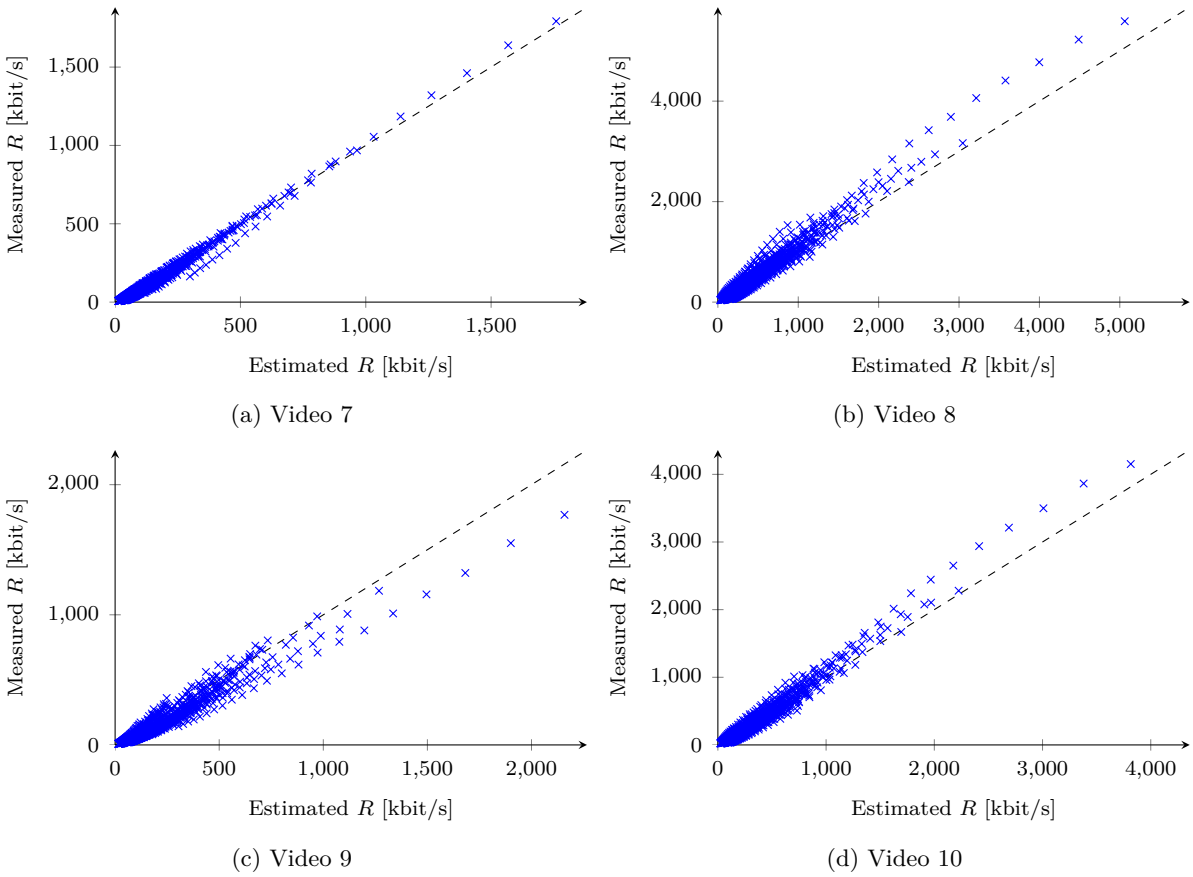



Figure 3.8.: Measured $R$ versus estimated $R$ determined using $STRM^+$ for videos of the *Road* validation set.




(a) Video 7 (b) Video 8

(c) Video 9 (d) Video 10

Figure 3.9.: Measured $R$ versus estimated $R$ determined using $STRM^+$ for videos of the *CIF* validation set.

PC of around 0.97 and a RMSE relative to $R_{max,I}$ of less than 4% for the training sets. Similar results for the validation sets verify the robustness of $STRM^+$ for the bit rate estimation of videos outside the training sets. Figure 3.8 and Figure 3.9 display the measured versus the estimated bit rate for the first video segment of the validation videos for the *Road* and the *CIF* video set, respectively. It can be observed for the *Road* video set that $STRM^+$ tends to underestimate the bit rate for videos with low SA values and to overestimate the bit rate for videos with high SA values. For videos of the *CIF* video set, on the other hand, $STRM^+$ tends to underestimate the bit rate for videos with high SA values and to overestimate the bit rate of videos with low SA values. The reason for this contrary trend lies in the different estimators of STRM's model parameters which are employed for both video sets.

#### 3.3.3.2. Performance comparison

The estimation performance of $STRM^+$ is compared to the bit rate model of [LM+14a] (referred to as *STRM* in the following) and the bit rate model of [MX+12] (referred to as *Ma* in the following). Similar to $STRM^+$, *STRM* uses TA- and SA-based estimators for the

| GoP structure | Perf. metric | *Ma* [MX+12] | *STRM* [LM+14a] | $STRM^+$ |
|---|---|---|---|---|
| $m = 0$ | PC | 0.987 | 0.998 | 0.989 |
| | RMSE [kbit/s] | 484.1 | 492.15 | 518.68 |
| $m = 4$ | PC | 0.995 | 0.935 | 0.864 |
| | RMSE [kbit/s] | 575.34 | 569.47 | 447.64 |

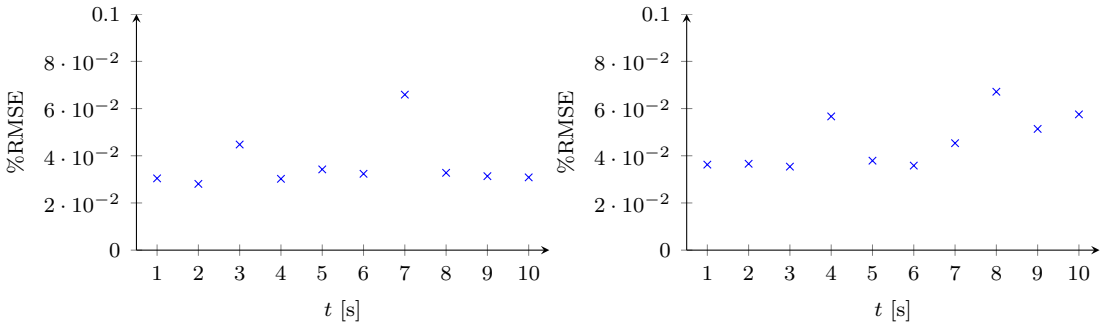
Table 3.6.: Estimation performance of *Ma*, *STRM*, and $STRM^+$ for the videos of the *Road* validation set: PC and absolute RMSE.

| GoP structure | Perf. metric | *Ma* [MX+12] | *STRM* [LM+14a] | $STRM^+$ |
|---|---|---|---|---|
| $m = 0$ | PC | 0.987 | 0.986 | 0.987 |
| | RMSE [kbit/s] | 190.33 | 175.66 | 111.73 |
| $m = 4$ | PC | 0.988 | 0.987 | 0.923 |
| | RMSE [kbit/s] | 216.84 | 199.03 | 89.19 |

Table 3.7.: Estimation performance of *Ma*, *STRM*, and $STRM^+$ for the videos of the *CIF* validation set: PC and absolute RMSE.

video content-dependent model parameters. The content-dependent features required for *Ma* are determined using a codec-dependent pre-processor, as proposed in [MX+12]. The different GoP lengths for each frame rate are considered such that the GoPs have a duration of 1 s. This GoP length is typical for AHS deployments, where the video segments usually have an integer length in seconds [Sto11]. The model parameters of *STRM* and *Ma* are determined using least squares non-linear fitting with the training videos of the *Road* and *CIF* video sets and the same spatio-temporal encoding settings as considered for $STRM^+$ (cf., Table 3.1) and $m = 0$. To compare the bit rate estimation performance of all three bit rate models, the PC and RMSE determined with the measured bit rates of an IPP...P GoP ($m = 0$) and an IPBBBBP... GoP ($m = 4$) are listed in Table 3.6 for the *Road* validation video set and in Table 3.7 for the *CIF* validation video set.

The results for an IPP...P GoP show a PC of roughly 0.99 for both considered video sets and for all three rate models which proves the linearity between the model estimations and the actual bit rate values. The RMSE of $STRM^+$ is significantly lower for videos of the *CIF* video set and approximately at the same level for videos of the *Road* video set as compared to *STRM* and *Ma*. The performance gain of $STRM^+$ is high if an IPBBBBP... GoP structure is considered. The RMSE for the *CIF* video set is around 60% lower and for the *Road* video set around 20% lower compared to *STRM* and *Ma*. The main reason for the lower estimation error of $STRM^+$ for the IPBBBBP... GoP structure is that *STRM* and *Ma* are only able to capture the bit rate of IPP...P GoP structures and are not able to account for the influence of B-frames.




(a) $\tau_{TA,SA} = 1\,\text{s}$. (b) $\tau_{TA,SA} = 10\,\text{s}$.

Figure 3.10.: Influence of $\tau_{TA,SA}$ on the bit rate estimation performance for video 3 of the *CIF* video set: RMSE relative to $R_{max,I}$.

#### 3.3.3.3. Influence of the TA and SA window length

Since all model parameters of $STRM^+$ depend on TA and SA, both values have a significant impact on the bit rate estimation performance. In $STRM^+$, the TA and SA values are considered available for each video segment separately. In some video processing systems, however, the TA and SA information might not be available for each video segment individually. Instead, the TA and SA values might be available only for a longer window length, which might cover several video segments. As a consequence, the TA and SA window length ($\tau_{TA,SA}$) does not match the video segment length $\tau$. This might become problematic, especially in longer multimedia domain videos where the content of a video changes significantly over the sequence, such as video 3 of the *CIF* video set (cf., Figure 3.1c). If the TA and SA values are computed for a longer duration rather than for each segment separately, the overall bit rate estimation might become inaccurate. To demonstrate the influence of $\tau_{TA,SA}$ on the bit rate estimation accuracy, the estimation performance of video 3 (*CIF* video set) is determined for $\tau = 1\,\text{s}$ video segments for two different $\tau_{TA,SA}$ values (1 s and 10 s). Figure 3.10 displays the RMSE of the proposed rate model relative to $R_{max,I}$ for all video segments based on the TA and SA values of both window lengths. The results show a RMSE of 3.4% for $\tau_{TA,SA} = 1\,\text{s}$ and a 2 percentage points higher RMSE of 5.4% for $\tau_{TA,SA} = 10\,\text{s}$. For other video sequences with more stable TA and SA values, such as video 1 of the *CIF* video set (Figure 3.3a) or all videos of the *Road* video set, the RMSE is roughly the same for both TA and SA window lengths with a RMSE difference of less than 0.5 percentage points which does not have a large impact in real-life deployments.

## 3.4. Application in perceptual quality-aware rate control

This section applies the developed bit rate model to perceptual quality-aware rate control. In Section 3.4.1 the perceptual quality-aware rate control problem for spatio-temporal encoding

settings is defined and a solution using $STVQM$ and $STRM^{+}$ is developed. In Section 3.4.2, the developed solution of the rate control problem is applied to an AHS video source.

### 3.4.1. Problem definition and solution

In the previous section it has been demonstrated how the bit rate of encoded videos can be modified by an adaptation of the spatial quality, the temporal resolution, and the GoP encoding settings. To achieve a desired bit rate constraint, modern rate controllers for hybrid video coding typically modify the spatial quality of the encoded video by an adaptation of the quantization parameter. To capture the trade-off between the distortion introduced by the quantization and the bit rate of the encoded video, rate-distortion models are employed which consider the distortion measured as PSNR. However, it has been shown that PSNR offers a low correlation with the subjective quality perceived by the viewer [Gir93; GHT08]. In contrast, more recently proposed objective video quality metrics are able to capture the perceptual quality $Q$ (measured in DMOS) more accurately, as demonstrated in Section 2.3.3.3. Furthermore, subjective studies demonstrated that, depending on the video content, the perceptual quality might suffer more from spatial quality reductions rather than a downsampling of the temporal resolution [WMO09]. Based on these two findings, the classical rate control problem of Eq. (2.1) can be reformulated as a perceptual quality-aware rate control optimization problem:

$$(q_j, f_j) = \underset{q,f}{\operatorname{argmax}}\, Q(q, f) \tag{3.11}$$

$$\begin{aligned} \text{subject to} \quad & R(q, f) \leq R_{c,j}, \\ & f_{min} \leq f \leq f_{max}, \\ & q_{min} \leq q \leq q_{max}, \end{aligned}$$

where $q_j$ and $f_j$ are the spatial and temporal encoding settings to maximize the perceptual quality $Q$ for a given rate constraint $R_{c,j}$. Furthermore, $f_{min} \leq f \leq f_{max}$ and $q_{min} \leq q \leq q_{max}$ define the considered frame rate and the quantization parameter encoding settings.

To solve the rate control problem of Eq. (3.11) and to determine the optimal $q$ and $f$ encoding settings for given rate constraints, $STRM^{+}$ of Eq. (3.10) and $STVQM$ of Eq. (2.13) are employed, which both rely on TA and SA values to compute the video content-dependent model parameters. Unlike $STRM^{+}$, which directly depends on the quantization parameter to consider spatial quality modifications, $STVQM$ uses PSNR of the encoded video as a measure of the spatial quality. However, to realize $STQVM$ that directly depends on $q$ for the spatial encoding setting, the PSNR model proposed in [SE+13] (referred to as $PSNR_{[\text{SE+13}]}$ in the following) is applied to the video quality metric of Eq. (2.13). $PSNR_{[\text{SE+13}]}$ makes it possible to estimate the PSNR of encoded videos while using the mean bit rate of the encoded video, the quantization parameter at full frame rate and an IPP...P GoP structure ($m = 0$). The model uses a logarithmic relation between the PSNR value and the bit rate and an approximately

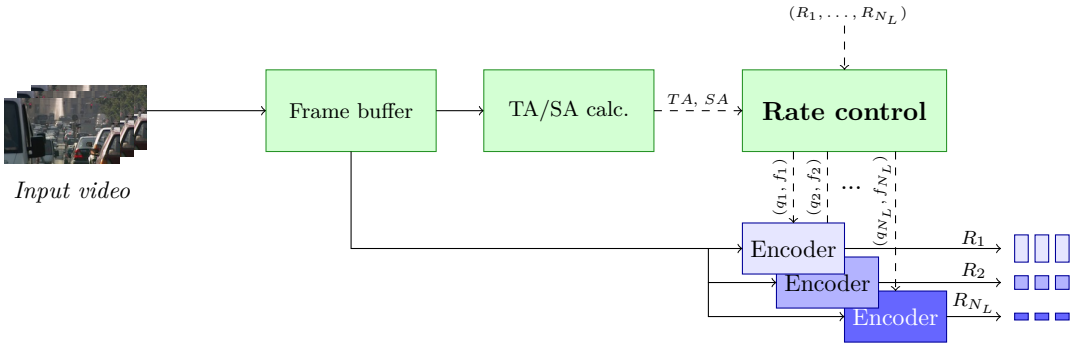



Figure 3.11.: System view of a MBR encoding entity installed at an AHS source with $N_L$ desired bit rates. The rate controller determines optimal encoding settings as solutions to the perceptual quality-aware rate control problem of Eq. (3.11) using TA and SA computed from the source video.

linear relation between the PSNR value and the quantization parameter. Based on these two relations, the PSNR is estimated as:

$$PSNR_{[\text{SE+13}]} = b_1 + b_2 \cdot \log(R) + b_3 \cdot q + b_4 \cdot R \cdot q, \tag{3.12}$$

with a term depending on the bit rate of the encoded video (expressed in kbit/s), a term depending on $q$, an interaction term and a constant offset. Therefore, the estimated PSNR of an encoded video can be computed as

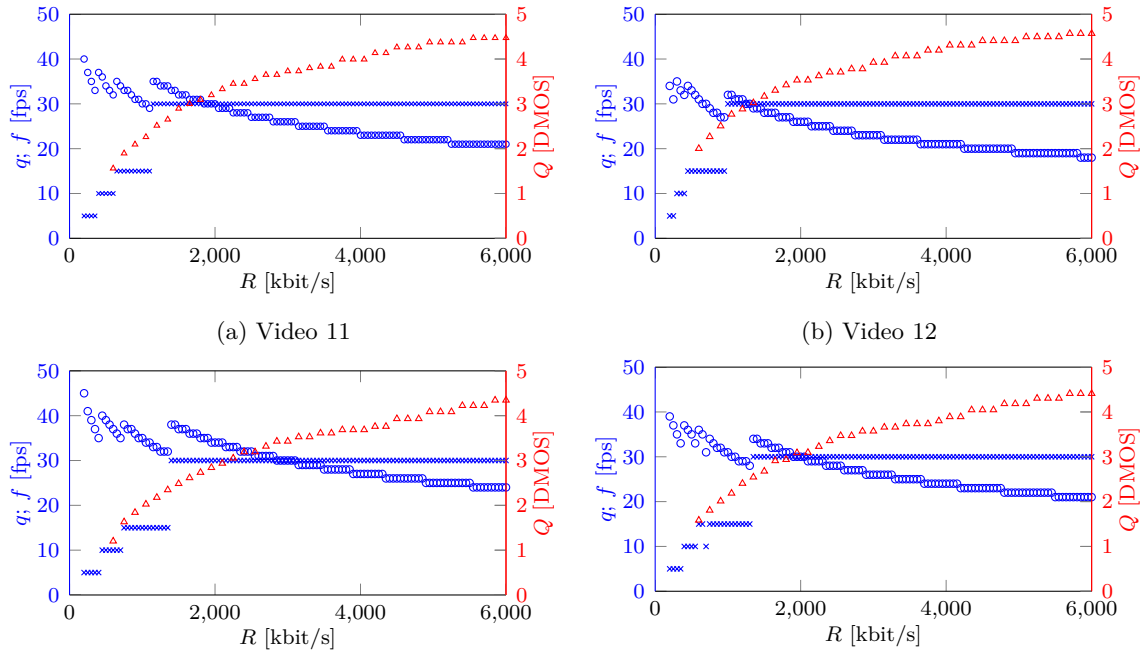
$$\widehat{PSNR} = PSNR_{[\text{SE+13}]}(STRM^{+}(f_{max}, q, n, 0)), \tag{3.13}$$

which is integrated into $STVQM$ as the PSNR value of the encoded video in the following.

For a given rate constraint $R_{c,j}$, the optimization problem of Eq. (3.11) is solved by using exhaustive search to determine the optimal $(q_j,\ f_j)$ values which maximize $Q$. Since typically only a finite number of integer frame rates $(N_t)$ and quantization parameters $(N_q)$ are employed in the encoding process, the computational complexity of the exhaustive search is low, and the optimal solution can be found at a low computation time, i.e., $\mathcal{O}(N_t \cdot N_q)$.

### 3.4.2. Application and performance evaluation

Figure 3.11 displays a system view of a typical application of a rate control entity which employs the proposed rate control problem and its solution. The rate control entity is installed at an AHS source where video segments of a defined duration from a video source need to be encoded at $N_L$ desired bit rates using MBR coding. To this end, the video frames of an uncompressed video segment are first stored in a frame buffer. A further entity computes the TA and SA values of the video segment, which are then used at the rate control entity to compute the model parameters of $STVQM$ and $STRM^{+}$. The determined optimal $(q,\ f)$




(a) Video 11

(b) Video 12

(c) Video 13

(d) Video 14

Figure 3.12.: $q$ (∘) and $f$ (×) determined as solutions of Eq. (3.11) and corresponding $Q$ (△) for given bit rate constraints for videos of the *Road* validation set.

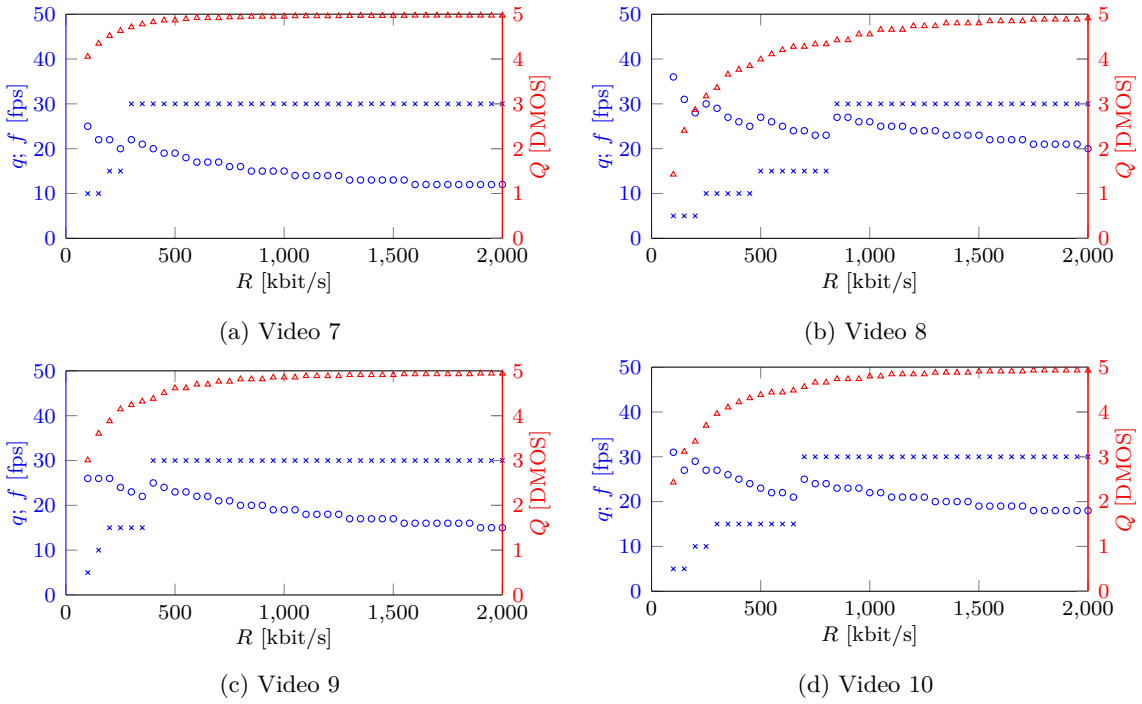
values for the $N_L$ desired bit rate constraints are then used as encoding settings.

In the following it is assumed that all considered video segments offer a duration of $\tau = 1\,\mathrm{s}$ and are encoded with H.264/AVC using the $(q,\ f)$ encoding settings listed in Table 3.1. Since $PSNR_{[\mathrm{SE+13}]}$ considers only IPP...P GoPs, $m$ is set to 0 for the investigation. Figure 3.12 and Figure 3.13 display the determined optimal $(q,\ f)$ encoding settings and the corresponding $Q$ values as a function of the rate constraint $R$ for videos of the *Road* and *CIF* validation set. For larger bit rate constraints, $f$ increases, $q$ reduces continuously, and the perceptual quality $Q$ improves steadily. For all videos, an abrupt decrease of the quantization parameter for increasing rates can be observed which is caused by the coarse integer frame rate steps.

To quantitatively investigate the accuracy of the proposed approach in achieving the target bit rates of the solutions, a set of video bit rate constraints is introduced for both, the *Road* video set ($R_{c,Road} = \{200, 230, 280, 350, 430, 530, 700, 1000, 1700, 2600, 3700, 5000\}\,\mathrm{kbit/s}$) and the *CIF* video set ($R_{c,CIF} = \{100, 125, 175, 235, 285, 375, 475, 600, 850, 1100, 1300, 1500\}\,\mathrm{kbit/s}$). The bit rates for the AHS source[6] are selected according to the guidelines of [TAP+14].

The videos are encoded with the $(q,\ f)$ pairs determined as solutions of Eq. (3.11) for the defined bit rate constraints. To investigate the accuracy in achieving the rate constraints, the RMSE between the bit rate constraints and the measured bit rates of the videos encoded with the determined $(q,\ f)$ pairs relative to $R_{m=0}$ (IPP...P GoP, $q = q_{min}$, $f = f_{max}$, $m = 0$,

[6] A detailed introduction to the selection process of the AHS video levels is given in Section 5.4.1.2.




(a) Video 7

(b) Video 8

(c) Video 9

(d) Video 10

Figure 3.13.: $q$ (∘) and $f$ (×) determined as solutions of Eq. (3.11) and corresponding $Q$ (△) for given bit rate constraints for videos of the *CIF* validation set.

$n = f_{max} \cdot \tau$) is determined. The accuracy of the proposed approach is further compared to solutions based on *STRM* and *Ma*. Table 3.8 lists the RMSE of the solutions of the rate control problem of Eq. (3.11) using all three rate models for the *Road* and *CIF* video sets. For $STRM^+$ and *STRM*, the trained *STVQM* parameters developed in Section 2.3.3.3 are used for the *Road* video set and for the *CIF* video set the parameters of [PS11] are employed. For *Ma*, *VQMTQ* introduced in Section 2.3.3.1 is used as the objective video quality metric. The trained metric parameters for *VQMTQ* are taken from the investigation of Section 2.3.3.3 for the *Road* video set and from [MX+12] for the *CIF* video set.

The determined solution based on $STRM^+$ offers an about 4 percentage points lower RMSE for the *CIF* video set and an around 1 percentage points lower RMSE for the *Road* video set compared to the solution based on *STRM*. The better estimation performance of the solution using $STRM^+$ lies in the more accurate rate estimation than of *STRM*. A similar trend can be obtained for the solution based on *Ma*, which offers an about 6 percentage points worse RMSE for the *CIF* video set and a 3 percentage points worse RMSE for the *Road* video set compared to the $STRM^+$ based solution.

To quantitatively compare the achieved perceptual quality of the different solutions, first the perceptual quality $Q$ is computed for all rate constraints using STVQM of Eq. (2.13) with the $(q, f)$ pairs determined as solutions of Eq. (3.11) based on all three different rate models. In a second step, the mean perceptual quality $\mu_Q$ is computed as the mean over the $Q$ values of all rate constraints and videos of the validation set. The results listed in Table 3.8 show that

| Video set | Perf. metric | *Ma* [MX+12] | *STRM* [LM+14b] | *STRM*$^+$ |
|---|---|---|---|---|
| *Road* | %RMSE | 8.63 | 6.79 | 5.64 |
| | $\mu_{R_c,Q}$ [DMOS] | 3.75 | 3.38 | 3.36 |
| *CIF* | %RMSE | 13.86 | 11.58 | 7.91 |
| | $\mu_{R_c,Q}$ [DMOS] | 4.24 | 4.24 | 4.22 |

Table 3.8.: Mean relative RMSE (normalized by $R_{m=0}$) and mean perceptual quality $\mu_{R_c,Q}$ of the solutions of the rate control problem of Eq. (3.11) for the validation videos of the *Road* and *CIF* video sets using *Ma*, *STRM*, and *STRM*$^+$.

roughly the same mean DMOS is achieved for all three solutions.

## 3.5. Chapter summary

In this chapter, a video bit rate model is developed which considers the impact of the temporal resolution, spatial quality impairments, and GoP characteristics. The rate model is based on the quantization parameter, frame rate, GoP length, and GoP structure encoding settings as well as video content-dependent parameters. TA- and SA-based estimators of the video content-dependent model parameters are developed. Statistical analysis with the measured bit rates of two different video sets show that the proposed model is highly accurate in estimating the bit rate of H.264/AVC encoded videos. The estimation performance is better than or as good as the performance of two other related estimation models, however, with the advantage that GoP characteristics are considered. Finally, the proposed video bit rate model is applied in perceptual quality-aware rate control to determine encoding settings for given rate constraints. The results show that a high accuracy in achieving the bit rate constraints can be achieved while using the proposed video bit rate model.

| Features \ Model param. | $\widehat{R}_{max,I}$ | $s$ | $l_1$ | $l_2$ | $g_1(n)$ | $g_2(n)$ | $g_3(n)$ |
|---|---|---|---|---|---|---|---|
| $\alpha_0$ | -3565.10 | -6.47 | 0.44 | 0.22 | $-1.87 \cdot 10^{-2}$ | 0.12 | -0.25 |
| | | | | | $2.88 \cdot 10^{-4} \cdot n$ | 0 | 0 |
| TA | 159.73 | 0 | 0 | 0 | 0 | 0 | 0 |
| SA | 333.91 | 1.22 | 0 | $-1.75 \cdot 10^{-3}$ | 0 | $4.24 \cdot 10^{-3} \cdot \frac{1}{n}$ | $-1.31 \cdot 10^{-2} \cdot n$ |
| TA · SA | -2.5 | 0 | 0 | 0 | 0 | 0 | 0 |
| log(SA) | 0 | $2.58 \cdot 10^{-3}$ | 0.14 | 0 | 0 | 0 | 0 |
| log(TA) | 0 | 0 | -0.10 | 0 | 0 | 0 | 0 |
| $\frac{\text{TA}}{\text{SA}}$ | 0 | 0 | 0 | 0.49 | 0 | 0 | 0 |
| log(TA·SA) | 0 | 0 | 0 | 0 | $6.60 \cdot 10^{-3} \cdot \frac{1}{n}$ | $-9.68 \cdot 10^{-2} \cdot \frac{1}{n}$ | $0.23 \cdot \frac{1}{n}$ |

Table 3.9.: Coefficients of the TA- and SA-based estimators for the content-dependent model parameters for videos of the *Road* set.

| Features \ Model param. | $\widehat{R}_{max,I}$ | $s$ | $l_1$ | $l_2$ | $g_1(n)$ | $g_2(n)$ | $g_3(n)$ |
|---|---|---|---|---|---|---|---|
| $\alpha_0$ | 868.17 | -2.53 | 0.91 | $-8.20 \cdot 10^{-3}$ | $-8.52 \cdot 10^{-5}$ | $9.78 \cdot 10^{-3}$ | $1.99 \cdot 10^{-2}$ |
| TA | -59.55 | 0 | $-1.09 \cdot 10^{-2}$ | 0 | 0 | 0 | 0 |
| SA | 14.87 | 0 | 0 | $-9.68 \cdot 10^{-4}$ | $3.89 \cdot 10^{-6} \cdot n$ | $9.80 \cdot 10^{-4}$ | $-7.77 \cdot 10^{-4} \cdot \log(n)$ |
| | | | | | $-5.96 \cdot 10^{-5} \cdot \log(n)$ | $-3.70 \cdot 10^{-3} \cdot \frac{1}{n}$ | |
| TA · SA | 1.79 | $2.83 \cdot 10^{-4}$ | 0 | 0 | 0 | 0 | $5.09 \cdot 10^{-7} \cdot n$ |
| log(SA) | 0 | -0.36 | 0 | 0 | 0 | 0 | 0 |
| $\frac{\text{TA}}{\text{SA}}$ | 0 | 0 | 0 | 1.06 | 0 | 0 | 0 |

Table 3.10.: Coefficients of the TA- and SA-based estimators for the content-dependent model parameters for videos of the *CIF* set.

# Chapter 4

# Context-aware estimation of temporal and spatial activities

In this chapter, low-complexity estimators of the temporal and spatial activity values[1] for videos captured with an ADAS front-facing camera of a vehicle are developed by using camera context information. To this end, the estimators comprise camera context features which exploit information about the status and the dynamics of the vehicle and other vehicles in the field-of-view of the front-facing camera. The estimators are applied to a video bit rate model ($STRM^+$), to an objective video quality metric ($STVQM$), and to the solution of the perceptual quality-aware rate control problem previously proposed in Chapter 3 to determine optimal spatio-temporal encoding settings for desired bit rate constraints.

## 4.1. Introduction

Nowadays, vehicles are equipped with a variety of different sensors and camera systems, which are primarily used for on-board ADAS applications. In order to further extend the viewing range of other road users, the video content of the ADAS cameras can be streamed to devices outside the vehicle. To this end, the bit rate of the source video stream needs to be adapted to the transmission capacity of the communication path between the vehicle and the streaming sink.

Rate controllers are required to select the encoding settings to achieve video bit rates according to the current network performance. State-of-the-art rate controllers for hybrid video coding typically employ video bit rate models and objective video quality metrics, which rely on video content-dependent model parameters. The proposed solution to the perceptual quality aware rate control problem developed in Section 3.4.2, for example, employs $STRM^+$ and $STVQM$, which both depend on TA and SA values as video content information. Both,

[1] The TA and SA estimators presented in this chapter have been proposed previously in [LSS].

however, need to be determined from the uncompressed source video. This is problematic in automotive deployments, where the camera modules and video processing ECUs are typically developed as black boxes by automotive suppliers. A direct access to the uncompressed source video streams of ADAS cameras and to the internal functions of the video encoder is typically not possible. Besides that, the automotive suppliers typically employ off-the-shelf hardware encoder modules for the video processing. As a consequence the content-dependent video features cannot be computed along with the video coding process and a separate pre-processor might be required. The calculation of the features at a separate pre-processor, however, is computationally demanding and quickly exceeds the computational capacities of modern ECUs.

For this purpose, camera context-based estimators for the TA and SA values of video sequences recorded with an ADAS front-facing camera have been proposed in [LM+14b], which employ basic context information of the vehicle (mean velocity and scenario of the captured video sequence). While these estimators are well suited for video sequences recorded at a constant velocity and homogeneous traffic conditions, they might offer a low estimation performance for video sequences with diverse movements of the vehicles in the field-of-view or during turnings. To overcome this limitation, this chapter proposes advanced TA and SA estimators which additionally take further dynamics of the vehicle and vehicles in the field-of-view of the ADAS camera into consideration. To this end, a novel set of camera context features is introduced which incorporates advanced information from ADAS sensors and status information of the vehicle. The developed estimators are applied to $STVQM$ and $STRM^{+}$ which both consider spatio-temporal impairments and depend on TA and SA values. Finally, both models are applied to the solution of the rate control optimization problem defined in Eq. (3.11) in order to determine optimal encoding settings which maximize the perceptual quality for given rate constraints. The determined solution based on the developed TA and SA estimators offers similar accuracies in achieving given rate constraints and perceptual quality characteristics as the solution based on computed TA and SA values, however, with the advantage that no access to the uncompressed source video is required.

Camera context-based video processing has been previously considered for motion estimation in hybrid video encoders to simplify the motion estimation process [CZ+11; SS02]. These approaches, however, do not consider the estimation of the spatial information of the source video.

The remainder of the chapter is organized as follows. The camera context features used to estimate the temporal and spatial activities are introduced in Section 4.2. Section 4.3 presents the developed TA and SA estimators which are based on the proposed camera context features. Both estimators are applied to $STRM^{+}$, $STVQM$, and the solution of the rate control optimization problem of Eq. (3.11) in Section 4.4. Finally, Section 4.5 summarizes this chapter.

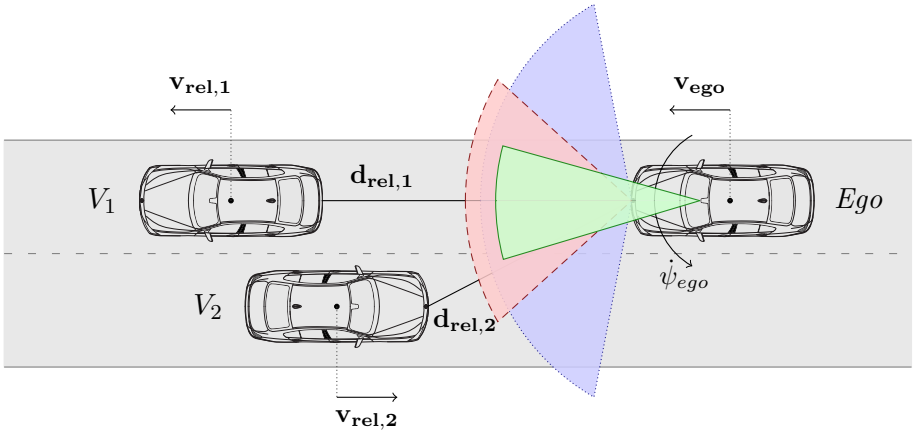



Figure 4.1.: Dynamics of the ego vehicle ($\mathbf{v_{ego}}(t)$, $\dot{\psi}_{ego}(t)$), dynamics of vehicles ($V_i$) in the field-of-view of the ADAS camera ($\mathbf{d_{rel,i}}(t)$, $\mathbf{v_{rel,i}}(t)$) and sensor configuration of the ego vehicle (front-facing LIDAR scanner: , front-facing RADAR: , ADAS front-facing camera: ).

## 4.2. Camera context features

In the following section, camera context features to estimate TA and SA values for videos captured with an ADAS front-facing camera are developed. Information about the dynamics of the vehicle where the ADAS camera is installed (referred to as the *ego vehicle* in the following) and other vehicles in the field-of-view of the ADAS camera are used to describe the context of the captured video.

### 4.2.1. Experimental settings and context information

The on-board sensors of the ego vehicle introduced in Section 2.4.1 are used to determine the status and the dynamics of the ego vehicle, such as the velocity ($\mathbf{v_{ego}}(t) = \begin{bmatrix} v_{x,ego} & v_{y,ego} \end{bmatrix}^{\mathrm{T}}$) and the yaw rate ($\dot{\psi}_{ego}(t)$). Furthermore, the position of the vehicle ($\mathbf{p_{ego}}(t) = \begin{bmatrix} p_{lat,ego} & p_{lon,ego} \end{bmatrix}^{\mathrm{T}}$) is determined by a GPS receiver module. Additional information about the scenario, i.e., the location where the video is recorded and the type of road is determined by matching the GPS position with a digital map, which is performed by the navigation ECU.

In order to gather information about the dynamics and properties of the other objects in the field-of-view of the ADAS camera, the object information provided by a front-facing LIDAR scanner and a front-facing RADAR sensor mounted at the ego vehicle are combined. Figure 4.1 displays the experimental setup of the sensors installed at the ego vehicle. A high-level sensor data fusion architecture proposed in [AK11], which has initially been developed for highly autonomous driving and ADAS applications, is set up at the ADAS ECU and applied to combine the raw sensor data. The fusion architecture considers a three level processing

approach. At the lowest level, *sensor-level* processing is performed, where each sensor creates an abstracted list of detected objects, their state (such as their relative velocity to the ego vehicle), and their existence probability. At the *fusion-level* processing, the object lists are combined to produce a global list of detected objects by the fusion algorithm proposed in [AS+12]. This global list of detected objects is provided to the *application level*, where each application filters the objects and state information based on the relevance for its function.

In the present approach, the global list of objects is filtered in order to create a list of relevant vehicles in the field-of-view of the ADAS front-facing camera ($D_t = \{V_1, V_2, \ldots, V_K\}$). The relative distance between the ego vehicle and vehicle $i$ ($\mathbf{d_{rel,i}}(t) = \begin{bmatrix} d_{x,i} & d_{y,i} \end{bmatrix}^{\mathrm{T}}$) as well as the relative velocity to the $i$th vehicle ($\mathbf{v_{rel,i}}(t) = \begin{bmatrix} v_{x,i} & v_{y,i} \end{bmatrix}^{\mathrm{T}}$) is used to describe the state of the $i$th vehicle at time $t$. It is assumed for each video sequence that $S$ samples of the raw sensor data are available for the computation of the features.

### 4.2.2. Temporal activity related features

In the following, the TA-related context features are introduced. Recall that the TA value of a video sequence (defined in Eq. (2.4)) quantifies the amount of temporal change in a video sequence. To this end, camera context parameters are required which describe the dynamics of the ego vehicle, the dynamics of vehicles in the field-of-view of the ADAS front-facing camera, and the scenario where the video sequence is recorded.

**Mean ego velocity $\nu$:** The velocity of the ego vehicle has a direct influence on the temporal activity of the video content. Hence, the mean velocity over $S$ samples of the ego vehicle, similar to [LM+14b], is defined as

$$\nu = \frac{1}{S} \cdot \sum_{t=1}^{S} \left| \mathbf{v_{ego}}(t) \right| , \tag{4.1}$$

where $|\mathbf{v_{ego}}(t)|$ is the absolute value of the ego vehicle's velocity $\mathbf{v_{ego}}(t)$ at time $t$.

**Mean yaw rate $\omega$:** Sequences recorded while turning offer a fast change of the video content in successive frames. This applies in particular for video sequences recorded in urban scenarios where the temporal change among the frames is low for situations while driving straight and high while turning. To take the rotary dynamics of the ego vehicle into consideration, the absolute mean yaw rate over time is introduced as

$$\omega = \frac{1}{S} \cdot \sum_{t=1}^{S} \left| \dot{\psi}_{ego}(t) \right| , \tag{4.2}$$

where $\left|\dot{\psi}_{ego}(t)\right|$ is the absolute value of the yaw rate of the ego vehicle at time $t$.

**Mean number of detected vehicles $\beta$:** Depending on the traffic situation of the recorded video, a different number of vehicles is visible in the field-of-view of the ADAS front-facing camera. The mean number of detected vehicles in the field-of-view is defined as

$$\beta = \frac{1}{S} \cdot \sum_{t=1}^{S} \|D_t\| \,, \tag{4.3}$$

where $\|D_t\|$ is the number of detected vehicles a time $t$.

**Mean relative velocity of detected vehicles $\lambda$:** The vehicles in the field-of-view of the ADAS front-facing camera are moving at different relative velocities $\mathbf{v_{rel,i}}(t)$ to the ego vehicle. Depending on the distance to the ego vehicle, vehicles in the field-of-view of the ADAS front-facing camera have a different influence on the temporal activity of the video sequence. Vehicles with a small distance to the ego vehicle appear large, whereas vehicles with a large distance to the ego vehicle appear small. Therefore, for $K$ vehicles in the field-of-view, $\lambda$ is defined as

$$\lambda = \frac{1}{S} \cdot \frac{1}{K} \cdot \sum_{t=1}^{S} \sum_{i=1}^{K} \left|\mathbf{v_{rel,i}}(t)\right| \cdot \left|\mathbf{d_{rel,i}}(t)\right|^{-1} \,, \tag{4.4}$$

where $\left|\mathbf{v_{rel,i}}(t)\right|$ is the absolute value of the relative velocity and $\left|\mathbf{d_{rel,i}}(t)\right|^{-1}$ the inverse of the absolute value of the distance between the ego vehicle and the $i$th vehicle at time $t$ in the field-of-view.

**Scenario $\zeta$:** In order to realize a clear distinction between the different types of scenes where the video is recorded, the categorical variable $\zeta$ is introduced. It has been shown in a pre-study that a binary categorical variable is sufficient to describe the different scenes and to cover the typical driving scenes. Similar as in [LM+14b], a value of 0 expresses highway scenarios and 1 urban scenarios.

### 4.2.3. Spatial activity related features

In the following, the SA-related camera context features are introduced. Recall that SA values indicate the amount of spatial detail of video sequences (cf., Eq. (2.5)). To this end, SA-related features are defined which employ information about the status of the ego vehicle and the other vehicles in the field-of-view of the ADAS camera.

**Mean inverse distance to detected vehicles $\delta$:** Depending on the distance to the ego vehicle, vehicles in the field-of-view of the ADAS camera have a different influence on the spatial

activity of the video sequence. Similar as for $\lambda$, the further the other vehicles are apart from the ego vehicle, the less influence they have on the spatial activity. To gather the influence of the vehicles, the mean absolute inverse distance to the $K$ detected vehicles in a video sequence is defined as

$$\delta = \frac{1}{S} \cdot \frac{1}{K} \cdot \sum_{t=1}^{S} \sum_{i=1}^{K} \left|\mathbf{d_{rel,i}}(t)\right|^{-1}, \tag{4.5}$$

where $\left|\mathbf{d_{rel,i}}(t)\right|^{-1}$ is the inverse of the absolute value of the distance between the ego vehicle and the $i$th vehicle at time $t$.

**Yaw standard deviation $\alpha$:** Turnings of the ego vehicle have a significant influence on the spatial activity measure, since the spatial information might change significantly over the sequence. This holds especially for sequences in urban scenarios while turning at corners, where the amount of spatial detail is low while driving straight and high while turning. Hence, the standard deviation of the angle of yaw over time is used to capture the influence of turnings on the spatial activity. It is defined as

$$\alpha = \sigma_S(\psi_{ego}(t)), \tag{4.6}$$

which is the standard deviation of the angle of yaw relative to the start position ($\psi_{ego}(t)$) over all $S$ samples.

Since $\beta$ and $\zeta$ are also related to the spatial details of the video frames, they are additionally considered in the SA feature set in the course of this chapter.

## 4.3. Estimation of temporal and spatial activity values

In the following section, the camera context-based TA and SA estimators using the aforementioned camera context features are developed. For the investigation, the *Road* video set introduced in Section 2.3.3.2 is considered. Analogous to the investigations in Chapter 3.3.1, the video pool is separated into a *training set* (video 1-10), which is used for the training of the estimators, and a separate *validation set* (video 11-14) which is used to investigate the estimation performance of the developed estimators for videos outside the training set.

In order to develop the TA and SA estimators, the iterative GLM-based feature selection approach with an analysis on the CVE introduced in Section 3.3.2.2 is applied to combine and select a minimal set of camera context features. Therefore, the TA estimator (referred to as $\widehat{TA}$ in the following) and the SA estimator (referred to as $\widehat{SA}$ in the following) for $L$ camera context-based features are developed as

$$\widehat{TA}, \widehat{SA} = \sum_{i=1}^{L} \alpha_i \cdot f(x_i) + \alpha_0, \tag{4.7}$$

| Video ID | TA | SA | $\nu$ | $\omega$ | $\beta$ | $\lambda$ | $\sigma$ | $\delta$ | $\alpha$ |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 26.28 | 57.28 | 5.97 | 10.02 | 1.75 | 0.37 | 1 | 0.17 | 31.70 |
| 2 | 13.28 | 49.80 | 7.01 | 0.41 | 3.41 | 0.33 | 1 | 0.13 | 0.68 |
| 3 | 22.37 | 50.72 | 5.54 | 9.17 | 1.13 | 0.22 | 1 | 0.13 | 33.15 |
| 4 | 6.57 | 41.74 | 2.86 | 0.31 | 1.75 | 0.23 | 1 | 0.14 | 0.49 |
| 5 | 6.75 | 23.68 | 32.07 | 0.29 | 4.36 | 0.09 | 0 | 0.12 | 0.69 |
| 6 | 6.74 | 28.36 | 28.69 | 0.32 | 7.30 | 0.05 | 0 | 0.22 | 1.02 |
| 7 | 2.37 | 27.82 | 5.58 | 0.25 | 2.63 | 0.06 | 0 | 0.17 | 0.45 |
| 8 | 4.09 | 29.48 | 9.97 | 0.35 | 4.60 | 0.08 | 0 | 0.15 | 0.20 |
| 9 | 5.39 | 61.53 | 0.00 | 0.03 | 2.53 | 0.17 | 1 | 0.15 | 0.01 |
| 10 | 18.92 | 48.56 | 5.78 | 8.85 | 1.46 | 0.14 | 1 | 0.06 | 23.14 |
| 11 | 12.45 | 53.06 | 5.48 | 3.33 | 1.84 | 0.26 | 1 | 0.12 | 11.15 |
| 12 | 9.98 | 30.34 | 35.77 | 0.21 | 4.45 | 0.17 | 0 | 0.14 | 0.25 |
| 13 | 23.19 | 61.23 | 12.78 | 1.77 | 2.81 | 0.61 | 1 | 0.17 | 3.52 |
| 14 | 14.57 | 46.93 | 8.54 | 2.25 | 1.98 | 0.38 | 1 | 0.10 | 4.19 |

Table 4.1.: Computed TA, SA and feature values for all videos of the *Road* training and validation set.

| #Iteration | Parameters | PC | CVE |
|---|---|---|---|
| 1 | $\zeta$ | 0.642 | 54.58 |
| | $\beta$ | 0.526 | 82.44 |
| | $\nu$ | 0.243 | 79.65 |
| | $\omega$ | 0.935 | 12.15 |
| | $\lambda$ | 0.689 | 51.04 |
| 2 | $\omega$, $\lambda$ | 0.980 | 5.30 |
| 3 | $\omega$, $\lambda$, $\nu$ | 0.993 | 3.44 |
| 4 | $\omega$, $\lambda$, $\nu$, $\beta$ | 0.994 | 3.14 |

Table 4.2.: TA cross-validation results for the videos of the *Road* training set.

where the $\alpha_i$ are determined by least squares non-linear fitting. In the implementation, the sensor information of the ego vehicle and the vehicles in the field-of-view is provided at a frequency of 10 Hz which leads to an overall number of samples per video sequence of $S = 100$. The computed values of all features for all videos are listed in Table 4.1. For the investigation, $\widehat{TA}$ and $\widehat{SA}$ are computed for the full video sequence, i.e., $\tau_{TA,SA} = 10\,\text{s}$.

### 4.3.1. Temporal activity estimator development

$\widehat{TA}$ is developed by applying the GLM with the temporal activity related features. The videos of the training set are used in order to evaluate the dependency of the TA value and the camera context-based features. Figures 4.2 (a)-(e) show the scatter plots of the single camera context-based features $\zeta$, $\beta$, $\nu$, $\omega$, and $\lambda$ versus the computed TA values. Table 4.2 lists the corresponding estimation performance in terms of the CVE and PC of the different single features. The results reveal that the single feature $\omega$ offers the lowest CVE among all single features with a PC of almost 0.94, which is the highest estimation performance of the single features.

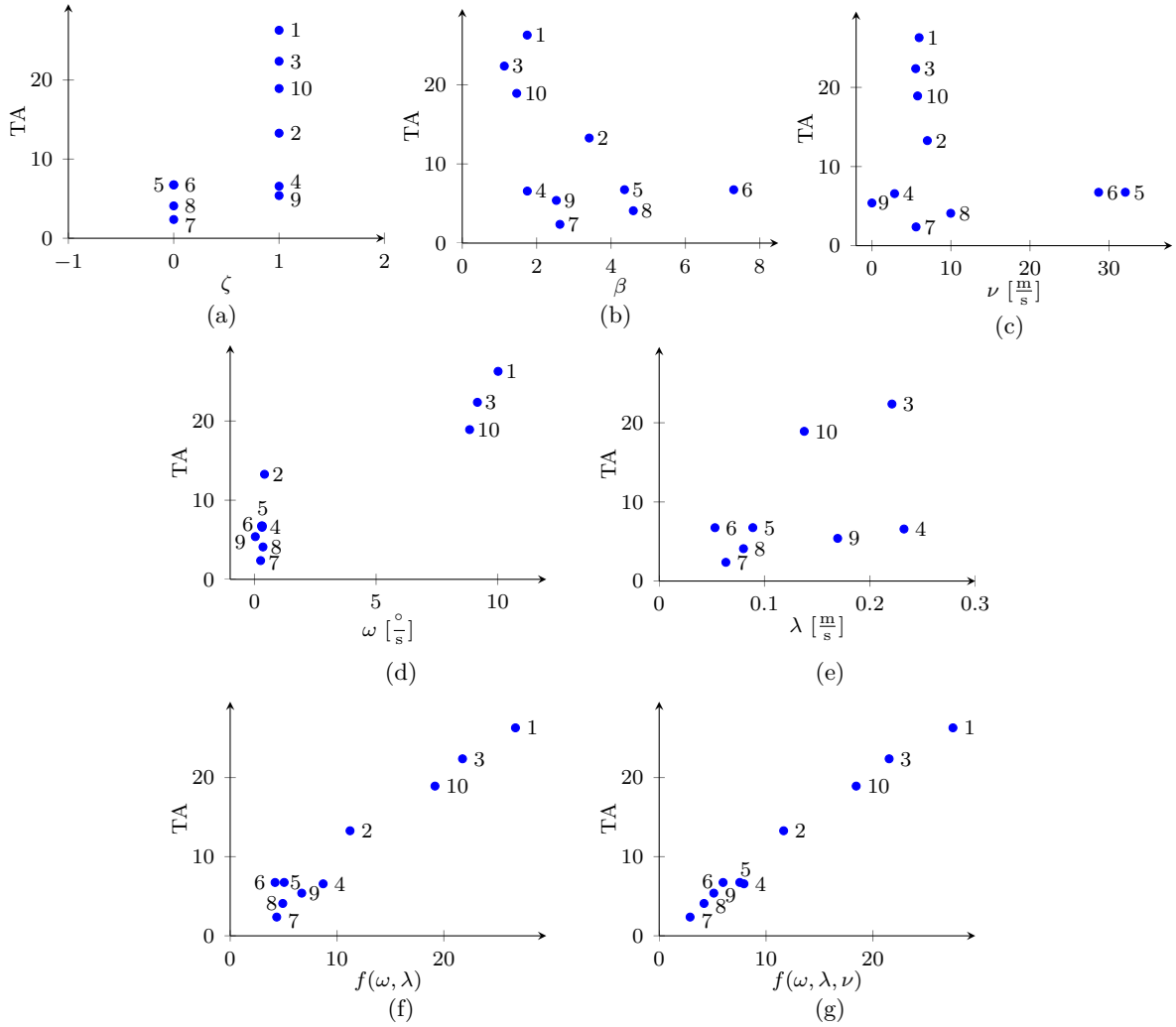



Figure 4.2.: Computed TA values versus temporal activity related features and feature combinations for videos of the *Road* training set.

To further improve the estimation performance, the iterative GLM with an analysis of the CVE is applied. Table 4.2 displays the PC and CVE of the feature combinations of the different iterations. Furthermore, Figures 4.2 (f)-(g) display the computed versus the estimated TA values. The results show that a linear combination of the four camera context-based features $\omega$, $\lambda$, $\beta$, and $\nu$ at the fourth iteration offers the lowest CVE of all feature combinations, which cannot be further reduced by adding more features to the feature set. Hence, the resulting camera context-based TA estimator is

$$\widehat{TA} = t_1 \cdot \omega + t_2 \cdot \lambda + t_3 \cdot \nu + t_4 \cdot \beta + t_5, \tag{4.8}$$

where the values for the model parameters trained with videos of the training set using least squares non-linear fitting are $t_1 = 1.26\ \frac{\mathrm{s}}{^\circ}$, $t_2 = 28.46\ \frac{\mathrm{s}}{\mathrm{m}}$, $t_3 = 1.51\ \frac{\mathrm{s}}{\mathrm{m}}$, $t_4 = 4.60 \cdot 10^{-3}$, and $t_5 = -0.66$.

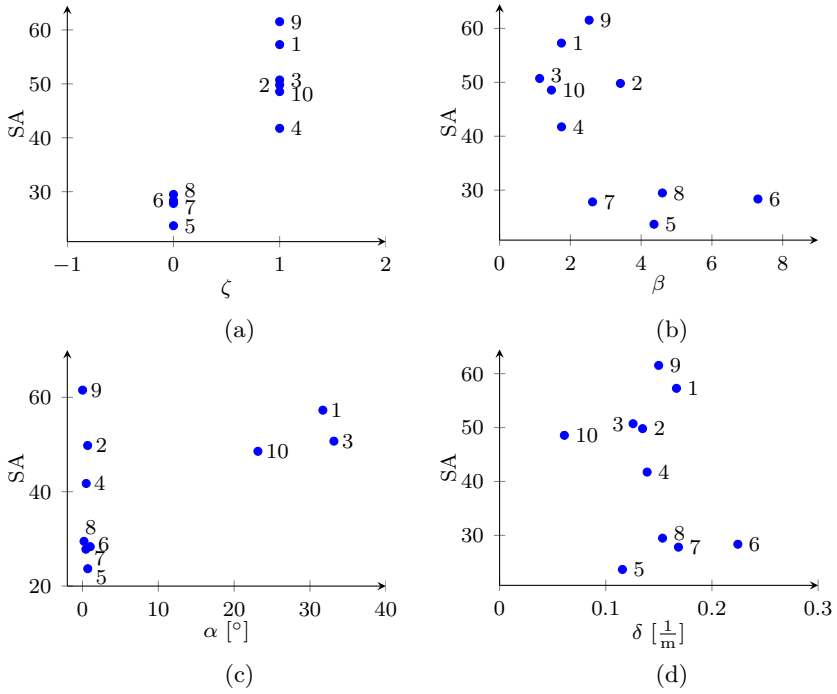



Figure 4.3.: Computed SA values versus spatial activity related features for the videos of the *Road* training set.

| #Iteration | Parameters | PC | CVE |
|---|---|---|---|
| 1 | $\zeta$ | 0.919 | 38.17 |
| | $\mu$ | 0.642 | 151.58 |
| | $\alpha$ | 0.520 | 170.04 |
| | $\delta$ | 0.264 | 216.82 |
| 2 | $\zeta$, $\delta$ | 0.933 | 35.49 |

Table 4.3.: SA cross-validation results for the videos of the *Road* training set.

### 4.3.2. Spatial activity estimator development

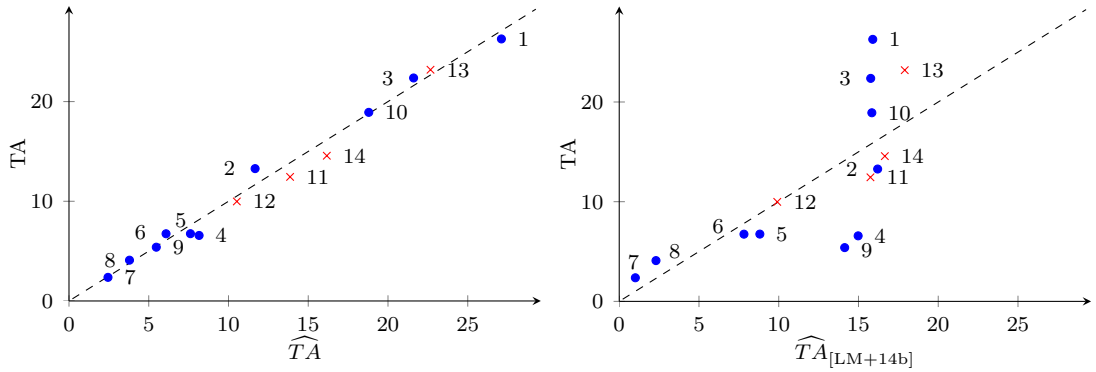
Analogous to $\widehat{TA}$, $\widehat{SA}$ is developed by applying the iterative GLM process with the spatial activity related features introduced in Section 4.2.3. Figure 4.3 displays the computed SA values versus the single features for videos of the training set. Furthermore, Table 4.3 lists the estimation performance in terms of CVE and PC for the different features and feature combinations. At the first iteration of the GLM, $\zeta$ offers the lowest GLM of 38.17. The CVE is further reduced down to 35.49 at the second iteration by additionally adding $\delta$ to the feature set. The GLM is terminated at the second iteration step since no further reductions could be achieved by adding other features to the feature set. Hence, the resulting SA estimator is

$$\widehat{SA} = s_1 \cdot \zeta + s_2 \cdot \delta + s_3, \tag{4.9}$$

where $s_1 = 26.37$, $s_2 = 58.14\text{m}$, and $s_3 = 17.71$.

| Perf. metric | Training set | Validation set |
|---|---|---|
| PC | 0.994 | 0.991 |
| %RMSE | 3.33 | 4.31 |

Table 4.4.: $\widehat{TA}$ estimation performance: PC and RMSE relative to the maximum TA value of all videos of the *Road* video set.




(a) $\widehat{TA}$ estimation. (b) $\widehat{TA}_{[LM+14b]}$ estimation.

Figure 4.4.: $\widehat{TA}$ and $\widehat{TA}_{[LM+14b]}$ estimation performance for videos of the *Road* training (•) and validation set (×).

### 4.3.3. Performance evaluation

#### 4.3.3.1. $\widehat{TA}$ performance evaluation

In the following, the estimation performance of $\widehat{TA}$ is assessed and compared to the previously proposed TA estimator of [LM+14b], and the computational complexity of $\widehat{TA}$ is determined.

**$\widehat{TA}$ estimation performance:** In order to assess the estimation performance of the developed TA estimator, the PC and RMSE of $\widehat{TA}$ of Eq. (4.8) versus the computed TA values are determined for the videos of the training and the validation video set. For consistency, the RMSE is normalized by the maximum TA value of all videos. The results listed in Table 4.4 indicate a high estimation performance with a PC of 0.99 as well as a RMSE of less than 4% for videos of the training set. A similar estimation performance for the validation set with a PC of more than 0.99 and a RMSE of less than 5% verifies the robustness of the model for videos outside the training set.

**$\widehat{TA}$ performance comparison:** The estimation performance of $\widehat{TA}$ is compared with the TA estimator proposed in [LM+14b] (referred to as $\widehat{TA}_{[LM+14b]}$ in the following). In comparison to $\widehat{TA}$, $\widehat{TA}_{[LM+14b]}$ uses limited camera context information ($\zeta$ and $\nu$). Analogous to $\widehat{TA}$, the model parameters of $\widehat{TA}_{[LM+14b]}$ are trained with the videos of the training set using least squares non-linear fitting. A graphical representation of the estimation performance

of both estimators for all videos is given in Figure 4.4. The performance comparison shows that the estimation performance of $\widehat{TA}$ is significantly better as compared to $\widehat{TA}_{\text{[LM+14b]}}$ ($PC_{\widehat{TA},\text{[LM+14b]}} = 0.765$, $\text{RMSE}_{\widehat{TA},\text{[LM+14b]}} = 14.18\%$ for videos of the validation set). The main reason is that $\widehat{TA}_{\text{[LM+14b]}}$ does not consider the yaw rate of the ego vehicle ($\omega$) and the dynamics of the other vehicles in the field-of-view ($\lambda$, $\beta$, and $\nu$). This limits the model to sequences with no turnings and homogeneous driving conditions of other vehicles in the field-of-view of the ADAS front-facing camera.

**Computational complexity of $\widehat{TA}$:** The computational complexities of TA (Eq. (2.4)) and $\widehat{TA}$ (Eq. (4.8)) are determined and compared. The computational complexity of TA, which depends on the frame size, is $\mathcal{O}_{\text{TA}}(N_x \cdot N_y)$ with $N_x$ horizontal and $N_y$ vertical pixels. It is assumed for $\widehat{TA}$ that the context features are provided by the ADAS ECU and do not need to be computed separately. As a consequence, the computational complexity of $\widehat{TA}$ is independent of the actual video content and can be seen as constant ($\mathcal{O}_{\widehat{TA}}(1)$). Similar as for $\widehat{TA}_{\text{[LM+14b]}}$, the computational complexity of $\widehat{TA}$ is significantly lower as compared to TA.

#### 4.3.3.2. $\widehat{SA}$ performance evaluation

In the following, the estimation performance of $\widehat{SA}$ is determined, a performance comparison to the SA estimator of [LM+14b] is conducted, and the computational complexity of $\widehat{SA}$ is determined.

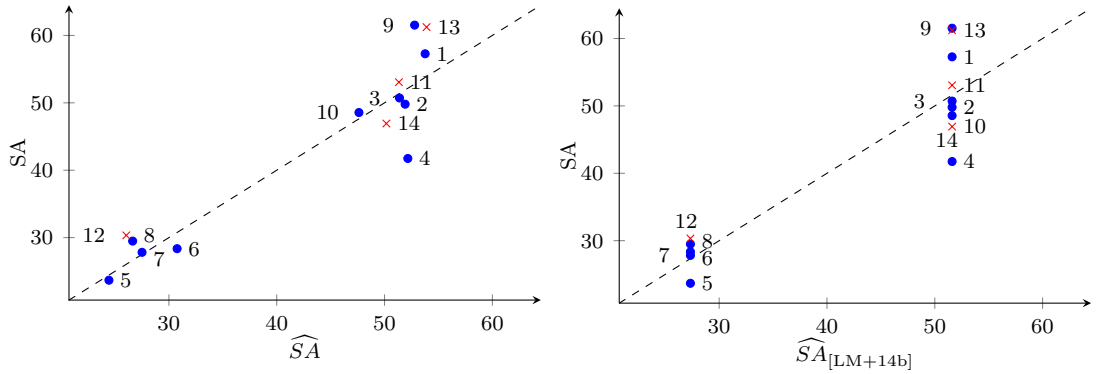
**$\widehat{SA}$ estimation performance:** Similar as for the $\widehat{TA}$ performance evaluation, the PC and RMSE between $\widehat{SA}$ and the computed SA values are determined for videos of the training and validation set. For consistency, the RMSE is normalized by the maximum computed SA value of all videos. Table 4.5 lists the results, which indicate an overall high estimation performance with a PC of 0.93 and a RMSE of less than 8% for videos of the training set, and even a marginally better estimation performance for videos of the validation set.

**$\widehat{SA}$ performance comparison:** $\widehat{SA}$ is compared to the SA estimator proposed in [LM+14b] (referred to as $\widehat{SA}_{\text{[LM+14b]}}$ in the following). In contrast to $\widehat{SA}$, $\widehat{SA}_{\text{[LM+14b]}}$ is solely based on $\zeta$. Figure 4.5 displays the measured SA values versus the estimated SA values for both estimators. The estimation performance of $\widehat{SA}$ is higher compared to $\widehat{SA}_{\text{[LM+14b]}}$ ($PC_{\widehat{SA},\text{[LM+14b]}} = 0.894$, $RMSE_{\widehat{SA},\text{[LM+14b]}} = 9.15\%$ for the videos of the validation set), which is due to the dependency of $\widehat{SA}_{\text{[LM+14b]}}$ on a single feature.

**Computational complexity of $\widehat{SA}$:** Finally, the computational complexities of SA (Eq. (2.5)) and $\widehat{SA}$ (Eq. (4.9)) are determined and compared. The computational complexity of SA,

| Perf. metric | Training set | Validation set |
|---|---|---|
| PC | 0.933 | 0.941 |
| %RMSE | 7.58 | 7.56 |

Table 4.5.: $\widehat{SA}$ estimation performance: PC and RMSE relative to the maximum SA value of all videos of the *Road* video set.




(a) $\widehat{SA}$ estimation. (b) $\widehat{SA}_{[LM+14b]}$ estimation.

Figure 4.5.: $\widehat{SA}$ and $\widehat{SA}_{[LM+14b]}$ estimation performance for videos of the *Road* training (•) and validation set (×).

which depends on the settings of the Sobel filter[2], is $\mathcal{O}_{\mathrm{SA}}(N_x \cdot N_y \cdot k^2)$. On the other hand, the asymptotic time complexity of $\widehat{SA}$ is independent of the actual video content assuming that, similar as for $\widehat{TA}$, the context features are provided by the ADAS ECU and do not need to be computed additionally. Hence, similar as for $\widehat{SA}_{[LM+14b]}$, the computational complexity can be seen as constant, i.e., $\mathcal{O}_{\widehat{SA}}(1)$, which is significantly lower as compared to the computational complexity of SA.

In this section, estimators for the temporal and spatial activity values of video sequences recorded with an ADAS front-facing camera of a vehicle based on context information of the vehicle are developed. For the proposed low-complexity TA and SA estimators, a high estimation performance could be achieved with the additional advantage that no access to the raw video stream is required. In comparison to the TA and SA estimators proposed in [LM+14b], which only consider limited camera-context features, the developed TA and SA estimators improve the estimation performance significantly.

## 4.4. Application in perceptual quality-aware video rate control

In this section, $\widehat{TA}$ and $\widehat{SA}$ are applied to *STVQM* and to $STRM^+$. Both consider spatio-temporal encoding settings and employ TA and SA values as video content information.

[2]Within the thesis a search range of $k = 3$ is considered, which is the suggested setting proposed by the ITU for the SI determination [IR08].

| VQM | Perf. metric | Training set | Validation set |
|---|---|---|---|
| *STVQM* | PC | 0.984 | 0.991 |
| | RMSE [DMOS] | 0.25 | 0.17 |
| *CSTVQM* | PC | 0.983 | 0.982 |
| | RMSE [DMOS] | 0.237 | 0.275 |
| $CSTVQM_{\text{[LM+14b]}}$ | PC | 0.981 | 0.981 |
| | RMSE [DMOS] | 0.29 | 0.25 |

Table 4.6.: VQM estimation performance: PC and absolute RMSE values for videos of the *Road* training and validation set.

Analogous to the investigation in Section 3.4, both models are used to determine a solution to the perceptual quality-aware rate control problem of Eq. (3.11).

### 4.4.1. Spatio-temporal video quality metric

In the following, $\widehat{TA}$ and $\widehat{SA}$ are integrated into *STVQM* of Eq. (2.13) and the performance in estimating the perceptual quality for different encoding settings is compared to *STVQM*, which is based on the computed TA and SA values and *STVQM* which employs $\widehat{TA}_{\text{[LM+14b]}}$ and $\widehat{SA}_{\text{[LM+14b]}}$.

For this purpose, *CSTVQM* is introduced as the camera context-aware *STVQM*, which depends on $\widehat{TA}$ and $\widehat{SA}$:

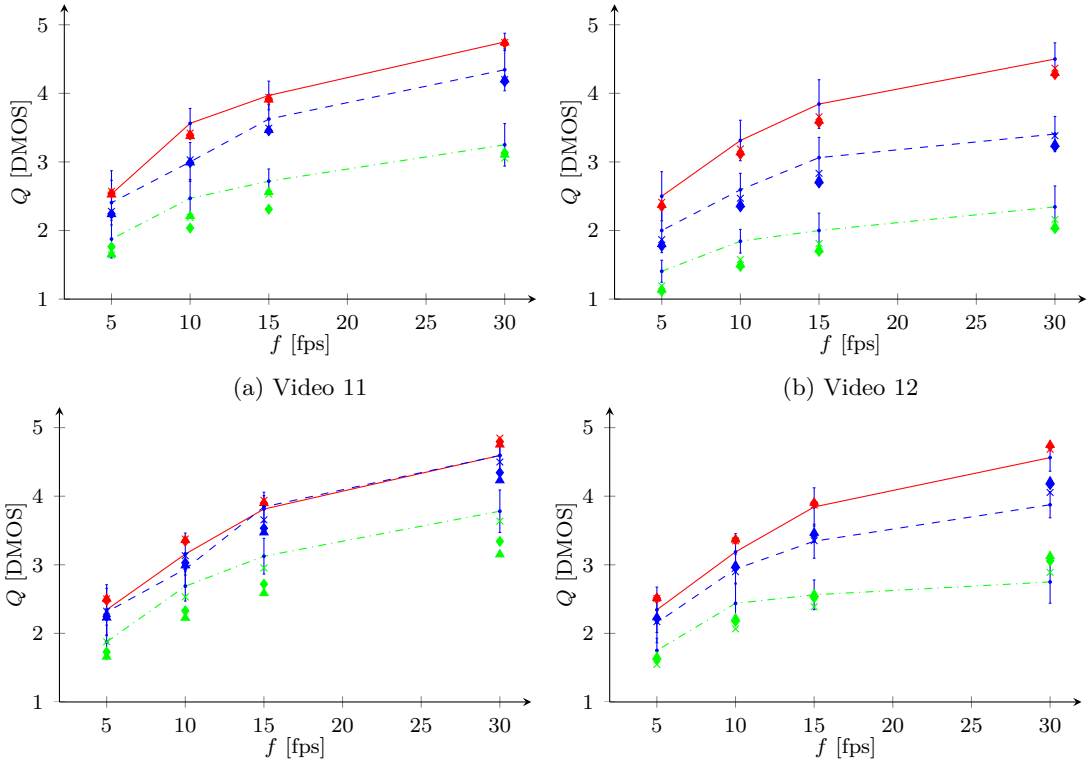
$$Q_{CSTVQM} = Q_{STVQM}(\widehat{TA}, \widehat{SA}). \tag{4.10}$$

Furthermore, $CSTVQM_{\text{[LM+14b]}}$ is defined as *STVQM*, which depends on $\widehat{TA}_{\text{[LM+14b]}}$ and $\widehat{SA}_{\text{[LM+14b]}}$:

$$Q_{CSTVQM_{\text{[LM+14b]}}} = Q_{STVQM}(\widehat{TA}_{\text{[LM+14b]}}, \widehat{SA}_{\text{[LM+14b]}}). \tag{4.11}$$

The estimation performance of *STVQM*, *CSTVQM*, and $CSTVQM_{\text{[LM+14b]}}$ is determined for the full video sequences (i.e., $\tau = 10\,\text{s}$) of the *Road* video set using the $Q$ values determined from the subjective quality assessment of Section 2.3.3.2. Table 4.6 lists the estimation performance in terms of PC and absolute RMSE for the evaluation of the videos from the training and the validation set. Furthermore, Figure 4.6 displays a graphical representation of the DMOS values of the videos from the validation set along with the 95% CI and the model estimations of *STVQM* using the three different TA and SA estimators.

The results for the training set show that the estimation performance of *STVQM*, *CSTVQM*, and $CSTVQM_{\text{[LM+14b]}}$ is comparable. For the validation set, a slightly worse RMSE of 0.1 DMOS for *CSTVQM* and $CSTVQM_{\text{[LM+14b]}}$ compared to *STVQM* can be observed. The results confirm the findings of [LM+14b], that some inaccuracies in the TA and SA estimation are tolerable in *STVQM* without a significant degradation of the perceptual quality estimation performance.




(a) Video 11 (b) Video 12

(c) Video 13 (d) Video 14

Figure 4.6.: Performance evaluation of $STVQM$ (×), $CSTVQM$ (◇), $CSTVQM_{[\mathrm{LM+14b}]}$ (△) for videos of the validation set; measured $Q$ obtained from the subjective test of Section 2.3.3.2 for 42 dB (——), 38 dB (- - -), 34 dB (- · - ·) with 95% CI (——).

### 4.4.2. Video bit rate model

Similar as for the objective video quality metric, $\widehat{TA}$ and $\widehat{SA}$ are integrated into $STRM^+$ of Eq. (3.10). The bit rate estimation performance is further compared to $STRM^+$ based on the computed TA and SA values, and $STRM^+$ based on $\widehat{TA}_{[\mathrm{LM+14b}]}$ and $\widehat{SA}_{[\mathrm{LM+14b}]}$.

For this purpose, $CSTRM^+$ is defined as $STRM^+$ depending on $\widehat{TA}$ and $\widehat{SA}$:

$$R_{CSTRM^+} = R_{STRM^+}(\widehat{TA}, \widehat{SA}). \tag{4.12}$$

Besides that, $CSTRM^+_{[\mathrm{LM+14b}]}$ is introduced as the $\widehat{TA}_{[\mathrm{LM+14b}]}$ and $\widehat{SA}_{[\mathrm{LM+14b}]}$ dependent $STRM^+$:

$$R_{CSTRM^+_{[\mathrm{LM+14b}]}} = R_{STRM^+}(\widehat{TA}_{[\mathrm{LM+14b}]}, \widehat{SA}_{[\mathrm{LM+14b}]}). \tag{4.13}$$

For the investigation, the trained model parameters developed in Section 3.3.2.2 are applied to the $\tau = 1\,\mathrm{s}$ segments of the videos of the *Road* video set. To assess and compare the estimation performance of $STRM^+$, $CSTRM^+$, and $CSTRM^+_{[\mathrm{LM+14b}]}$, the PC and RMSE relative to $R_{max,I}$ are determined with the measured bit rate values for the same encoding settings as listed in Table 3.1. The results summarized in Table 4.7 show that $CSTRM^+$ offers a slightly worse estimation performance as compared to $STRM^+$ with an about 0.5 percentage points

| Model | Perf. metric | Training set | Validation set |
|---|---|---|---|
| $STRM^{+}$ | PC | 0.975 | 0.966 |
| | %RMSE | 3.72 | 3.84 |
| $CSTRM^{+}$ | PC | 0.968 | 0.970 |
| | %RMSE | 4.26 | 4.20 |
| $CSTRM^{+}_{[\text{LM+14b}]}$ | PC | 0.969 | 0.963 |
| | %RMSE | 4.95 | 4.84 |

Table 4.7.: Video bit rate estimation performance: PC and RMSE relative to $R_{max,I}$.

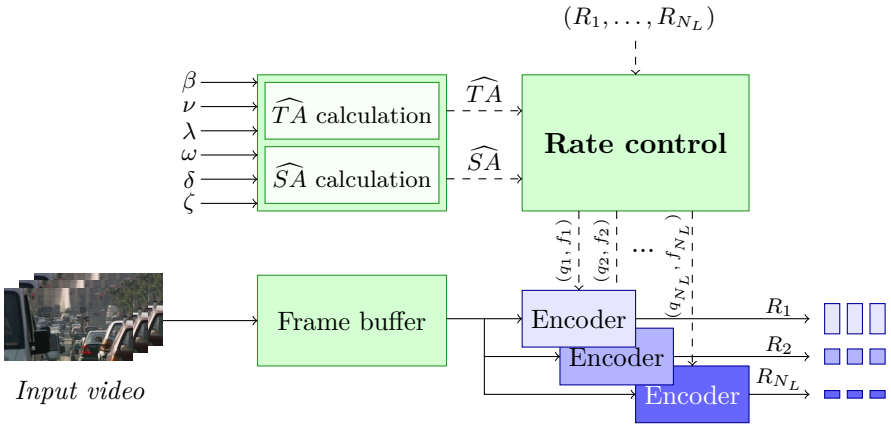



Figure 4.7.: System view of a MBR encoding entity installed at an AHS source with $N_L$ desired video bit rates. The rate controller determines optimal encoding settings as solutions to the rate control optimization problem of Eq. (3.11) using $\widehat{TA}$ and $\widehat{SA}$.

higher RMSE for the videos of both the training and the validation set. In all cases, $CSTRM^{+}$ offers a slightly better estimation performance compared to $CSTRM^{+}_{[\text{LM+14b}]}$ with an about 0.7 percentage points lower RMSE considering both the training and the validation set.

The results of the performance evaluation show that $CSTRM^{+}$ (based on $\widehat{TA}$ and $\widehat{SA}$) offers a similar bit rate estimation performance as compared to $STRM^{+}$. Besides that, the bit rate estimation performance can be improved compared to $CSTRM^{+}_{[\text{LM+14b}]}$ at the same computational complexity.

### 4.4.3. Perceptual quality-aware video rate control

Figure 4.7 displays a system view of a MBR encoding entity installed at an AHS source where the segments of a defined duration of a source video need to be encoded at different desired bit rates. Compared to the AHS source system introduced in Section 3.4, which employs a pre-processor to compute the TA and SA values directly from the uncompressed source video, the rate control entity of Figure 4.7 depends on a camera context-based pre-processor to compute $\widehat{TA}$ and $\widehat{SA}$. To this end, the rate control entity of Figure 4.7 determines the optimal $(q,\ f)$ values for the $N_L$ desired rate constraints based on $\widehat{TA}$ and $\widehat{SA}$.

| Perf. metric | TA, SA | $\widehat{TA}$, $\widehat{SA}$ | $\widehat{TA}_{\text{[LM+14b]}}$, $\widehat{SA}_{\text{[LM+14b]}}$ |
|---|---|---|---|
| %RMSE | 5.64 | 8.25 | 10.95 |
| $\mu_{R_c,Q}$ [DMOS] | 3.36 | 3.38 | 3.35 |

Table 4.8.: Mean relative RMSE (normalized with $R_{m=0}$) and mean perceptual quality $\mu_{R_c,Q}$ of the solutions of the rate control problem of Eq. (3.11) for the videos of the *Road* validation set.

*CSTVQM* and $CSTRM^+$ are applied to the solution of the perceptual quality-aware rate control problem of Eq. (3.11) to determine optimal spatio-temporal encoding settings for given rate constraints. In order to realize *CSTVQM* which directly depends on $q$ for the spatial encoding settings the $\widehat{PSNR}$ model of Eq. (3.13) is integrated into *CSTVQM*. Analogous to the approach of Section 3.4.1, an IPP...P GoP is considered (i.e., $m = 0$). Exhaustive search is applied to determine the optimal solutions for given rate constraints. It is assumed that all PVSs are encoded with H.264/AVC with the ($q$, $f$) ranges defined in Table 3.1.

The accuracy of the approach in achieving desired target bit rates using $\widehat{TA}$ and $\widehat{SA}$ is assessed and compared to the solutions which employ the TA and SA estimators of [LM+14b] and the computed TA and SA values. The defined AHS bit rate constraints for the *Road* video set defined in Section 3.4.1 are used for the performance assessment. Table 4.8 shows the RMSE between the bit rate constraints and the measured bit rates of the videos encoded with the parameters determined as solutions of Eq. (3.11) relative to $R_{m=0}$ using the three different TA and SA estimators for videos of the validation set. The solution based on the proposed $\widehat{TA}$ and $\widehat{SA}$ offers an about 2.5 percentage points higher RMSE in achieving the bit rate constraints as the solution based on computed TA and SA values. In comparison, the solution which employs $\widehat{TA}_{\text{[LM+14b]}}$ and $\widehat{SA}_{\text{[LM+14b]}}$ offers a RMSE of about 5.3 percentage points higher compared to the computed TA- and SA-based solution. The better estimation performance of the solution using $\widehat{TA}$ and $\widehat{SA}$ lies in the more accurate rate estimation of $CSTRM^+$ compared to $CSTRM^+_{\text{[LM+14b]}}$.

Furthermore, the perceptual quality (in DMOS) for the ($q$, $f$) pairs determined as solutions of Eq. (3.11) for all rate constraints using STVQM of Eq. (2.13) is computed. Similar as in Section 3.4.2, first the perceptual quality $Q$ is computed for all rate constraints using STVQM of Eq. (2.13) with the ($q$, $f$) pairs determined as solutions of Eq. (3.11) based on all three TA and SA estimators. In a second step, the mean perceptual quality $\mu_{R_c,Q}$ is computed as the mean over the $Q$ values of all rate constraints and videos of the validation set. The results listed in Table 4.8 show that roughly the same $\mu_{R_c,Q}$ is achieved for all three TA and SA estimators.

Since the developed TA and SA estimators depend on features which are determined independently of the raw video, the presented rate control entity can be applied in automotive scenarios to encode the source video captured with an ADAS front-facing camera where an access to the source video stream or to internal functions of video encoders might not be possible.

## 4.5. Chapter summary

In this chapter, low-complexity TA and SA estimators are developed for video sequences captured with an ADAS front-facing camera of a vehicle based on camera context information of the vehicle. Using the proposed TA and SA estimators, a high estimation performance for both estimators with the computed TA and SA values is achieved with the additional advantage that no access to the raw video stream is required. In comparison to the previously proposed TA and SA estimators of [LM+14b], which only consider limited camera-context features, the developed TA and SA estimators are able to improve the estimation performance significantly at the same computational complexity. The developed TA and SA estimators are applied to a video bit rate model ($STRM^{+}$), an objective video quality metric ($STVQM$), and the solution of the perceptual quality-aware rate control problem to determine the optimal spatio-temporal encoding settings for desired bit rate constraints. The results show that, using the proposed estimators, a similar estimation performance compared to the solution based on computed TA and SA values is achieved for the bit rate model, the video quality metric, and the solution of the rate control optimization problem.

# Chapter 5

# Dynamic video level encoding for uplink adaptive HTTP streaming

This chapter studies the uplink delivery of live video content using AHS from mobile video sources. Three context-aware video level selection algorithms[1] are developed which use different context information to select a subset of video levels from a full static video level set. By using the proposed algorithms, the number of video levels at the AHS source can be reduced significantly. The developed algorithms are applied in an automotive scenario where the video of a vehicle's ADAS front-facing camera is upstreamed to a video portal deployed in the Internet.

## 5.1. Introduction

Video streaming to and from mobile devices over RANs is challenging due to the time-varying network performance and frequent inter-RAN handovers. Therefore, adaptive streaming technologies are required in order to avoid an empty video playout buffer at the streaming client caused by a mismatch of the playout and transmission bit rate. RTP/UDP-based adaptive streaming systems offer intra-session rate adaptation mechanisms, however, are often filtered out by firewalls [Sto11]. AHS-based systems, on the contrary, which have originally been developed for adaptive downlink streaming from CDNs, use HTTP/TCP-based transmissions. Since most of the network components deployed in the Internet are designed to support HTTP/TCP-based traffic, AHS-based streaming systems do not suffer from firewall filterings. So far, AHS-based systems have not been considered for uplink streaming from mobile devices since the MBR encoding process of creating the different video levels is computationally demanding and quickly exceeds the computational resources of modern mobile devices. The computational capacities and the number of hardware encoders installed at mobile devices

[1] Parts of this chapter appeared in preliminary form in [LG+15].

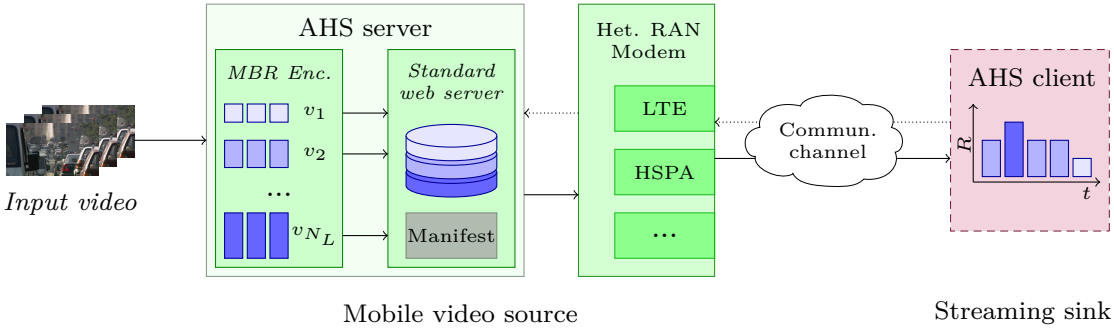



Figure 5.1.: System image of the considered uplink AHS architecture. The AHS server (marked in green) is installed on the mobile video source. The AHS client, which uses standard AHS adaptation algorithms, is deployed at the streaming sink (marked in purple).

are typically limited and as a consequence such devices might not be able to create the same number of video levels in real time as CDN systems, which typically employ 10-15 video levels [App; TAP+14]. However, due to its favorable deployment and transport characteristics, AHS offers major advantages compared to RTP/UDP-based streaming systems for video uplink streaming from mobile devices.

To this end, this chapter considers an uplink video streaming scenario, in which the video captured with a camera of a mobile device is upstreamed to a single remote sink using AHS. The streaming sink, for example, can be another mobile device or a video portal, which can act as an intermediate node to offer further live consumption or the storage of the video content for on-demand retrieval. Figure 5.1 displays the system model of the considered AHS streaming architecture. An AHS server entity, deployed at a mobile device, splits the video into segments of a defined duration and encodes the segments at different desired bit rates using MBR. The desired bit rates are defined in a full static video level set and used as control information for the MBR encoding entity. The encoded video segments are stored on a locally installed web server along with a manifest file which is used to inform the client about the different video segments. An AHS client entity is installed at the streaming sink, which adaptively requests the video segments from the AHS server set up at the mobile video source using standard AHS adaptation algorithms. The communication channel between the mobile source and the streaming sink consists of a cascade of the uplink RAN used by the mobile video source, a WAN infrastructure, and a wired or wireless network technology to the streaming sink. It is assumed that the mobile device uses a heterogeneous RAN modem which offers support for different cellular communication standards.

In order to reduce the number of video levels considered for the AHS-based uplink streaming, this chapter proposes three different context-aware algorithms, which dynamically select a reduced set of video levels from the full static video level set. The reduced video level set is used as the control information at the MBR encoding entity. The first two algorithms are network

performance-aware, since they employ different statistics of the TCP throughput performance to select the reduced video level set. The third algorithm employs the history of the previously requested video levels of the streaming client to determine the reduced video level set. All three algorithms are applied to a mobile uplink streaming scenario, where the video of a vehicle's ADAS front-facing camera is upstreamed to a video portal. In the experimental evaluation, TCP network performance traces from measurements in real HSPA and LTE networks are used. The results show that with the proposed video level selection algorithms, a similar user experience can be achieved as compared to a reference implementation which employs the full static set of video levels. At the same time, the number of video levels considered in the streaming session can be reduced significantly using all three proposed algorithms.

The remainder of this chapter is organized as follows. Section 5.2 reviews the related work on AHS rate adaption algorithms and the context-aware selection of AHS video levels. Section 5.3 introduces the concept of context-aware dynamic video level selection and proposes the three algorithms to determine a reduced set of video levels out of a full static video level set. In Section 5.4 the experimental evaluation of the three algorithms in an automotive environment is conducted. Finally, Section 5.5 summarizes this chapter.

## 5.2. Related work

The overall behavior and user experience in a streaming session of AHS systems depend, besides the performance of the communication channel between the streaming source and sink, to a large degree on the system settings and mechanisms used both at the AHS server and the AHS client. In the following, major AHS client rate adaptation algorithms are reviewed and an overview of the advances in the area of video level selection in AHS systems is given.

Since AHS is a pull-based streaming system, rate adaptation algorithms deployed at the streaming sink are used in order to react to network performance fluctuations and to select the most appropriate video levels. The adaptation algorithms can be classified into three categories regarding their input control information:

- **Throughput-based adaptation algorithms:** Network performance measures directly derived from the TCP throughput of the previously fetched video segments are used as the control input for the successive video level selection decisions. The adaptation algorithm proposed by Liu et al. [LBG11] employs the segment fetch time for the video level selection, which describes the duration from the request until the successful transmission of the segment to the client. A smooth playback is guaranteed if the segment fetch time is less than or equal to the segment duration. Based on this requirement, the authors use the ratio between the segment duration and the segment fetch time to decide on the video level for the successive segment. In their work, the authors consider a sequential segment fetch, i.e., the segments are requested and received sequentially

one at a time. They extended their work in [LB+12a], where they also consider parallel fetching of video segments using multiple parallel TCP connections. Despite their simplicity, both approaches offer some limitations inherited from the employed TCP network performance information. Huang et al. [HH+12] demonstrated that inaccurate TCP performance estimations might introduce an oscillating feedback loop which in turn leads to frequent quality switches in streaming sessions and an overall poor user experience. This occurs presumably in scenarios where various AHS clients share one network performance bottleneck. To overcome this limitation, Li et al. [LZ+14] proposed a *probe-and-adapt* mechanism which performs additional probes at the application layer to reliably quantify the network throughput and thus to prevent oscillation effects. Similar to that, Mok et al. [ML+12] proposed a quality-of-experience (QoE)-aware AHS system which additionally employs bandwidth probes at the application layer.

- **Buffer status-based adaptation algorithms:** The algorithms solely employ information about the buffer status in the rate adaptation process. The adaptation algorithm proposed by Huang et al. [HJM13] uses the fullness of the video playback buffer of the streaming client to decide on the video levels of the successive video segments, which makes the potentially error-prone process of TCP network performance estimation superfluous. A control-theoretical approach using a proportional-derivative controller was proposed by Zhou et al. [ZL+14], which uses the buffer fullness information of the client for the video level selection. Besides that, De Cicco et al. [DCMP11] proposed a further control-theoretical approach which employs the sender buffer in the rate adaptation process.

- **Buffer- and throughput-based adaptation algorithms:** The adaptation algorithms use information about the buffer status as well as the TCP throughput information for the rate adaptation process. The algorithm proposed by Miller et al. [MQ+12] targets to accomplish an optimal playout buffer level throughout the streaming session by reacting to the buffer status as well as to the measured TCP throughput of the last segment fetches. The authors additionally employed a *fast-start* mechanism to increase the bit rates of the video levels rapidly at the beginning of a streaming session in order to achieve a high user experience directly after the start-up. Similar to that, the algorithm proposed by Tian et al. [TL13] determines the video levels of the successive video segments based on the video time buffered at the client, the recent TCP throughput history, and the video levels of the previously transmitted segments. In order to avoid buffer underflows at all times, the rate adaptation algorithm considers a throughput safety margin and uses a conservative quantization process to select the video levels of the requested segments.

The set of video levels employed in an AHS-based streaming system has a major influence on the overall user satisfaction in a streaming session. To this end, the video levels need to be selected thoroughly prior to the setup of the AHS system taking the contents and

deployment characteristics, such as the expected throughput range of the streaming clients as well as the capabilities of the user devices (screen resolution, processing capacities, etc.), into consideration. Most commercial AHS systems, such as Microsoft's *Smooth Streaming* [Zam09], Adobe's *HTTP Dynamic Streaming* [Ado10], or Apple's *HTTP Live Streaming* [PM14; App] recommend a static set of video levels for different spatial resolutions. Toni et al. [TAP+14] demonstrated that these recommendation sets have major weaknesses in terms of the overall user satisfaction in streaming sessions. To overcome these issues, they defined and solved an optimization problem to select a set of optimal video levels with respect to the user experience. Based on the optimal solution, they proposed guidelines to select the video levels in real AHS deployments.

## 5.3. Dynamic video level selection

This section first defines the objective of dynamic video level selection (Section 5.3.1). It proposes the three context-aware algorithms to select a subset of video levels out of a full static video level set based on (i) network performance information (Section 5.3.2) and (ii) the history of the previous video level requests of the clients (Section 5.3.3).

### 5.3.1. Video level selection objective

The goal of dynamic video level selection is to choose a reduced set of video levels with $N_R$ elements ($\tilde{\mathbf{V}} = \{\tilde{v}_1, \tilde{v}_2, ..., \tilde{v}_{N_R}\}$) from a full static video level set with $N_L$ elements ($\mathbf{V} = \{v_1, v_2, ..., v_{N_L}\}$), which is used as control information at the AHS server to produce the video segments employed in the streaming session. Ideally, the user experience in streaming sessions using $\tilde{\mathbf{V}}$ should be the same as in streaming sessions which use the full static video level set $\mathbf{V}$.

In order to quantify the user experience of a streaming session, four performance measures are introduced, which are determined at the streaming client:

- $\gamma$ is defined as the total duration of buffer emptiness due to stalling events in a streaming session which results in an interrupted playback. A large duration of interrupted playback leads to a low user satisfaction [SE+14].
- $\epsilon$ is defined as the number of video level changes during a streaming session. A low number of switches during a session leads to a high user satisfaction [NE+11]. $\epsilon$ is determined as

$$\epsilon = \sum_{i=0}^{I} a(v_i), \qquad a(v_i) = \begin{cases} 0 & v_{i-1} = v_i \\ 1 & v_{i-1} \neq v_i,\ i = 0 \end{cases}$$

where $v_i$ is the video level of the $i$th fetched segment, and $I$ is the total number of fetched segments in the streaming session.

- $\mu_R$ defines the mean bit rate of the transmitted video levels in a streaming session and is computed by
$$\mu_R = \frac{1}{I} \cdot \sum_{i=1}^{I} v_i.$$
- $\mu_Q$ is defined as the mean perceptual quality of the transmitted video segments of a streaming session and is determined by
$$\mu_Q = \frac{1}{I} \cdot \sum_{i=1}^{I} Q(v_i),$$
where $Q(v_i)$ is the perceptual quality of the $i$ the video level (measured in DMOS).

Based on these definitions, the goal of the dynamic video level selection can be formulated as

$$N_R = \underset{\tilde{\mathbf{V}} \subseteq \mathbf{V}}{\operatorname{argmin}} \left\| \tilde{\mathbf{V}} \right\| \tag{5.1}$$

$$\text{subject to} \quad \begin{aligned} \gamma_{\mathbf{V}} &= \gamma_{\tilde{\mathbf{V}}}, \\ \epsilon_{\mathbf{V}} &= \epsilon_{\tilde{\mathbf{V}}}, \\ \mu_{Q,\mathbf{V}} &= \mu_{Q,\tilde{\mathbf{V}}}, \end{aligned}$$

where $\left\| \tilde{\mathbf{V}} \right\|$ describes the number of video levels of $\tilde{\mathbf{V}}$, which is a subset of $\mathbf{V}$. Besides that, $\gamma_{\mathbf{V}}$, $\epsilon_{\mathbf{V}}$, and $\mu_{Q,\mathbf{V}}$ denote the user experience of a streaming session if the full static video level set $\mathbf{V}$ is considered. $\gamma_{\tilde{\mathbf{V}}}$, $\epsilon_{\tilde{\mathbf{V}}}$, and $\mu_{Q,\tilde{\mathbf{V}}}$ quantify the user experience if the reduced video level set $\tilde{\mathbf{V}}$ is employed.

### 5.3.2. Network performance-aware dynamic video level selection

In the following subsection, first the system model employed for the two network performance-aware dynamic video level selection algorithms is presented, followed by an introduction of the two algorithms.

#### 5.3.2.1. System model

To determine $\tilde{\mathbf{V}}$, the network performance-aware dynamic video level selection algorithms employ two different sources for the TCP network performance information $T$: $T_M$ is the TCP uplink throughput performance of the streaming session for a window length of $W$ seconds, measured by the RAN modem installed at the mobile device. $T_{DB}$ is the TCP uplink throughput information provided by a remote lookup database. It is assumed that the throughput information stored in the remote database is created through previous TCP

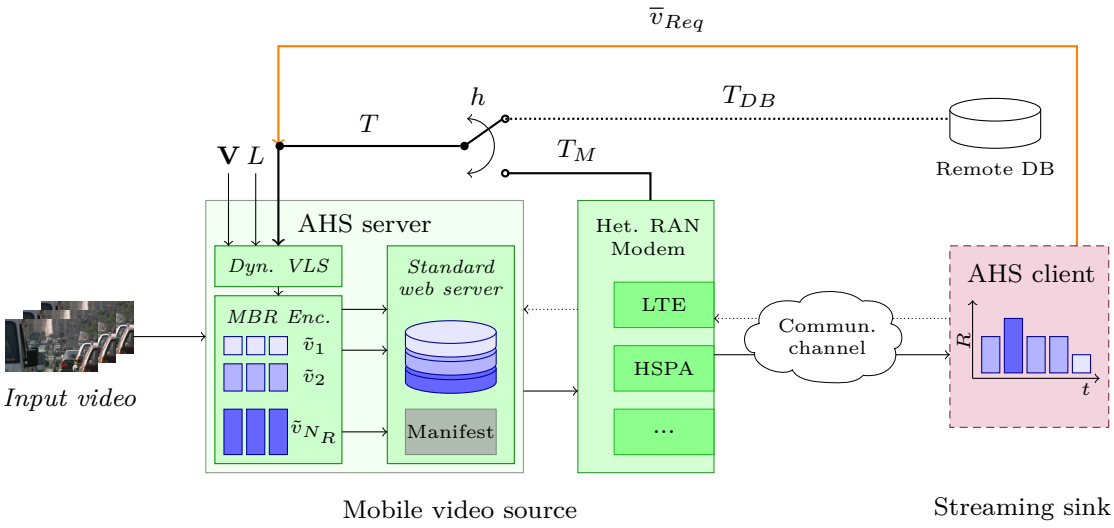



Figure 5.2.: System model of the dynamic video level selection approaches (referred to as *Dyn. VLS*): network performance-aware dynamic video level selection employs measured and requested TCP network performance information $T$ (——), client request-aware dynamic video level selection uses history of client requests $\overline{v}_{Req}$ (——).

throughput measurements of other mobile devices with the same network context[2]. Besides that, it is considered that the mobile devices also provide the measured uplink TCP throughput information to the remote database during or after the streaming session in order to keep the system up-to-date. The variable $h$ is introduced to select the source of the TCP throughput performance information ($h = 1$: $T = T_{DB}$, $h = 0$: $T = T_M$).

Two network performance-aware dynamic video level selection algorithms are developed to produce $\tilde{\mathbf{V}}$ from the static video level set $\mathbf{V}$ based on $T$. The *measured network performance-aware dynamic video level selection algorithm* (referred to as *NetVLS-M* in the following) employs $T_M$ as the primary source for the TCP network performance information and $T_{DB}$ during the start-up phase and after inter-RAN handovers. The *requested network performance-aware dynamic video level selection algorithm* (referred to as *NetVLS-DB* in the following) uses solely $T_{DB}$ as the TCP network performance information. The system model of the network performance-aware dynamic video level selection is displayed in Figure 5.2.

[2]A *network context* of a connection denotes the connected RAN, the geographical area of a connection using the same base station, and the time of the day.

**Algorithm 1:** NetVLS-M/NetVLS-DB

**Input**: $\mathbf{V}$: Full set of available video levels
$N_R$: Number of video levels considered in $\tilde{\mathbf{V}}$
$T_M$: TCP uplink throughput measured at the mobile device
$T_{DB}$: TCP uplink throughput requested from remote database
$h$: Variable to select the source of $T$

**Output**: $\tilde{\mathbf{V}}$: Reduced set of video levels

**if** $h = 1$ **then**
  $T = T_{DB}$
**else**
  $T = T_M$
$v_n = \underset{\{v_i, 1 \leq i \leq N_L\}}{\arg\min} |v_i - T|$
**if** $n < \lfloor {}^{N_R}/_{2} \rfloor + 1$ **then**
  $\tilde{\mathbf{V}} \leftarrow \{v_1, v_2, \cdots, v_{N_R}\}$
**else if** $n > N_L - \lceil {}^{N_R}/_{2} - 1 \rceil$ **then**
  $\tilde{\mathbf{V}} \leftarrow \{v_{N_L - L + 1}, v_{N_L - N_R + 2}, \cdots, v_{N_L}\}$
**else**
  **if** $N_R$ *is odd* **then**
    $\tilde{\mathbf{V}} \leftarrow \{v_{n - \lfloor {}^{N_R}/_{2} \rfloor}, v_{n - \lfloor {}^{N_R}/_{2} \rfloor + 1}, \cdots, v_{n + \lfloor {}^{N_R}/_{2} \rfloor}\}$
  **else**
    $\tilde{\mathbf{V}} \leftarrow \{v_{n - \frac{N_R}{2}}, v_{n - \frac{N_R}{2} + 1}, \cdots, v_{n + \frac{N_R}{2} - 1}\}$
**return** $\tilde{\mathbf{V}}$

#### 5.3.2.2. Measured network performance-aware dynamic video level selection

Algorithm 1 displays the algorithm's pseudocode. It is assumed that the algorithm is called after the request of a segment when one of the two following events occurs: (i) the streaming session is started or an inter-RAN handover between two RANs occurs, (ii) a duration of $W$ seconds of the streaming session has elapsed. The algorithm needs to be re-invoked after an inter-RAN handover since the two RANs involved in the handover might feature significantly different TCP network performance properties. It is assumed that the manifest file, which contains references to the video segments of $\tilde{\mathbf{V}}$, is updated after the algorithm is invoked.

The algorithm requires the input arguments $\mathbf{V}$, $N_R$, and $T$ and determines $\tilde{\mathbf{V}}$ as the output argument. In order to reduce the number of video levels, the algorithm requires $N_R < N_L$.

After the algorithm is called, first the TCP uplink throughput information is determined. After inter-RAN handovers and the start-up, $T$ is set to $T_{DB}$ (i.e., $h = 1$). At all other times, $T$ is set to $T_M$ (i.e., $h = 0$). In the successive step, the video levels of the reduced set $\tilde{\mathbf{V}}$ are

determined according to $T$ using a symmetrical windowing approach. To this end, first, the video level $v_n$ is determined as the center element of $\tilde{\mathbf{V}}$, which offers the smallest absolute difference to $T$ among all other video levels in $\mathbf{V}$ (line 5). The remaining $N_R - 1$ video levels are selected symmetrically around $v_n$ (line 6-14). In the case of an odd $N_R$, a perfectly symmetrical selection of the remaining $N_R - 1$ video levels is performed by choosing the same number of adjacent video levels above and below $v_n$ (line 12). Otherwise, a conservative selection approach is conducted and $\frac{N_R}{2}$ adjacent video levels below and $\frac{N_R}{2} - 1$ adjacent video levels above $v_n$ are selected (line 14). Depending on $N_R$, if not enough video levels are available below or above $v_n$, the algorithm constructs $\tilde{\mathbf{V}}$ asymmetrically around $v_n$. In this case, the algorithm selects the available video levels above (or below, respectively) of $v_n$ and the remaining video levels from below (or above, respectively) (line 6-9).

#### 5.3.2.3. Requested network performance-aware dynamic video level selection

The application of the previously proposed *NetVLS-M* makes it possible to determine the video levels specifically for the actual network connection performance due to the primary usage of $T_M$. However, in order to determine $T_M$, a measurement interface at the RAN modem is required. This might be problematic in automotive deployments, where a full access to the ECUs and thus to the TCP network performance information is typically not available. To overcome this limitation, *NetVLS-DB* is proposed which considers $T_{DB}$ as the only source of the TCP throughput information, and, hence, does not require any additional interfaces at the RAN modem of the mobile video source.

*NetVLS-DB* (cf., Algorithm 1) follows a similar approach as *NetVLS-M* with the difference that $h = 1$ at all times, since only $T_{DB}$ is employed as the TCP throughput information. The selection process of $\tilde{\mathbf{V}}$ is performed symmetrically around $T$ and follows the same symmetrical windowing approach as *NetVLS-M* (line 6-14).

### 5.3.3. Client request-aware dynamic video level selection

The following subsection first presents the system model of the client request-aware dynamic video level selection, followed by an introduction to the corresponding video level selection algorithm (referred to as *CliVLS* in the following).

#### 5.3.3.1. System model

The system model of the client request-aware dynamic video selection approach, displayed in Figure 5.2, considers a similar system model as the network performance-aware dynamic video level selection approach with the difference that no TCP throughput performance information is required to generate the reduced video level set.

**Algorithm 2:** CliVLS

**Input**: $\mathbf{V}$ : Full set of available video levels

$N_R$: Number of video levels considered in $\tilde{\mathbf{V}}$

**Output**: $\tilde{\mathbf{V}}$: Reduced set of video levels

$v_n = \underset{\{v_i, 1 \leq i \leq N_L\}}{\arg\min} |v_i - \overline{v}_{\text{Req}}|$

**if** $n < \lfloor {}^{N_R}/_{2} \rfloor + 1$ **then**

  $\tilde{\mathbf{V}} \leftarrow \{v_1, v_2, \cdots, v_{N_R}\}$

**else if** $n > N_L - \lceil {}^{N_R}/_{2} - 1 \rceil$ **then**

  $\tilde{\mathbf{V}} \leftarrow \{v_{N_L - N_R + 1}, v_{N_L - N_R + 2}, \cdots, v_{N_L}\}$

**else**

  **if** $N_R$ *is odd* **then**

    $\tilde{\mathbf{V}} \leftarrow \{v_{n - \lfloor {}^{N_R}/_{2} \rfloor}, v_{n - \lfloor {}^{N_R}/_{2} \rfloor + 1}, \cdots, v_{n + \lfloor {}^{N_R}/_{2} \rfloor}\}$

  **else**

    $\tilde{\mathbf{V}} \leftarrow \{v_{n - \frac{N_R}{2}}, v_{n - \frac{N_R}{2} + 1}, \cdots, v_{n + \frac{N_R}{2} - 1}\}$

**return** $\tilde{\mathbf{V}}$

Instead of the TCP throughput information, the video level selection approach employs the *client request history* to determine $\tilde{\mathbf{V}}$ which comprises the previously requested video levels of the AHS client. It is assumed that the manifest file, employed by the client for the segment requests during the overall streaming session, contains the full video set $\mathbf{V}$. The AHS server generates and maintains $\tilde{\mathbf{V}}$ as a subset of $\mathbf{V}$ which is generated based on the mean of the previous client requests of the previous $W$ seconds ($\overline{v}_{Req}$). Based on $\tilde{\mathbf{V}}$, the AHS server entity produces the video levels employed in the streaming session. If, however, the AHS client requests a video level which is not contained in $\tilde{\mathbf{V}}$, the AHS server instead selects the video level which offers the smallest absolute difference to the requested video level out of $\tilde{\mathbf{V}}$ as an alternative.

#### 5.3.3.2. Client request-aware dynamic video level selection algorithm

Algorithm 2 depicts the algorithm's pseudocode. The algorithm is performed at time $t$ immediately after the request of a segment when a window of $W$ seconds is completed. Other than the network performance-aware dynamic video level selection algorithms, *CliVLS* does not handle start-up and inter-RAN handover events separately. The algorithm requires $\mathbf{V}$ and $N_R$ as the input arguments and generates $\tilde{\mathbf{V}}$ as the output argument.

At the beginning of the streaming session, the manifest file which contains the full set of video levels $\mathbf{V}$ is provided to the client. After the start-up of the streaming session, no client request history is available at the AHS server yet. Hence, the client selects the video levels from the full video level set according to its rate adaptation strategy for the first $W$ seconds and the

server produces the requested video levels on demand. For all successive time instants, the algorithm determines the video levels of the reduced set $\tilde{\mathbf{V}}$ based on $\overline{v}_{Req}$. The video level in $\mathbf{V}$ which offers the smallest absolute difference to $\overline{v}_{Req}$ is selected as the center element $v_n$ in $\tilde{\mathbf{V}}$. The $N_R - 1$ other video levels of $\tilde{\mathbf{V}}$ are selected symmetrically around $v_n$ according to the windowing selection process proposed in the *NetVLS-M* approach (cf., lines 2-10).

## 5.4. Performance evaluation in automotive streaming scenarios

The proposed network performance-aware and client request-aware dynamic video level selection algorithms are applied to an automotive streaming scenario where the video of an ADAS front-facing camera of a vehicle is upstreamed to a video portal using AHS. The AHS server module, i.e., the MBR encoder and the web server, is installed at the vehicle. It splits the source video into segments of a defined duration, encodes the segments at the defined video levels, and stores the encoded video segments at a locally installed web server along with the manifest file, which is used to inform the client about the video segments. The heterogeneous RAN ECU is used for wireless connectivity. In the course of the performance assessment, HSPA and LTE RANs in an urban deployment are considered which offer almost full coverage in urban areas as of today [Map]. The AHS source entities can be integrated into modern vehicle deployments with minor modifications of the architecture and the ECUs. To this end, the hardware encoder modules of the ADAS camera ECUs can be used for the encoding of the video segments and a web server installed on the HU ECU, which is typically employed in remote HMI applications [EPS10], can be used to store and provide the video segments to the streaming client. The AHS client is installed at a remote video portal deployed in the Internet, which adaptively requests segments from the vehicle using a standard AHS adaptation algorithm. This scenario is especially interesting since the network performance of moving vehicles fluctuates significantly over time due to the properties of the wireless connectivity in automotive environments which suffers from fast-changing wireless channels and frequent inter-RAN handovers [LB+15].

The following section first describes the developed simulation model (Section 5.4.1) followed by a performance assessment of the three video level selection algorithms with respect to the user experience (Section 5.4.2).

### 5.4.1. Simulation model

In the simulation model, two different implementations are considered at the encoder side: (i) a reference implementation (referred to as *Reference/Ref* in the following) where the MBR encoder entity at the AHS server produces the full static video level set $\mathbf{V}$ with $N_L$ different video levels, and (ii) an implementation where the proposed dynamic video level selection algorithms are employed. In order to ensure reliability and generalizability of the results,

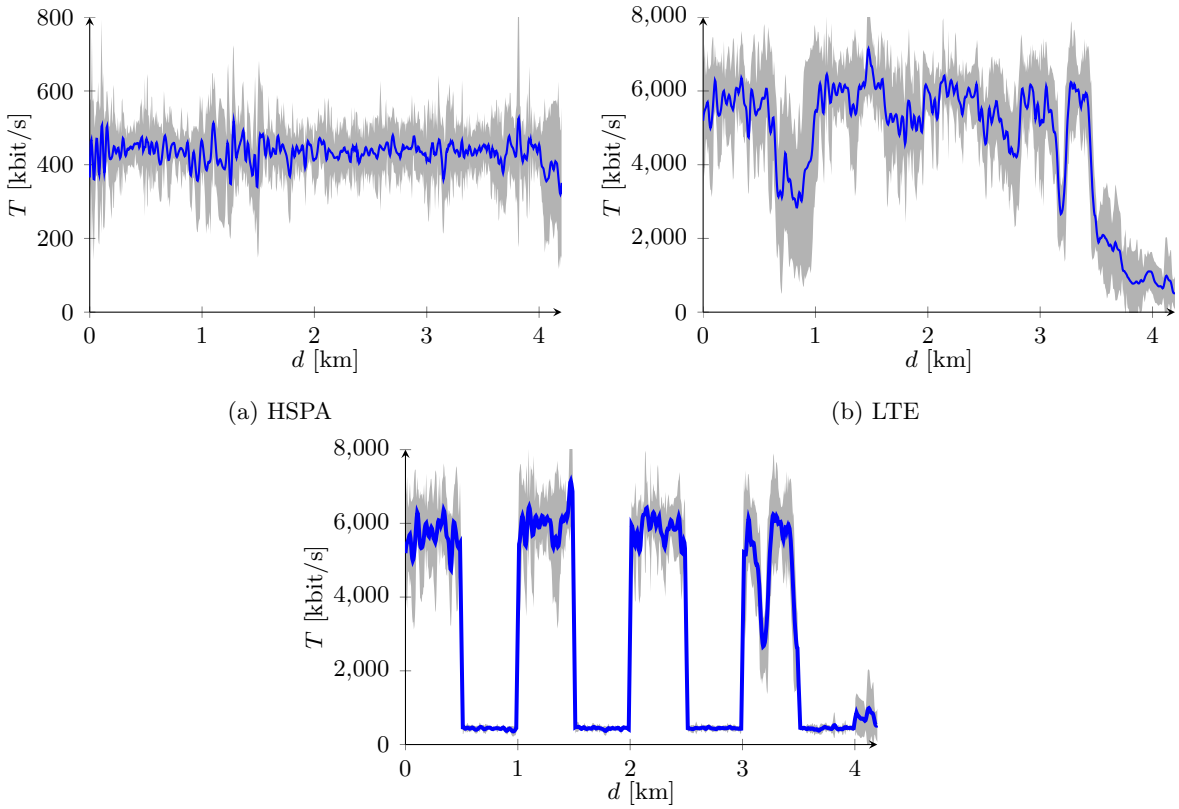
three different standard AHS adaptation algorithms are considered at the streaming sink to determine the performance measures during the streaming sessions: (i) the adaptation algorithm proposed by Liu et al. [LBG11] (referred to as *Liu* in the following), (ii) the adaptation algorithm proposed by Miller et al. [MQ+12] (referred to as *Miller* in the following), and (iii) the adaptation algorithm proposed by Tian et al. [TL13] (referred to as *Tian* in the following).

A modular, discrete *MATLAB* simulation framework is developed to investigate and compare the performance of the different video level selection approaches. The AHS server entities and AHS client entities are implemented as separate modules. The client buffer model proposed in [HJM13] is used to model the playout buffer for the three adaptation algorithms. The network performance between the mobile source and the video portal is modeled by measurements conducted in real HSPA and LTE networks in an automotive environment.

In the following, the conducted TCP network performance measurements and the selection of the static set of video levels employed at the AHS streaming server are introduced.

#### 5.4.1.1. Network performance modeling

In order to model the uplink TCP throughput performance using HSPA and LTE RANs in an urban automotive environment, repeated TCP uplink throughput measurements of a single TCP connection are performed while driving. TCP uplink performance traces are recorded with a server deployed in the Internet using *iperf* [TQ+] along a 4.3 km long route in the urban area of Munich. For connectivity, a ZTE MF821 data modem was connected to the heterogeneous RAN ECU and to an external automotive-grade rooftop antenna system [ET+13]. Nine repeated measurements in the network of Telefónica for both HSPA and LTE networks were conducted. To reduce the effect of direction-dependent gain differences inherent from MIMO antennas, especially in the vicinity of LTE base stations [ET+13], the same driving direction was selected for all measurements. All traces were recorded at an average velocity of 30 km/h. The data points were recorded at a frequency of 1 Hz containing uplink TCP throughput measurements, geographical position (latitude and longitude determined by a GPS module), velocity of the vehicle, and timestamp of the measurements. Figures 5.3a and 5.3b show the mean and standard deviation of the measured uplink TCP throughput $T$ of all measurements for HSPA and LTE networks versus the distance to the start position $d$. It can be observed that for LTE the uplink TCP throughput performance fluctuates around a mean of approximately 5000 kbit/s (except for a significant drop in the last 500 m). On the other hand, the TCP throughput performance of the HSPA traces is more stable over time, but significantly lower as compared to the LTE traces with a mean of roughly 450 kbit/s. For the further analysis, an artificial scenario is created where alternating inter-RAN handovers between HSPA and LTE networks are considered every 60 s. To generate the handover traces, the HSPA and LTE trace segments of 60 s length obtained from respective




(a) HSPA

(b) LTE

(c) Handover

Figure 5.3.: Mean (——) and standard deviation (——) of the measured uplink TCP throughput for HSPA, LTE, and HO measurements over nine traces each.

uplink TCP throughput measurements are piecewise combined (cf., Figure 5.3c).

The mean over all traces for each RAN and geographical position is used to create the TCP uplink performance information in the remote database which provides $T_{DB}$ for the considered network contexts.

#### 5.4.1.2. Selection of the static video level set

For the further investigation, typical CDN settings are considered for the number of video levels in $\mathbf{V}$ [App; TAP+14], i.e., $N_L = 12$. Furthermore, three representative videos of the *Road* video set (introduced in Section 2.3.3.2) are used: video 1, 2, and 6. In order to select the video levels of $\mathbf{V}$, the guidelines for the selection of an optimal video level set proposed in [TAP+14] are followed. To this end, two factors are taken into account: (i) the TCP throughput performance characteristics of the automotive scenario, and (ii) the properties of the *R-Q* curves of the considered videos. First, the lowest and highest video levels in $\mathbf{V}$ are determined such that the uplink throughput range observed during the conducted measurements in the HSPA and LTE networks is fully covered. Secondly, more video levels

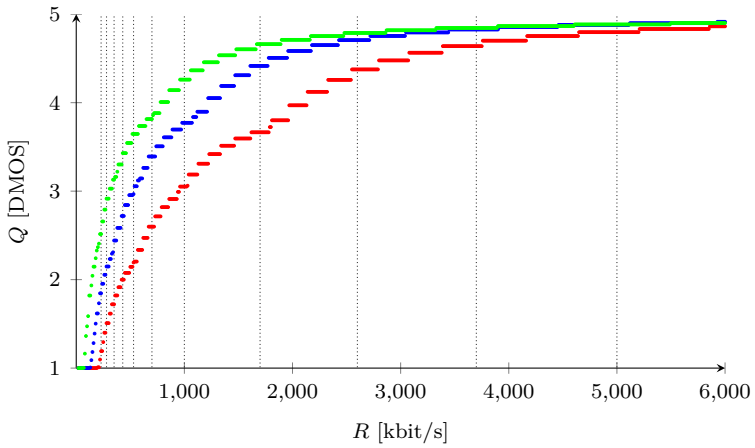



Figure 5.4.: *R-Q* curves of video 1 (·), 2 (·), and 6 (·) of the *Road* video set determined as optimal solutions of Eq. (3.11) using $STRM^{+}$ and $STVQM$; bit rates of selected video levels (·····).

are selected for lower bit rates ($R \leq 1000\,\text{kbit/s}$), since the $Q$ gains are high in this range. The *R-Q* curves displayed in Figure 5.4 reveal that in this bit rate range, a small increase in bit rate leads to a large $Q$ increment. On the contrary, the $Q$ gain is small for bit rate increases above a certain threshold. Figure 5.4 shows that this threshold is roughly 5000 kbit/s for the considered videos, which is selected as the video level with the highest bit rate in $\mathbf{V}$.

Considering these two factors, $\mathbf{V}$ is constructed as:

$$\mathbf{V} = \{200, 230, 280, 350, 430, 530, 700, 1000, 1700, 2600, 3700, 5000\}\,\text{kbit/s}$$

It is assumed that the source videos are separated into video segments of $\tau = 2\,\text{s}$ length and are encoded using the integer $q$ and $f$ settings determined as solutions of the perceptual quality-aware rate control problem of Eq. (3.11), which employ $STRM^{+}$ of Eq. (3.10) and $STVQM$ of Eq. (2.13). For the investigation, the $q$ and $f$ values listed in Table 3.1 are considered.

### 5.4.2. Simulation results

In the following, the performance of the different dynamic video level selection algorithms is determined in a two step approach. First, the influence of the encoder side adaptation time $W$ on the video level selection approaches is investigated. To this end, the minimum number of video levels $N_{R,min}$ is determined for different $W$ values which is required to fulfill the goals of Eq. (5.1), and thus to achieve the same user experience (i.e., $\gamma$, $\epsilon$, $\mu_Q$) as the *Reference* implementation. For this investigation, Liu's adaptation algorithm and the LTE TCP uplink throughput traces are used, since this combination offers the lowest user experience among the single-RAN adaptation scenarios. Table 5.2 reveals that Liu's algorithm offers a

| $W$ [s] | 10 | 20 | 40 | 60 |
|---|---|---|---|---|
| $N_{R,min}$ | 2 | 4 | 4 | 5 |

Table 5.1.: Feasible ($W$, $N_{R,min}$)-pairs of *NetVLS-M*.

| | | $\overline{\gamma}$ [s] | | $\overline{\epsilon}$ | | $\overline{\mu_R}$ [kbit/s] | | $\overline{\mu_Q}$ [DMOS] Video 1 | | Video 2 | | Video 6 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* |
| LTE | Liu | 0.0 | 0.0 | 37 | 27 | 4365.2 | 4203.5 | 4.63 | 4.64 | 4.71 | 4.72 | 4.77 | 4.74 |
| | Miller | 0.0 | 0.0 | 19 | 16 | 4321.9 | 3544.0 | 4.59 | 4.54 | 4.67 | 4.67 | 4.73 | 4.70 |
| | Tian | 0.0 | 0.0 | 13 | 21 | 4383.6 | 3932.3 | 4.51 | 4.57 | 4.60 | 4.67 | 4.68 | 4.71 |
| HO | Liu | 0.0 | 6.4 | 49 | 22 | 2804.5 | 2912.7 | 3.64 | 3.72 | 3.85 | 3.90 | 4.17 | 4.20 |
| | Miller | 4.9 | 6.2 | 47 | 20 | 2190.4 | 1928.8 | 3.17 | 3.40 | 3.39 | 3.64 | 3.76 | 4.00 |
| | Tian | 0.8 | 0.4 | 26 | 25 | 2775.7 | 2632.7 | 3.49 | 3.63 | 3.69 | 3.83 | 4.03 | 4.16 |

Table 5.2.: Performance overview of the *Reference* and *NetVLS-M* (referred to as *Prop* in the table) implementations in LTE and HO scenarios for $N_R = 2$, $W = 10\,\text{s}$, mean over nine traces each.

significantly higher number of video level switches compared to Miller's and Tian's algorithm for the *Reference* implementation, while the other parameters offer approximately the same performance. Second, the identified $W$ and $N_{R,min}$ parameters are used to determine the user experience performance measures for the AHS adaptation algorithms and different network performance scenarios. For this investigation, the LTE and the HO traces are employed, which both feature significant network performance changes as opposed to the HSPA traces which offer stable performance characteristics.

#### 5.4.2.1. Network performance-aware dynamic video level selection

**NetVLS-M:** First, $N_{R,min}$ is determined for different $W$ values ($W \in \{10, 20, 40, 60\}\,\text{s}$). The results listed in Table 5.1 show that $N_{R,min}$ increases for larger $W$ values since the responsiveness of the video level selection approach to throughput fluctuations decreases significantly as $W$ increases. The lowest possible value of $N_{R,min}$ can be achieved for $W = 10\,\text{s}$ ($N_{R,min} = 2$), which is used in the following to assess the performance for the LTE and handover network performance scenarios.

Table 5.2 lists the mean performance over all nine LTE and handover traces of all four performance measures ($\overline{\gamma}$, $\overline{\epsilon}$, $\overline{\mu_R}$, $\overline{\mu_Q}$) for the three different rate adaptation algorithms. The performance of *NetVLS-M* in terms of mean subjective quality $\overline{\mu_Q}$ and interrupted playback time due to stalling events $\overline{\gamma}$ is comparable to the performance of the *Reference* implementation for all client side adaptation algorithms. For Liu's [LBG11] and Miller's [MQ+12] adaptation algorithms, a decrease in the number of video level switches $\overline{\epsilon}$ can be observed. This is due to the small number of video levels which in turn lead to a decreased probability of video level switches. Besides that, the mean video rate $\overline{\mu_R}$ is lower for the scenarios where

| $W$ [s] | 10 | 20 | 40 | 60 |
|---|---|---|---|---|
| $N_{R,min}$ | 4 | 4 | 4 | 5 |

Table 5.3.: Feasible ($W$, $N_{R,min}$)-pairs of *NetVLS-DB*.

| | | $\overline{\gamma}$ [s] | | $\overline{\epsilon}$ | | $\overline{\mu_R}$ [kbit/s] | | $\overline{\mu_Q}$ [DMOS] Video 1 | | Video 2 | | Video 6 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* |
| LTE | Liu | 0.0 | 0.5 | 37 | 28 | 4365.2 | 4275.1 | 4.63 | 4.66 | 4.71 | 4.75 | 4.77 | 4.81 |
| | Miller | 0.0 | 0.0 | 19 | 12 | 4321.9 | 3782.2 | 4.59 | 4.56 | 4.67 | 4.68 | 4.73 | 4.73 |
| | Tian | 0.0 | 0.0 | 13 | 13 | 4383.6 | 4241.3 | 4.51 | 4.63 | 4.60 | 4.73 | 4.68 | 4.79 |
| HO | Liu | 0.0 | 5.6 | 49 | 42 | 2804.5 | 2787.7 | 3.64 | 3.66 | 3.85 | 3.89 | 4.17 | 4.21 |
| | Miller | 4.9 | 6.2 | 47 | 27 | 2190.4 | 1635.9 | 3.17 | 3.17 | 3.39 | 3.50 | 3.76 | 3.95 |
| | Tian | 0.8 | 0.6 | 26 | 25 | 2775.7 | 2216.1 | 3.49 | 3.40 | 3.69 | 3.68 | 4.03 | 4.07 |

Table 5.4.: Performance overview of the *Reference* and *NetVLS-DB* (referred to as *Prop* in the table) implementations in LTE and HO scenarios for $N_R = 4$, $W = 10\,\text{s}$, mean over nine traces each.

*NetVLS-M* is applied compared to the scenarios where the *Reference* implementation is employed (except for the handover scenario where Liu's algorithm is used). However, this does not necessarily lead to a lower $\overline{\mu_Q}$ value, since the *R-Q* relation is not linear (cf., Figure 5.4). Figure 5.5 displays the selected video levels for both *NetVLS-M* and the *Reference* implementation for an exemplary LTE network trace. It can be observed that higher video levels are requested in the start-up phase using *NetVLS-M*, which is due to the usage of $T_{DB}$ in the adaptation process already at $t = 0\,\text{s}$. In contrast to that, the rate adaptation algorithms start with the lowest bit rate video level of $\mathbf{V}$ for the *Reference* implementation, since no information about the potential TCP network performance is available and no pre-selection of the potential video levels is conducted.

**NetVLS-DB:** Table 5.3 lists $N_{R,min}$ for different $W$ values. Similar as for *NetVLS-M*, $N_{R,min}$ increases for larger $W$ values. Compared to *NetVLS-M*, two more video levels ($N_{R,min} = 4$) are required for $W = 10\,\text{s}$ in order to fulfill the goals defined in Eq. (5.1) for the LTE traces and Liu's adaptation algorithm. The reason for this larger number of $N_{R,min}$ lies in the throughput information, which considers solely the TCP network performance from the remote database ($T = T_{DB}$) and as a consequence introduces inaccuracies in the TCP throughput estimation.

The user experience performance results for all three client adaptation algorithms for the LTE and handover network traces displayed in Table 5.4 and the selected video segments for one exemplary LTE network trace displayed in Figure 5.6 indicate a similar trend as *NetVLS-M*. To this end, a significant reduction of the number of quality switches $\overline{\epsilon}$ compared to the *Reference* implementation is achieved for Liu's and Miller's adaptation algorithm. Besides that, the mean perceptual quality in the streaming sessions $\overline{\mu_Q}$ for all algorithms is comparable to the *Reference* implementation.

| $W$ [s] | 10 | 20 | 40 | 60 |
|---|---|---|---|---|
| $N_{R,min}$ | 4 | 5 | 7 | 7 |

Table 5.5.: Feasible ($W$, $N_{R,min}$)-pairs of *CliVLS*.

| | | $\overline{\gamma}$ [s] | | $\bar{\epsilon}$ | | $\overline{\mu_R}$ [kbit/s] | | $\overline{\mu_Q}$ [DMOS] Video 1 | | Video 2 | | Video 6 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* | *Ref* | *Prop* |
| LTE | Liu | 0.0 | 0.0 | 37 | 33 | 4365.2 | 4191.8 | 4.63 | 4.54 | 4.71 | 4.64 | 4.77 | 4.72 |
| | Miller | 0.0 | 0.0 | 19 | 15 | 4321.9 | 4062.2 | 4.59 | 4.34 | 4.67 | 4.45 | 4.73 | 4.58 |
| | Tian | 0.0 | 0.0 | 13 | 5 | 4383.6 | 4364.9 | 4.51 | 4.47 | 4.60 | 4.56 | 4.68 | 4.65 |
| HO | Liu | 0.0 | 1.7 | 49 | 44 | 2804.5 | 2408.3 | 3.64 | 3.41 | 3.85 | 3.63 | 4.17 | 4.02 |
| | Miller | 4.9 | 17.3 | 47 | 35 | 2190.4 | 2295.7 | 3.17 | 3.10 | 3.39 | 3.33 | 3.76 | 3.74 |
| | Tian | 0.8 | 5.0 | 26 | 26 | 2775.7 | 2720.0 | 3.49 | 3.51 | 3.69 | 3.72 | 4.03 | 4.07 |

Table 5.6.: Performance overview of the *Reference* and *CliVLS* (referred to as *Prop* in the table) implementations in LTE and HO scenarios for $N_R = 4$, $W = 10\,\text{s}$, mean over nine traces each.

#### 5.4.2.2. Client request-aware video level selection

Finally, the performance of *CliVLS* is assessed. The dependency between $W$ and $N_{R,min}$ is listed in Table 5.5 and follows a similar trend as for the network performance-aware dynamic video level selection algorithms. For $W = 10\,\text{s}$, four video levels are required in order to achieve the same user experience in the streaming sessions as the *Reference* implementation for Liu's adaptation algorithm and the LTE network performance traces.

Table 5.6 lists the user experience for all client adaptation algorithms for the LTE and handover network traces. Similar as for the network performance-aware dynamic video level selection algorithms, the number of video level switches is reduced significantly as compared to the *Reference* implementation. However the interrupted playback time caused by stalling events $\overline{\gamma}$ is significantly higher in inter-RAN handover scenarios (especially for Miller's algorithm). This behavior is due to the indirect and potentially outdated network performance information inherited from the video level requests. After an inter-RAN handover from LTE to HSPA, for example, the requested video levels might still be inappropriately high. As a consequence, more buffer underflows might occur which lead to more frequent and longer playback interruptions.

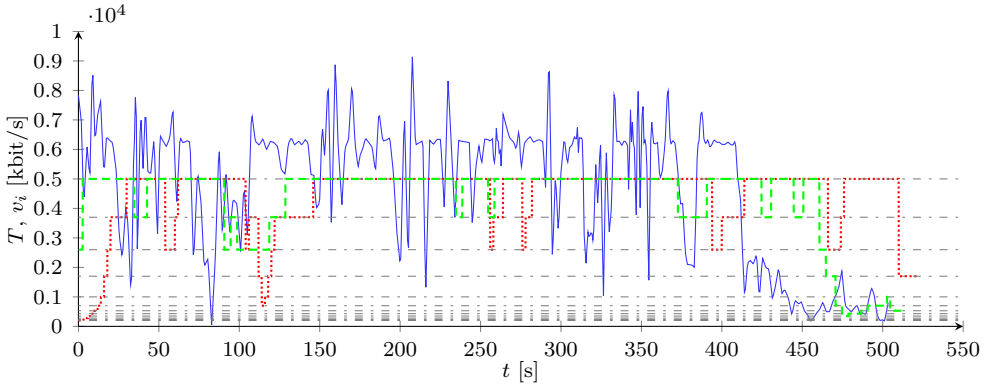
Figure 5.7 displays the video level selections of *CliVLS* for an exemplary LTE network trace. It can be observed that the video levels after the start-up are lower compared to the network performance-aware dynamic video level selection algorithms, since no network performance information is available at $t = 0\,\text{s}$, and, thus, a pre-selection of the video levels cannot be conducted. Due to their rate adaptation strategy, the client adaptation algorithms request the lowest video level of $\mathbf{V}$ at the start-up of the streaming session.

To summarize, the performance assessment reveals that all three proposed dynamic video

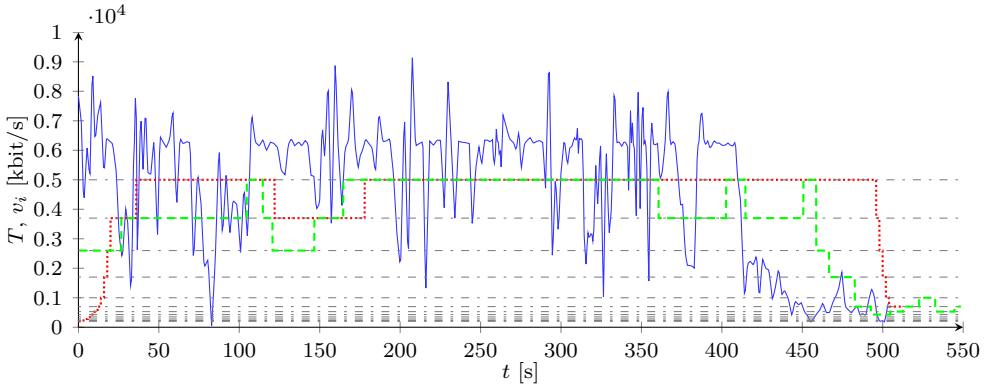
level selection algorithms are able to reduce the number of video levels employed at the AHS video source while achieving the goals of Eq. (5.1). *NetVLS-M*, which primarily employs the TCP network performance measured at the heterogeneous RAN modem of the vehicle for the dynamic video level selection makes it possible to significantly reduce the number of video levels from 12 all the way down to two video levels, as opposed to *NetVLS-DB* and *CliVLS*, which require two more video levels. Despite the larger number of required video levels compared to *NetVLS-M*, *NetVLS-DB* and *CliVLS* offer the major advantage that no additional interface at the heterogeneous RAN modem is required to determine the TCP throughput information. Both algorithms merely depend on context information which can be determined on the application layer. This is particularly beneficial in automotive deployments, where the ECUs are typically developed and produced by automotive suppliers which generally do not grant full access to statistics and the internal functions of the ECUs.

## 5.5. Chapter summary

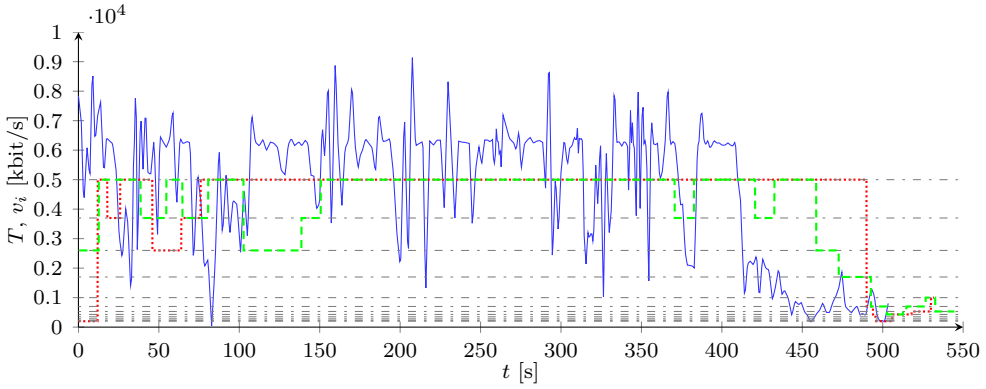
In this chapter, an AHS-based streaming system is considered to upstream videos from a mobile video source with limited computational resources. To reduce the number of video levels which need to be encoded at the AHS server installed at the mobile video source, three context-aware video level selection algorithms are proposed which select a reduced video level set out of a full static video level set. Two algorithms are network performance-aware since they employ TCP throughput information measured or requested from a remote database, and one algorithm uses information about the previous video level requests of the streaming client. The algorithms are applied to an automotive streaming scenario where the video content of an ADAS front-facing camera is upstreamed to a remote video portal using HSPA and LTE RANs. The results of the investigation show that with all three algorithms the number of video levels in a streaming session can be reduced significantly while offering a similar user experience in the streaming sessions as an implementation which employs the full static video level set.




(a) Liu [LB+12a]

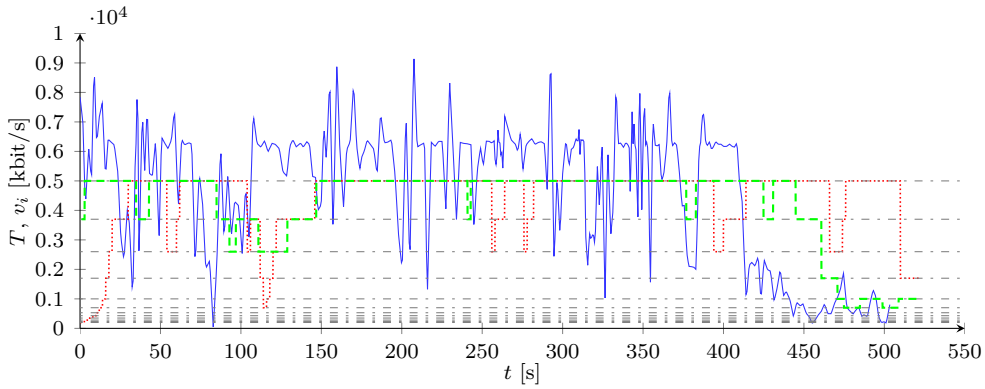



(b) Miller [MQ+12]

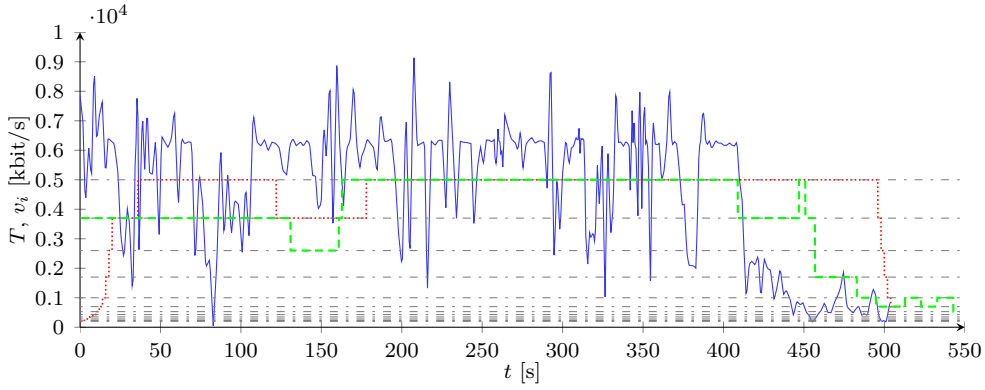



(c) Tian [TL13]

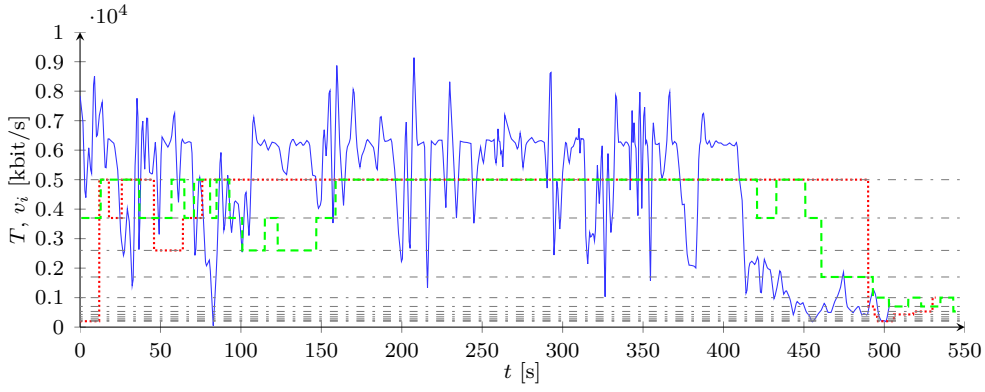
Figure 5.5.: Exemplary LTE TCP uplink network performance trace (——) with the requested video levels using the *Reference* (·····) and *NetVLS-M* (- - - -) AHS server implementations for Liu's, Miller's, and Tian's adaptation algorithms; video levels of **V** (- · - · -).




(a) Liu [LBG11]

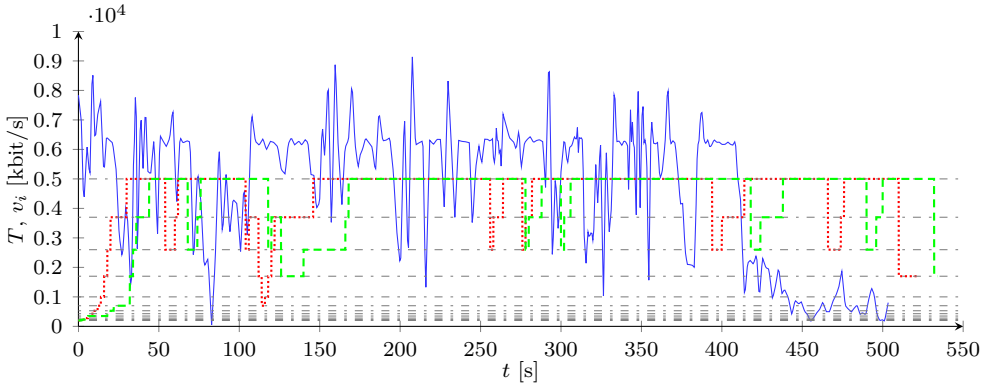



(b) Miller [MQ+12]

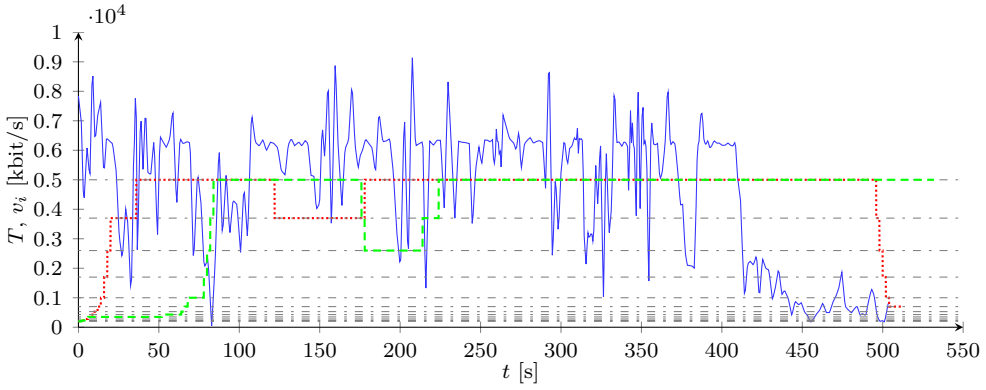



(c) Tian [TL13]

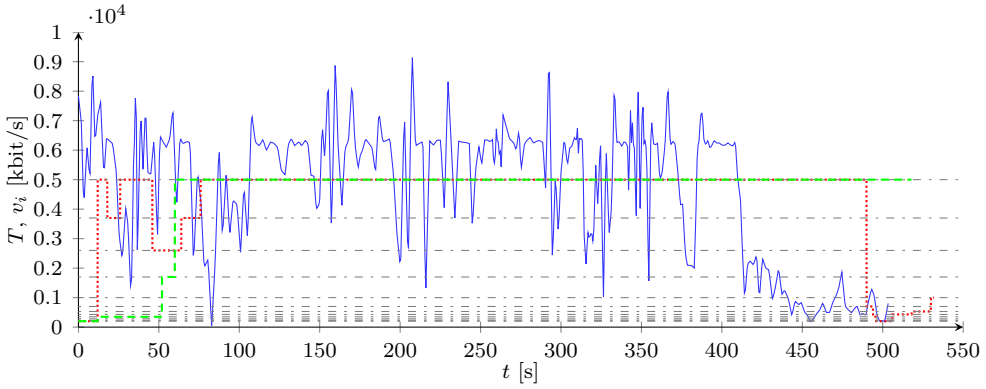
Figure 5.6.: Exemplary LTE TCP uplink network performance trace (——) with the requested video levels for the *Reference* (·····) and *NetVLS-DB* (- - - -) implementations for Liu's, Miller's, and Tian's AHS client adaptation algorithms; video levels of **V** (- - - -).




(a) Liu [LBG11]




(b) Miller [MQ+12]




(c) Tian [TL13]

Figure 5.7.: Exemplary LTE TCP uplink network performance trace (——) with the requested video levels using the *Reference* (·····) and *CliVLS* (- - - -) implementations for Liu's, Miller's, and Tian's AHS client adaptation algorithms; video levels of **V** (· · · ·).

# Chapter 6

# Conclusions and future directions

## 6.1. Conclusions

In this thesis, means to enable user experience-driven video streaming from mobile video sources with limited computational capacity are developed and applied to an automotive environment.

First, a video bit rate model is developed, which takes the impact of spatial quality, temporal resolution, and the GoP settings into account. To this end, the model considers quantization parameter, frame rate, GoP length, and GoP structure as encoding settings and relies on video content-dependent model parameters. To estimate the video content-dependent parameters, temporal and spatial activity dependent estimators are developed, which make it possible to employ the rate model in automated video processing systems, such as rate controllers. An extensive performance assessment with the measured bit rates of two video sets reveals that the proposed bit rate model is very accurate in predicting the bit rate of H.264/AVC encoded videos. Compared to two other related bit rate estimation models, the performance in estimating the video bit rates is comparable or slightly better, however, with the additional advantage that the proposed model takes GoP encoding settings into account. Along with a perceptual quality metric, the developed bit rate model is integrated into the solution of an optimization problem to determine spatio-temporal encoding settings in order to maximize the perceptual quality for desired bit rate constraints. The results show a high accuracy in achieving desired bit rate constraints while using the proposed video bit rate model.

Second, camera context-aware temporal and spatial activity estimators are developed for videos recorded with an ADAS front-facing camera of a vehicle. The estimators employ TA- and SA-related features which are based on the status and the dynamics of the vehicle and the vehicles in the field-of-view of the ADAS camera. A performance assessment shows that the proposed estimators offer a high estimation performance of the computed TA and SA values

with the additional advantage that no access to the uncompressed source video or the internal functions of video encoders is required. Both estimators are applied to the aforementioned bit rate model, the perceptual quality metric, and the solution of the optimization problem to determine spatio-temporal encoding settings for desired bit rate constraints. For all three a similar performance is achieved compared to the solution based on computed TA and SA values.

Finally, an AHS-based system is assumed to upstream video content from a mobile video source with limited computational capacity to a remote streaming sink in the Internet. Mobile video sources might not be able to simultaneously create the same number of video levels which are typically employed in AHS systems installed at CDNs. To significantly reduce the number of video levels at the AHS source, three context-aware video level selection algorithms are proposed. These algorithms select a reduced set of video levels out of a full static video level set. One algorithm employs statistics of the previous video level requests from the client and two algorithms use different sources of the TCP network performance between the video source and the video sink. The algorithms are applied to an automotive streaming scenario where the content of an ADAS front-facing camera of a vehicle is upstreamed to a remote video portal deployed in the Internet. Using all three proposed algorithms, a significant reduction of video levels at the AHS source can be realized while achieving a similar user experience in streaming sessions as an AHS deployment which considers the full static video level set.

## 6.2. Future directions

In this section, potential extensions of the work presented in this thesis are discussed in three directions.

**Video quality metric:** The solution of the rate control optimization problem proposed in Chapter 3 of this thesis employs a perceptual video quality metric which captures the perceived quality of the viewer for a certain range of spatio-temporal encoding settings. While this metric can be employed to maximize the overall QoE in streaming applications, it might not be suitable for some task-related streaming applications where some objects or events contained in the video frames need to be detected by humans. For example, videos of a road scene captured with an ADAS front-facing camera might be used in overtaking applications [GOMF12] where faraway approaching vehicles should be detected. To this end, task-related quality metrics, such as task performance metrics, may be developed and employed in the rate control process rather than perceptual quality metrics.

**Uplink video streaming:** Chapter 5 of this thesis proposes an AHS-based approach to upstream captured videos from a mobile device with limited computational capacity to a single

streaming sink. The approach may be extended in two directions:

- The proposed AHS-based approach to upstream video content from a camera to a remote device in the Internet can be applied for delay tolerant applications, such as automotive surveillance and convenience applications, which allow an end-to-end delay in the order of several seconds. Other applications, however, have more strict delay constraints in the order of milliseconds, such as automotive safety applications, which feature a transmission of the video content of ADAS cameras between two road users to enhance the viewing range in overtaking situations [GOMF12]. To reduce the end-to-end delay between the video source and the sink, the settings of the AHS systems may be tuned targeting a reduction of the transmission time, e.g., by a reduction of the segment length in combination with AHS client algorithms which allow more frequent AHS requests. Besides that, device-to-device communication may be investigated for the transmission of video streams between a video source and a spatially adjacent sink instead of an over-the-top transmission over the Internet.

- The proposed approach focuses on the unicast transmission of videos from a mobile source to a sink. However, real-world scenarios might consider multicast transmissions of the source video to multiple sinks. For example, the video of an ADAS front-facing camera of a source vehicle can be simultaneously upstreamed to multiple streaming clients, such as smartphones, other vehicles, or off-board servers. In this case, the available uplink resources of the mobile video source need to be shared among the different video sinks. To this end, an additional scheduling entity may be considered as an extension at the AHS source entity, which performs a fair resource allocation among the different video sinks.

**Estimation of temporal and spatial activities:** Camera context-aware estimators of spatial and temporal activities for videos captured with an ADAS front-facing camera are developed in this thesis. Suggestions on extensions of the estimators are as follows:

- The developed estimators employ trained model parameters which are limited to video sequences with similar context properties (i.e., similar light and weather conditions). If, however, the source video is recorded at a significantly different context which is not captured by the proposed context features (such as different weather or light conditions), the developed estimators might lead to a poor performance in estimating TA and SA. As a remedy, different estimators may be developed which employ additional context features. To capture the additionally required context information, a remote server may be considered, which extends the proposed estimators by additional information provided by other sources (such as online weather information). The estimators may be re-trained and provided to the vehicle along with the additional context information.

- The sensor information employed in the context features might not be available at all times due to the characteristics of the underlying sensor technology. For example, the GPS information might not always be available as it suffers from shadowing especially in urban scenarios [Gro13]. As a consequence, the developed estimators might provide inaccurate estimates for the TA and SA values. To overcome this limitation, different estimators may be developed which consider subsets of the features of the originally proposed estimators, e.g., a SA estimator which does not take the scenario feature (which relies on the GPS information) into consideration. These estimators may be re-trained according to the proposed methodology, stored in a database, and employed in situations when some features are not available.

# Appendix A

# SAMVIQ guidelines

Figure A.1 displays the graphical user interface of the SAMVIQ-based subjective test. In the preparation phase of the subjective test, the following introduction was presented to the subjects:

**Guidelines:** Within the current investigation, an objective video quality metric to determine the perceived quality of videos captured with an ADAS front-facing camera of the vehicle based on the user experience is investigated. In order to train the objective video quality metric, it is necessary to perform the present subjective test.

In the target application, users are able to identify the traffic situation from different parts of different road situation through the front facing camera of other vehicles. The goal of the present subjective test is to evaluate ***how good the viewer is satisfied with the quality of a video***, when certain video characteristics are modified. Please note that the quality of the original video does not have perfect characteristics in all cases.


Figure A.1.: SAMVIQ graphical user interface applied in the subjective test.

**SAMVIQ Guidelines**

- The test will last approximately 1 hour. It is divided into two subtests of approximately 20 minutes and a break in-between.
- In each subtest, 7 different scenes will be evaluated. For each scene an explicit reference video will be shown along with 12 processed video sequences of the same scene and one hidden reference.
- All sequences from the same scene must be scored before the viewer can proceed to the next scene or previous scene.
- The interface presents a set of buttons that allow to view each of the sequences, one at time, in the video window. The duration of each sequence is 10 seconds.
- On the right hand side of the video window is an interactive slide-bar to rate the quality of the sequence. Based on the task mentioned above, the quality rating can be graded from 1 to 5 (Bad, Poor, Fair, Good, Excellent) with intermediate steps of 0.5 using the scoring slider.
- The viewer is able to jump between the different representations, replay each representation several times and compare the different representations with each other.
- Moving back to previous sequences recalls all the previous ratings.
- The button with label REF identifies the reference sequence. Buttons with letter labels $A$ to $N$ give access to either the hidden reference or one of the processed sequences.

## Appendix B

# Thesis website

A complementary website of this thesis is available at

`http://c.lotterm.org/thesis/`

It contains an overview about the contributions of this thesis and separate websites with supplementary material of the publications which cover a major part of the thesis:

- The complementary website of the *low-complexity and context-aware estimation of TA and SA values for automotive camera rate control* publication [LSS] provides the uncompressed source videos (YUV 4:2:0) of the *Road* video set. Furthermore, the subject ratings of the subjective test presented in Section 2.3.3.2 are available. Finally, the raw sensor data of the ego vehicle's status and the dynamics of the vehicles in the field-of-view of the ADAS front-facing camera for all video sequences are provided in order to reproduce or extend the developed estimators.
  Direct URL: `http://c.lotterm.org/thesis/tcsvt/`

- The complementary website of the *network-aware video level encoding for uplink adaptive HTTP streaming* publication [LG+15] provides the TCP uplink network performance traces for both LTE and HSPA RANs applied in Chapter 5 of this thesis. Additionally, TCP downlink network performance traces captured with the same hardware configuration and in the same scenarios are available which can be used to further extend the conducted research or be used by others for their own research.
  Direct URL: `http://c.lotterm.org/thesis/icc2015/`

- The complementary website of the *modeling the bit rate of H.264/AVC video encoding as a function of quantization parameter, frame rate and GoP characteristics* publication [LS14] offers links to the CIF video sequences and to the tools and settings applied for the encoding of the videos.
  Direct URL: `http://c.lotterm.org/thesis/icmews2014/`

# List of Abbreviations

| Term | Description |
|---|---|
| 3GPP | 3rd Generation Partnership Project |
| ACC | Adaptive cruise control |
| ACR | Absolute category rating |
| ADAS | Advanced driver assistance service |
| AHS | Adaptive HTTP streaming |
| AVC | Advanced Video Codec |
| CABAC | Context-adaptive binary arithmetic coding |
| CAN | Controller Area Network |
| CBR | Constant bit rate |
| CDN | Content delivery network |
| CGW | Central gateway |
| CI | Confidence interval |
| CIF | Common Intermediate Format |
| CVE | Cross-validation error |
| DASH | Dynamic Adaptive Streaming over HTTP |
| DCT | Discrete cosine transform |
| DMOS | Differential mean opinion score |
| DSCQS | Double-stimulus continuous quality scale |
| DSIS | Double-stimulus impairment scale |
| DSRC | Dedicated short range communication |
| ECU | Electronic control unit |
| EDGE | Enhanced Data rates for GSM Evolution |
| ETSI | European Telecommunications Standards Institute |
| FR | Full-reference |
| GLM | Generalized linear regression methodology |
| GNSS | Global navigation satellite system |
| GoP | Group of pictures |
| GPRS | General Packet Radio Service |
| GPS | Global Positioning System |
| GSM | Global System for Mobile Communications |
| HAD | Highly automated driving |
| HEVC | High Efficiency Video Coding |

| **Term** | **Description** |
|---|---|
| HMI | Human-machine interface |
| HSPA | High Speed Packet Access |
| HTTP | Hypertext Transfer Protocol |
| HU | Head unit |
| HVS | Human visual system |
| IP | Internet Protocol |
| ITU | International Telecommunication Union |
| JVT | Joint Video Team |
| LIDAR | Light detection and ranging |
| LIN | Local Interconnect Network |
| LTE | Long Term Evolution |
| MAD | Mean absolute difference |
| MBR | Multiple bit rate coding |
| MIMO | Multiple-input and multiple-output |
| MOS | Mean opinion score |
| MOST | Media Oriented Systems Transport |
| MPEG | Moving Picture Experts Group |
| MSE | Mean square error |
| NAL | Network abstraction layer |
| NR | No-reference |
| OFDMA | Orthogonal frequency-division multiple access |
| PC | Pearson correlation |
| PSNR | Peak signal-to-noise ratio |
| PVS | Processed video sequence |
| QoE | Quality-of-experience |
| QoS | Quality-of-service |
| RADAR | Radio detection and ranging |
| RAN | Radio access network |
| RMSE | Root mean square error |
| RR | Reduced-reference |
| RTCP | Real-Time Control Protocol |
| RTP | Real-Time Transport Protocol |
| RTSP | Real-Time Streaming Protocol |
| SA | Spatial activity |
| SAMVIQ | Subjective Assessment of Multimedia VIdeo Quality |
| SI | Spatial perceptual information |
| SSCQE | Single-stimulus continuous quality evaluation |
| SSIM | Structural similarity index |
| SVC | Scalable video coding |
| TA | Temporal activity |
| TCP | Transmission Control Protocol |
| TI | Temporal perceptual information |
| UDP | User Datagram Protocol |
| UMTS | Universal Mobile Telecommunications System |

| **Term** | **Description** |
|---|---|
| VBR | Variable bit rate |
| VCEG | Video Coding Expert Group |
| VCL | Video coding layer |
| VLC | Variable length coding |
| WAN | Wide area network |

# List of Symbols

In the course of the thesis, scalars are in *italics*, vectors and matrices are in **bold**.

## Video coding and quality estimation

| Symbol | Unit | Description |
|---|---|---|
| $b$ | bit/px | Number of bits per pixel |
| $D$ | dB | Distortion |
| $f$ | fps | Frame rate of a video |
| $m$ | | Number of consecutive B-frames in a GoP |
| $\mu_Q$ | DMOS | Mean perceptual quality of transmitted video segments in a streaming session |
| $\mu_{R_c,Q}$ | DMOS | Mean perceptual quality of a video bit rate set $R_c$ |
| $n$ | | Number of frames in a GoP |
| $N_f$ | | Number of frames of a video sequence |
| $N_q$ | | Number of quantization parameter settings considered for rate control |
| $N_s$ | | Number of subjects in a subjective test |
| $N_t$ | | Number of frame rates considered for rate control |
| $N_v$ | | Number of videos considered in a subjective test |
| $N_x$ | | Number of horizontal pixels of a video frame |
| $N_y$ | | Number of vertical pixels of a video frame |
| $P(x,y)$ | | Pixel value of pixel $(x,y)$ |
| $q$ | | Quantization parameter |
| $Q$ | DMOS | Perceptual quality of a video sequence |
| $r$ | | Subject rating for a video representation |
| $R$ | kbit/s | Bit rate of an encoded video |
| $R_c$ | kbit/s | Bit rate constraint |
| $R_{GoPL}$ | | GoP length bit rate factor (depends on $n$) |
| $R_{GoPS}$ | | GoP structure bit rate factor (depends on $m$ and $n$) |
| $R_{max,I}$ | kbit/s | Bit rate of an encoded video ($q = q_{min}$, $f = f_{max}$, $m = 0$, and $n = 1$) |
| $R_{m=0}$ | kbit/s | Bit rate of an encoded video ($q = q_{min}$, $f = f_{max}$, $m = 0$, and $n = f_{max} \cdot \tau$) |
| $R_s$ | | Spatial bit rate factor (depends on $q$) |
| $R_t$ | | Temporal bit rate factor (depends on $f$) |
| $\tau$ | s | Length of a video segment |
| $\tau_{TA,SA}$ | s | TA and SA window length |

## Camera context estimation

| Symbol | Unit | Description |
|---|---|---|
| $\alpha$ | ° | Standard deviation of the ego vehicle's yaw angle in a video sequence |
| $\beta$ | | Mean number of detected vehicles in the field-of-view in a video sequence |
| $D_t$ | | Vehicles in the field-of-view at time $t$ |
| $\mathbf{d_{rel,i}}$ | m | Relative distance between the ego vehicle and the $i$th vehicle |
| $\delta$ | m | Mean inverse distance to other vehicles in the field-of-view in a video sequence |
| $K$ | | Overall number of detected vehicles in the field-of-view in a video sequence |
| $\lambda$ | m/s | Mean relative velocity of vehicles in the field-of-view in a video sequence |
| $\nu$ | m/s | Mean velocity of the ego vehicle in a video sequence |
| $\omega$ | °/s | Mean yaw rate of the ego vehicle in a video sequence |
| $\mathbf{p_{ego}}$ | ° | Absolute position of the ego vehicle (in decimal degrees) |
| $\psi_{ego}$ | ° | Angle of yaw relative to the start angle of the ego vehicle |
| $\dot{\psi}_{ego}$ | °/s | Yaw rate of the ego vehicle |
| $S$ | | Number of camera context samples of a video sequence |
| $\mathbf{v_{ego}}$ | m/s | Velocity of the ego vehicle |
| $\mathbf{v_{rel,i}}$ | m/s | Relative velocity between the ego vehicle and the $i$th vehicle |
| $\zeta$ | | Scenario where a video sequence is taken |

## Adaptive HTTP streaming

| Symbol | Unit | Description |
|---|---|---|
| $B$ | s | Streaming client buffer fullness |
| $\epsilon$ | | Number of video level switches in a streaming session |
| $\gamma$ | s | Duration of interrupted playback in a streaming session |
| $I$ | s | Number of segments fetched in a streaming session |
| $N_L$ | | Number of video levels employed in an AHS streaming system |
| $N_R$ | | Reduced number of video levels employed in an AHS streaming system |
| $T$ | kbit/s | TCP throughput |
| $T_{DB}$ | kbit/s | Requested TCP throughput information from a remote database |
| $T_M$ | kbit/s | Measured TCP throughput information at the RAN modem of a mobile device |
| $\mathbf{V}$ | | Full static set of video levels |
| $\tilde{\mathbf{V}}$ | | Reduced set of video levels |
| $v$ | kbit/s | Video level element |
| $\overline{v}_{Req}$ | kbit/s | Mean of the previous client requests for a window length of $W$ |
| $W$ | s | Window length of the employed statistic information |

## Mathematical conventions

| Symbol | Description |
|---|---|
| $\lvert . \rvert$ | Absolute value (scalars) and Euclidean norm (vectors) |
| $\widehat{.}$ | Estimator |
| max(.) | Maximum value |
| $\mu(.)$ | Mean value |

| | |
|---|---|
| $\|.\|$ | Number of elements in a vector |
| $\lceil . \rceil$ | Rounding up to the next integer |
| $\lfloor . \rfloor$ | Rounding down to the next integer |
| $\sigma(.)$ | Standard deviation |
| $[.]^{\mathrm{T}}$ | Transpose of [.] |

# List of Figures

1.1. Schematic overview of the considered AHS-based uplink streaming scenario, where a source video stream captured with a mobile (vehicular) video source is upstreamed to a remote streaming sink. . . . . 2

2.1. Encoding and decoding process of a block-based hybrid coding system. Rate control blocks are marked in green. Adapted from [WOZ02]. . . . . 8
2.2. Example frame of the *Football* video [Seq] encoded with H.264/MPEG-4 AVC using x264 [Vid] at different quantization parameter settings. . . . . 9
2.3. Representative MPEG GoP with I-, P-, and B-frames ($n = 12$, $m = 2$). . . . . 11
2.4. Transcoding based adaptation of a video stream. Adapted from [DCMP11]. . . 14
2.5. Adaptation of a video stream using scalable video coding. Adapted from [DCMP11]. . . . . 15
2.6. Adaptation of the video stream using multiple bit rate coding. Adapted from [DCMP11]. . . . . 16
2.7. Representative AHS bit rate adaptation scenario with adaptations of the client as a response to network performance changes. Adapted from [BAB11]. . . . . 18
2.8. Double-stimulus testing system. Adapted from [Ric03]. . . . . 20
2.9. Rating scales of the different video quality assessment methods. Adapted from [IR12]. . . . . 21
2.10. Example of the test organization in SAMVIQ. Adapted from [IR09]. . . . . 22
2.11. Block diagram of no-reference, full-reference and reduced-reference video quality assessment systems. Adapted from [Pen12]. . . . . 25
2.12. Original and distorted versions of an example frame of the *Foreman* video [Seq] at different PSNR levels [Ric03]. . . . . 26
2.13. Example frames of videos of the *Road* video set recorded with an ADAS front-facing camera. . . . . 30
2.14. TA and SA values of the *Road* training and validation sets (full video sequences). 31
2.15. $Q$ versus $f$ for different PSNR values determined using the subjective test for videos of the *Road* training set: 42 dB, 38 dB, 34 dB with 95% CI. . . . . 33

2.16. Performance evaluation of *QM* [FS+07], *VQMTQ* [MX+12], *STVQM* [PS11] for videos of the *Road* validation set; measured $Q$ obtained from the subjective test for 42 dB, 38 dB, 34 dB with 95% CI. . . . . 35
2.17. Performance evaluation of *QM* [FS+07], *VQMTQ* [MX+12], and *STVQM* [PS11] for videos of the *Road* validation set: measured $Q$ vs. estimated $Q$. . . . 35
2.18. Surround sensor configuration of a typical modern vehicle: LIDAR-scanner, RADAR, mono ADAS front-facing camera. Adapted from [AS+12]. . . . . 37
2.19. Exemplary ECU architecture with sensor systems (LIDAR, RADAR, and ADAS front-facing camera) and interconnecting bus systems (CAN and Ethernet). . . . . 41

3.1. Example frames of the videos from the *CIF* video set. . . . . 47
3.2. TA and SA values of the *CIF* training and validation sets (full video sequences). 48
3.3. SA and TA values of video segments of the *CIF* set ($\tau = 1\,\mathrm{s}$). . . . . 49
3.4. $R_s$ versus $q$ for videos of the *Road* and *CIF* training sets. . . . . 50
3.5. $R_t$ versus $f$ for videos of the *Road* and *CIF* training sets. . . . . 51
3.6. $R_{GoPL}$ versus $n$ for videos of the *Road* and *CIF* training sets. . . . . 52
3.7. $R_{GoPS}$ versus $m$ for the considered $n$ values for video 1 of the *Road* and *CIF* video sets each. . . . . 52
3.8. Measured $R$ versus estimated $R$ determined using $STRM^+$ for videos of the *Road* validation set. . . . . 56
3.9. Measured $R$ versus estimated $R$ determined using $STRM^+$ for videos of the *CIF* validation set. . . . . 57
3.10. Influence of $\tau_{TA,SA}$ on the bit rate estimation performance for video 3 of the *CIF* video set: RMSE relative to $R_{max,I}$. . . . . 59
3.11. System view of a MBR encoding entity installed at an AHS source with $N_L$ desired bit rates. The rate controller determines optimal encoding settings as solutions to the perceptual quality-aware rate control problem of Eq. (3.11) using TA and SA computed from the source video. . . . . 61
3.12. $q$ and $f$ determined as solutions of Eq. (3.11) and corresponding $Q$ values for given bit rate constraints for videos of the *Road* validation set. . . . . 62
3.13. $q$ and $f$ determined as solutions of Eq. (3.11) and corresponding $Q$ values for given bit rate constraints for videos of the *CIF* validation set. . . . . 63

4.1. Dynamics of the ego vehicle ($\mathbf{v_{ego}}(t)$, $\dot{\psi}_{ego}(t)$), dynamics of vehicles ($V_i$) in the field-of-view of the ADAS camera ($\mathbf{v_{ego}}(t)$, $\dot{\psi}_{ego}(t)$), dynamics of vehicles ($V_i$) in the field-of-view of the ADAS camera (front-facing LIDAR scanner, front-facing RADAR, ADAS front-facing camera). . . . . 69
4.2. Computed TA values versus temporal activity related features and feature combinations for videos of the *Road* training set. . . . . 74

4.3. Computed SA values versus spatial activity related features for the videos of the *Road* training set. . . . . 75
4.4. $\widehat{TA}$ and $\widehat{TA}_{[\text{LM+14b}]}$ estimation performance for videos of the *Road* training and validation set. . . . . 76
4.5. $\widehat{SA}$ and $\widehat{SA}_{[\text{LM+14b}]}$ estimation performance for videos of the *Road* training and validation set. . . . . 78
4.6. Performance evaluation of *STVQM*, *CSTVQM*, $CSTVQM_{[\text{LM+14b}]}$ for videos of the validation set; measured $Q$ obtained from the subjective test of Section 2.3.3.2 for 42 dB, 38 dB, 34 dB with 95% CI. . . . . 80
4.7. System view of a MBR encoding entity installed at an AHS source with $N_L$ desired video bit rates. The rate controller determines optimal encoding settings as solutions to the rate control optimization problem of Eq. (3.11) using $\widehat{TA}$ and $\widehat{SA}$. . . . . 81

5.1. System image of the considered uplink AHS architecture. The AHS server is installed on the mobile video source. The AHS client, which uses standard AHS adaptation algorithms, is deployed at the streaming sink. . . . . 86
5.2. System model of the dynamic video level selection approaches (referred to as *Dyn. VLS*): network performance-aware dynamic video level selection employs measured and requested TCP network performance information $T$, client request-aware dynamic video level selection uses history of client requests $\overline{v}_{Req}$ . 91
5.3. Mean and standard deviation of the measured uplink TCP throughput for HSPA, LTE, and HO measurements over nine traces each. . . . . 97
5.4. *R-Q* curves of video 1, 2, and 6 of the *Road* video set determined as optimal solutions of Eq. (3.11) using $STRM^{+}$ and $STVQM$; bit rates of selected video levels. . . . . 98
5.5. Exemplary LTE TCP uplink network performance trace with the requested video levels using the *Reference* and *NetVLS-M* AHS server implementations for Liu's, Miller's, and Tian's adaptation algorithms. . . . . 103
5.6. Exemplary LTE TCP uplink network performance trace with the requested video levels for the *Reference* and *NetVLS-DB* implementations for Liu's, Miller's, and Tian's AHS client adaptation algorithms. . . . . 104
5.7. Exemplary LTE TCP uplink network performance trace with the requested video levels using the *Reference* and *CliVLS* implementations for Liu's, Miller's, and Tian's AHS client adaptation algorithms. . . . . 105

A.1. SAMVIQ graphical user interface applied in the subjective test. . . . . 111

# List of Tables

2.1. Subjective test hardware configuration. . . . . 32
2.2. *QM* [FS+07], *VQMTQ* [MX+12], and *STVQM* [PS11] estimation performance: PC and absolute RMSE values for videos of the *Road* training and validation set. 35

3.1. Considered $q$, $f$, $m$, and $n$ encoding settings in the video bit rate model. . . . . 48
3.2. Set of TA- and SA-based features used for the estimator of $R_{max,I}$, and the model parameter estimators of $R_s(q)$ and $R_{GoPL}(n)$. Set of $n$-based features are additionally employed for the model parameter estimators of $R_{GoPS}(m,n)$. 53
3.3. $\widehat{R}_{max,I}$ estimation performance for the videos of the *Road* and *CIF* training sets: PC and absolute RMSE. . . . . 55
3.4. $\widehat{R}_s(q)$, $R_t(f)$, $\widehat{R}_{GoPL}(n)$, $\widehat{R}_{GoPS}(m,n)$ estimation performance for the videos of the *Road* and *CIF* training sets: PC and absolute RMSE. . . . . 55
3.5. $STRM^+$ estimation performance for the videos of the *Road* and *CIF* training and validation sets: PC and RMSE (normalized by $R_{max,I}$). . . . . 56
3.6. Estimation performance of *Ma*, *STRM*, and $STRM^+$ for the videos of the *Road* validation set: PC and absolute RMSE. . . . . 58
3.7. Estimation performance of *Ma*, *STRM*, and $STRM^+$ for the videos of the *CIF* validation set: PC and absolute RMSE. . . . . 58
3.8. Mean relative RMSE (normalized by $R_{m=0}$) and mean perceptual quality $\mu_{R_c,Q}$ of the solutions of the rate control problem of Eq. (3.11) for the validation videos of the *Road* and *CIF* video sets using *Ma*, *STRM*, and $STRM^+$. . . . . 64
3.9. Coefficients of the TA- and SA-based estimators for the content-dependent model parameters for videos of the *Road* set. . . . . 65
3.10. Coefficients of the TA- and SA-based estimators for the content-dependent model parameters for videos of the *CIF* set. . . . . 65

4.1. Computed TA, SA and feature values for all videos of the *Road* training and validation set. . . . . 73
4.2. TA cross-validation results for the videos of the *Road* training set. . . . . 73
4.3. SA cross-validation results for the videos of the *Road* training set. . . . . 75

4.4. $\widehat{TA}$ estimation performance: PC and RMSE relative to the maximum TA value of all videos of the *Road* video set. . . . . 76
4.5. $\widehat{SA}$ estimation performance: PC and RMSE relative to the maximum SA value of all videos of the *Road* video set. . . . . 78
4.6. VQM estimation performance: PC and absolute RMSE values for videos of the *Road* training and validation set. . . . . 79
4.7. Video bit rate estimation performance: PC and RMSE relative to $R_{max,I}$. . . . 81
4.8. Mean relative RMSE (normalized with $R_{m=0}$) and mean perceptual quality $\mu_{R_c,Q}$ of the solutions of the rate control problem of Eq. (3.11) for the videos of the *Road* validation set. . . . . 82

5.1. Feasible ($W$, $N_{R,min}$)-pairs of *NetVLS-M*. . . . . 99
5.2. Performance overview of the *Reference* and *NetVLS-M* implementations in LTE and HO scenarios for $N_R = 2$, $W = 10\,\mathrm{s}$, mean over nine traces each. . . . . 99
5.3. Feasible ($W$, $N_{R,min}$)-pairs of *NetVLS-DB*. . . . . 100
5.4. Performance overview of the *Reference* and *NetVLS-DB* implementations in LTE and HO scenarios for $N_R = 4$, $W = 10\,\mathrm{s}$, mean over nine traces each. . . . 100
5.5. Feasible ($W$, $N_{R,min}$)-pairs of *CliVLS*. . . . . 101
5.6. Performance overview of the *Reference* and *CliVLS* implementations in LTE and HO scenarios for $N_R = 4$, $W = 10\,\mathrm{s}$, mean over nine traces each. . . . . 101

# Bibliography

## Publications by the author

[EL+13] L. Ekiz, C. Lottermann, D. Öhmann, T. Tran, O. Klemp, C. Wietfeld, and C.F. Mecklenbräuker. “Potential of cooperative information for vertical handover decision algorithms”. In: *IEEE Conference on Intelligent Transportation Systems (ITSC)*. The Hague, The Netherlands, 2013 (cit. on p. 41).

[LB+12b] C. Lottermann, M. Botsov, P. Fertl, and R. Müllner. “Performance evaluation of automotive off-board applications in LTE deployments”. In: *IEEE Vehicular Networking Conference (VNC)*. Seoul, South Korea, 2012 (cit. on pp. 40, 41).

[LB+15] C. Lottermann, M. Botsov, P. Fertl, R. Müllner, G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro. “LTE for Vehicular Connectivity”. In: *Vehicular Ad-hoc Networks – New Generation VANETs*. Ed. by R. Scopigno, A. Molinaro, and C. Campolo. Spinger, 2015 (cit. on pp. 40, 41, 95).

[LG+15] C. Lottermann, S. Gül, D. Schroeder, and E. Steinbach. “Network-aware video level encoding for uplink adaptive HTTP streaming”. In: *IEEE International Conference on Communications (ICC)*. London, United Kingdom, 2015 (cit. on pp. 5, 85, 113).

[LM+14a] C. Lottermann, A. Machado, D. Schroeder, Y. Peng, and E. Steinbach. “Bit Rate Estimation for H.264/AVC Video Encoding based on Temporal and Spatial Activities”. In: *IEEE International Conference on Image Processing (ICIP)*. Paris, France, 2014 (cit. on pp. 5, 44, 46, 50, 53, 57, 58).

[LM+14b] C. Lottermann, A. Machado, D. Schroeder, W. Hintermaier, and E. Steinbach. “Camera context based estimation of spatial and temporal activity parameters for video quality metrics in automotive applications”. In: *IEEE International Conference on Multimedia and Expo (ICME)*. Chengdu, China, 2014 (cit. on pp. 5, 64, 68, 70, 71, 76–83).

[LS14] C. Lottermann and E. Steinbach. “Modeling the bit rate of H.264/AVC video encoding as a function of quantization parameter, frame rate and GoP characteristics”. In: *Emerging Multimedia Systems and Applications Workshop in conjunction with IEEE International Conference on Multimedia and Expo (ICME)*. Chengdu, China, 2014 (cit. on pp. 5, 11, 43, 44, 113).

[LSS] C. Lottermann, D. Schroeder, and E. Steinbach. "Low-Complexity and Context-Aware Estimation of Spatial and Temporal Activity Parameters for Automotive Camera Rate Control". In: *IEEE Transactions on Circuits and Systems for Video Technology* (accepted for publication) (cit. on pp. 5, 28, 67, 113).

## General publications

[3GP14] 3GPP. *TS 26.247: Transparent end-to-end Packet-switched Streaming Service (PSS); Progressive Download and Dynamic Adaptive Streaming over HTTP (3GP-DASH)*. 2014 (cit. on p. 19).

[ABD11] S. Akhshabi, A. C. Begen, and C. Dovrolis. "An Experimental Evaluation of Rate-adaptation Algorithms in Adaptive Streaming over HTTP". In: *ACM Conference on Multimedia Systems (MMSys)*. San Jose, CA, USA, 2011 (cit. on pp. 17, 18).

[Ado10] Adobe. *HTTP Dynamic Streaming Datasheet*. Tech. rep. 2010 (cit. on pp. 19, 89).

[AK11] M. Aeberhard and N. Kaempchen. "High-level sensor data fusion architecture for vehicle surround environment perception". In: *International Workshop on Intelligent Transportation (WIT)*. Hamburg, Germany, 2011 (cit. on p. 69).

[Alb04] A. Albert. "Comparison of event-triggered and time-triggered concepts with regard to distributed control systems". In: *Embedded World*. Nuremberg, Germany, 2004 (cit. on p. 38).

[App] Apple. *Best Practices for Creating and Deploying HTTP Live Streaming Media for the iPhone and iPad*. URL: `http://goo.gl/LEjAEo`. Accessed May 15, 2015 (cit. on pp. 86, 89, 97).

[AS+12] M. Aeberhard, S. Schlichtharle, N. Kaempchen, and T. Bertram. "Track-to-Track Fusion With Asynchronous Sensors Using Information Matrix Fusion for Surround Environment Perception". In: *IEEE Transactions on Intelligent Transportation Systems* 13.4 (2012), pp. 1717–1726 (cit. on pp. 37, 38, 70).

[BAB11] A.C. Begen, T. Akgul, and M. Baugher. "Watching Video over the Web: Part 1: Streaming Protocols". In: *IEEE Internet Computing* 15.2 (2011), pp. 54–63 (cit. on pp. 16–18).

[BEK03] A.-M. Bruckstein, M. Elad, and R. Kimmel. "Down-scaling for better transform compression". In: *IEEE Transactions on Image Processing* 12.9 (2003), pp. 1132–1144 (cit. on p. 12).

[BG+09] B. Borsetzky, S. Gläser, J. Kahle, and U. Dietz. *CoCar Feasibility Study*. Tech. rep. 2009 (cit. on p. 40).

[Bli06] J.-L. Blin. "New quality evaluation method suited to multimedia context: SAMVIQ". In: *International Workshop on Video Processing and Quality Metrics (VPQM)*. Scottsdale, AZ, USA, 2006 (cit. on p. 22).

[BRK09] A. Bhat, I. Richardson, and S. Kannangara. "A new perceptual quality metric for compressed video". In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Taipei, Taiwan, 2009 (cit. on p. 27).

[BV+14] E. Belyaev, A. Vinel, M. Jonsson, and K. Sjoberg. "Live video streaming in IEEE 802.11p vehicular networks: Demonstration of an automotive surveillance application". In: *IEEE Conference on Computer Communications Workshops (INFOCOM Workshops)*. Toronto, Canada, 2014 (cit. on p. 1).

[Cis13] Cisco. *Cisco Visual Networking Index: Forecast and Methodology (2012-2017)*. Tech. rep. 2013 (cit. on p. 1).

[CN07] Z. Chen and K. N. Ngan. "Recent advances in rate control for video coding". In: *Signal Processing: Image Communication* 22.1 (2007), pp. 19–38 (cit. on p. 12).

[CS+11] S. Chikkerur, V. Sundaram, M. Reisslein, and L.J. Karam. "Objective Video Quality Assessment Methods: A Classification, Review, and Performance Comparison". In: *IEEE Transactions on Broadcasting* 57.2 (2011), pp. 165–182 (cit. on p. 27).

[CZ+11] X. Chen, Z. Zhao, A. Rahmati, Y. Wang, and L. Zhong. "Sensor-Assisted Video Encoding for Mobile Devices in Real-World Environments". In: *IEEE Transactions on Circuits and Systems for Video Technology* 21.3 (2011), pp. 335–349 (cit. on p. 68).

[DCMP11] L. De Cicco, S. Mascolo, and V. Palmisano. "Feedback Control for Adaptive Live Video Streaming". In: *ACM Conference on Multimedia Systems (MMSys)*. San Jose, CA, USA, 2011 (cit. on pp. 14–17, 88).

[Det89] J. Detlefsen. *Radartechnik: Grundlagen, Bauelemente, Verfahren, Anwendungen*. Nachrichtentechnik (Springer-Verlag). Springer, 1989 (cit. on p. 37).

[DPS11] E. Dahlman, S. Parkvall, and J. Skold. *4G: LTE/LTE-Advanced for Mobile Broadband*. Vol. 2011. Academic Press, 2011 (cit. on p. 40).

[DW+12] P. Dollar, C. Wojek, B. Schiele, and P. Perona. "Pedestrian Detection: An Evaluation of the State of the Art". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.4 (2012), pp. 743–761 (cit. on p. 36).

[EK+14] K. Evensen, T. Kupka, H. Riiser, P. Ni, R. Eg, C. Griwodz, and P. Halvorsen. "Adaptive media streaming to mobile devices: challenges, enhancements, and recommendations". In: *Advances in Multimedia* (2014), p. 10 (cit. on p. 18).

[Eki14] L. Ekiz. "Vehicular service delivery via hybrid access and antennas". PhD thesis. Technische Universität Wien, 2014 (cit. on p. 40).

[EPS10] M. Eichhorn, M. Pfannenstein, and E. Steinbach. "A Flexible In-vehicle HMI Architecture Based on Web Technologies". In: *International Workshop on Multimodal Interfaces for Automotive Applications*. Hong Kong, China, 2010 (cit. on pp. 41, 95).

[ET+13] L. Ekiz, A Thiel, O. Klemp, and C.F. Mecklenbräuker. "MIMO performance evaluation of automotive qualified LTE antennas". In: *European Conference on Antennas and Propagation (EUCAP)*. Gotenburg, Sweden, 2013 (cit. on p. 96).

[ETS12] ETSI. *EEN 302 663, Intelligent Transport Systems (ITS) - Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band*. 2012 (cit. on p. 40).

[FD+02] K. C. Fuerstenberg, K. C. J. Dietmayer, S. Eisenlauer, and V. Willhoeft. “Multilayer laserscanner for robust object tracking and classification in urban traffic scenes”. In: *Proceedings of ITS*. Chicago, IL, USA, 2002 (cit. on p. 37).

[FG+99] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. *RFC2616 - Hypertext Transfer Protocol – HTTP/1.1*. 1999 (cit. on p. 17).

[Fle01] W.J. Fleming. “Overview of automotive sensors”. In: *IEEE Sensors Journal* 1.4 (2001), pp. 296–308 (cit. on pp. 36, 38).

[FS+07] R. Feghali, F. Speranza, D. Wang, and A. Vincent. “Video Quality Metric for Bit Rate Control via Joint Adjustment of Quantization and Frame Rate”. In: *IEEE Transactions on Broadcasting* 53.1 (2007), pp. 441–446 (cit. on pp. 27, 28, 35).

[FS+11] D.H. Finstad, H.K. Stensland, H. Espeland, and P. Halvorsen. “Improved Multi-Rate Video Encoding”. In: *IEEE International Symposium on Multimedia (ISM)*. Dana Point, California, 2011 (cit. on p. 15).

[FS+14] Y. Fangchun, W. Shangguang, L. Jinglin, L. Zhihan, and S. Qibo. “An overview of Internet of Vehicles”. In: *IEEE Communications China* 11.10 (2014) (cit. on p. 36).

[Gha99] M. Ghanbari. *Video Coding: An Introduction to Standard Codecs*. Stevenage, UK: Institution of Electrical Engineers, 1999 (cit. on p. 10).

[GHT08] M. Ghanbari and Q. Huynh-Thu. “Scope of validity of PSNR in image/video quality assessment”. In: *IEEE Electronics Letters* 44.13 (2008), pp. 800–801 (cit. on pp. 3, 26, 60).

[Gir93] B. Girod. “Digital Images and Human Vision”. In: Cambridge, MA, USA: MIT Press, 1993. Chap. What’s Wrong with Mean-squared Error?, pp. 207–220 (cit. on pp. 3, 26, 60).

[GKL05] R. Gentile, D. Katz, and T. Lukasiak. “Video-Techniken für mehr Sicherheit im Auto”. In: *HANSER automotive - electronics systems*. 2005, pp. 66–71 (cit. on p. 38).

[GOMF12] P. Gomes, C. Olaverri-Monreal, and M. Ferreira. “Making Vehicles Transparent Through V2V Video Streaming”. In: *IEEE Transactions on Intelligent Transportation Systems* 13.2 (2012), pp. 930–938 (cit. on pp. 108, 109).

[Gro13] P. D. Groves. *Principles of GNSS, inertial, and multisensor integrated navigation systems*. Artech House, 2013 (cit. on pp. 37, 110).

[HC+06] Y.W. Huang, C.-Y. Chen, C.-H. Tsai, C.-F. Shen, and L.-G. Chen. “Survey on Block Matching Motion Estimation Algorithms and Architectures with New Results”. In: *Journal of VLSI signal processing systems for signal, image and video technology* 42.3 (2006), pp. 297–320 (cit. on p. 10).

[HC97] H.-M. Hang and J.-J. Chen. “Source model for transform video coder and its application. I. Fundamental theory”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 7.2 (1997), pp. 287–298 (cit. on p. 12).

[HF+09] I. Herranz, S. Fikar, E. Biebl, and A.L. Scholtz. "Automotive multi-standard RF front-end for GSM, WCDMA and Mobile WiMAX". In: *Wireless Telecommunications Symposium (WTS)*. Prague, Czech Republic, 2009 (cit. on pp. 40, 42).

[HH+06] Y.-W. Huang, B.-Y. Hsieh, S.-Y. Chien, S.-Y. Ma, and L.-G. Chen. "Analysis and complexity reduction of multiple reference frames motion estimation in H.264/AVC". In: *IEEE Transactions on Circuits and Systems for Video Technology* 16.4 (2006), pp. 507–522 (cit. on p. 11).

[HH+12] T.-Y. Huang, N. Handigol, B. Heller, N. McKeown, and R. Johari. "Confused, timid, and unstable: picking a video streaming rate is hard". In: *ACM Conference on Internet Measurement (IMC)*. Boston, MA, USA, 2012 (cit. on p. 88).

[Hin11] W. Hintermaier. "An IP-based System Architecture for Camera-based Driver Assistance Services". PhD thesis. Technische Universität München, 2011 (cit. on p. 42).

[HJM13] T.-Y. Huang, R. Johari, and N. McKeown. "Downton Abbey Without the Hiccups: Buffer-based Rate Adaptation for HTTP Video Streaming". In: *ACM SIGCOMM Workshop on Future Human-centric Multimedia Networking (FhMN)*. Hong Kong, China, 2013 (cit. on pp. 88, 96).

[HT00] H. Holma and A. Toskala. *WCDMA for UMTS*. Vol. 2006. Wiley, 2000 (cit. on p. 40).

[HT09] H. Holma and A. Toskala. *LTE for UMTS - OFDMA and SC-FDMA Based Radio Access*. Wiley, 2009 (cit. on p. 40).

[HT11] H. Holma and A. Toskala. *LTE for UMTS: Evolution to LTE-Advanced*. Wiley, 2011 (cit. on p. 41).

[HV91] P.G. Howard and J.S. Vitter. "Analysis of arithmetic coding for data compression". In: *IEEE Data Compression Conference (DCC)*. Snowbird, USA, 1991, pp. 3–12 (cit. on p. 10).

[IR00] ITU-R. *Rec. J.143: User requirements for objective perceptual video quality measurements in digital cable television*. 2000 (cit. on p. 24).

[IR08] ITU-R. *Rec. BT.910: Subjective video quality assessment methods for multimedia applications*. 2008 (cit. on pp. 20–24, 78).

[IR09] ITU-R. *Rec. BT.1788: Methodology for the subjective assessment of video quality in multimedia applications*. 2009 (cit. on pp. 20–22, 24, 31, 32).

[IR12] ITU-R. *Rec. BT.500-13: Methodology for the subjective assessment of the quality of television pictures*. 2012 (cit. on pp. 20–22, 24).

[IR98] ITU-R. *Rec. P.800: Methods for subjective determination of transmission quality*. 1998 (cit. on p. 20).

[ISO00] ISO/IEC. *13818-2: Generic coding of moving pictures and associated audio information – Part 2: Video*. 2000 (cit. on pp. 11, 13).

[ISO03] ISO/IEC. *14496-10: Coding of audio-visual objects – Part 10: Advanced Video Coding*. 2003 (cit. on pp. 11–13).

[ISO04] ISO/IEC. *14496-2: Coding of audio-visual objects – Part 2: Visual*. 2004 (cit. on p. 13).

[ISO13] ISO/IEC. *FDIS 23008-2: High efficiency coding and media delivery in heterogeneous environments – Part 2: High efficiency video coding*. 2013 (cit. on p. 13).

[ISO14] ISO/IEC. *23009-1: Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats*. 2014 (cit. on p. 19).

[ISO93] ISO/IEC. *11172-2: Coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbps*. 1993 (cit. on p. 11).

[IT96] ITU-T. *Rec. H.263: Video coding for low bit rate communication*. 1996 (cit. on p. 13).

[ITU08] ITU. *WP6Q/131L: Technical Report – Comparison of DSCQS and ACR*. 2008 (cit. on p. 21).

[ITU14] ITU. *Annex 11 to Document 5A/636-E: Radio interface standards of vehicle-to-vehicle and vehicle-to-infrastructure communications for intelligent transport systems applications*. 2014 (cit. on p. 40).

[JB+09] K. Johansson, J. Bergman, D. Gerstenberger, M. Blomgren, and A. Wallen. "Multi-Carrier HSPA Evolution". In: *IEEE Vehicular Technology Conference (VTC)*. Barcelona, Spain, 2009 (cit. on p. 40).

[KC+06] S. Kanumuri, P.-C. Cosman, A.-R. Reibman, and V.-A. Vaishampayan. "Modeling packet-loss visibility in MPEG-2 video". In: *IEEE Transactions on Multimedia* 8.2 (2006), pp. 341–355 (cit. on p. 25).

[KD+07] S. Khan, S. Duhovnikov, E. Steinbach, and W. Kellerer. "MOS-based Multiuser Multiapplication Cross-layer Optimization for Mobile Multimedia Communication". In: *Advances in Multimedia* 2007.1 (2007) (cit. on p. 27).

[KHM01] Y.-K. Kim, Z. He, and S.-K. Mitra. "A novel linear source model and a unified rate control algorithm for H.263/MPEG-2/MPEG-4". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Salt Lake City, UT, USA, 2001 (cit. on p. 45).

[Kra08] U. Kramer. *Kraftfahrzeugführung: Modelle - Simulation - Regelung*. Vol. 2008. Hanser Verlag, 2008 (cit. on p. 36).

[LB+12a] C. Liu, I. Bouazizi, M.-M. Hannuksela, and M. Gabbouj. "Rate adaptation for dynamic adaptive streaming over HTTP in content distribution network". In: *Signal Processing: Image Communication* 27.4 (2012), pp. 288–311 (cit. on pp. 88, 103).

[LBG11] C. Liu, I. Bouazizi, and M. Gabbouj. "Rate Adaptation for Adaptive HTTP Streaming". In: *ACM Conference on Multimedia Systems (MMSys)*. San Jose, CA, USA, 2011 (cit. on pp. 87, 96, 99, 104, 105).

[LCZ00] H.-J. Lee, T. Chiang, and Y.-Q. Zhang. "Scalable rate control for MPEG-4 video". In: *IEEE Transactions on Circuits and Systems for Video Technology* 10.6 (2000), pp. 878–894 (cit. on p. 45).

[LH87] D.-A. Lelewer and D.-S. Hirschberg. "Data Compression". In: *ACM Computing Surveys* 19.3 (1987), pp. 261–296 (cit. on p. 10).

[LK05] S. Liu and C.-J. Kuo. “Joint temporal-spatial bit allocation for video coding with dependency”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 15.1 (2005), pp. 15–26 (cit. on p. 12).

[LSW05] K.-P. Lim, G. J. Sullivan, and T. Wiegand. “Joint Model Reference Encoding Methods and Decoding Concealment Methods”. In: *Study of ISO/IEC 14496-10 and ISO/IEC 14496-5/AMD6* (2005) (cit. on p. 45).

[LVH11] H.-T. Lim, L. Völker, and D. Herrscher. “Challenges in a future IP/Ethernet-based in-car network for real-time applications”. In: *ACM/EDAC/IEEE Design Automation Conference (DAC)*. San Diego, CA, USA, 2011, pp. 7–12 (cit. on p. 39).

[LZ+14] Z. Li, X. Zhu, J. Gahm, R. Pan, H. Hu, A. C. Begen, and D. Oran. “Probe and adapt: Rate adaptation for HTTP video streaming at scale”. In: *IEEE Journal on Selected Areas in Communications* 32.4 (2014), pp. 719–733 (cit. on p. 88).

[Ma11] Z. Ma. “Modeling of Power, Rate, and Perceptual Quality of Scalable Video and Its Applications”. PhD thesis. NYU-Poly, 2011 (cit. on p. 54).

[Map] Telefonica Germany Coverage Map. URL: `http://goo.gl/7wT26x`. Accessed May 15, 2015 (cit. on p. 95).

[MFW13] Z. Ma, F.C.A. Fernandes, and Y. Wang. “Analytical rate model for compressed video considering impacts of spatial, temporal and amplitude resolutions”. In: *Emerging Multimedia Systems and Applications Workshop in conjunction with IEEE International Conference on Multimedia and Expo Workshops (ICME)*. San Jose, CA, USA, 2013 (cit. on p. 14).

[ML+12] K. P. Mok, X. Luo, W. W. Chan, and K. C. Chang. “QDASH: A QoE-aware DASH System”. In: *ACM Conference on Multimedia Systems (MMSys)*. Chapel Hill, NC, USA, 2012 (cit. on p. 88).

[MN90] P. McCullagh and J. A. Nelder. *Generalized Linear Models*. Chapman and Hall, 1990 (cit. on p. 53).

[MQ+12] K. Miller, E. Quacchio, G. Gennari, and A Wolisz. “Adaptation algorithm for adaptive streaming over HTTP”. In: *International Packet Video Workshop (PV)*. Munich, Germany, 2012 (cit. on pp. 88, 96, 99, 103–105).

[MX+12] Z. Ma, M. Xu, Y.-F. Ou, and Y. Wang. “Modeling of Rate and Perceptual Quality of Compressed Video as Functions of Frame Rate and Quantization Stepsize and Its Applications”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 22.5 (2012), pp. 671–682 (cit. on pp. 27–29, 35, 44, 46, 50, 54, 57, 58, 63, 64).

[NE+11] P. Ni, R. Eg, A. Eichhorn, C. Griwodz, and P. Halvorsen. “Spatial flicker effect in video scaling”. In: *Interantional Workshop on Quality of Experience (QoMEX)*. Mechelen, Belgium, 2011 (cit. on p. 89).

[NHB05] T. Nolte, H. Hansson, and L.L. Bello. “Automotive communications-past, current and future”. In: *IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*. Catania, Italy, 2005 (cit. on p. 39).

[NS+05] N. Navet, Yeqiong Song, F. Simonot-Lion, and C. Wilwert. "Trends in Automotive Communication Systems". In: *Proceedings of the IEEE* 93.6 (2005), pp. 1204–1223 (cit. on p. 39).

[ODZ07] T. Oelbaum, K. Diepold, and W. Zia. "A generic method to increase the prediction accuracy of visual quality metrics". In: *Picture Coding Symposium (PCS)*. Lisboa, Portugal, 2007 (cit. on p. 27).

[OR98] A. Ortega and K. Ramchandran. "Rate-distortion methods for image and video compression". In: *IEEE Signal Processing Magazine* 15.6 (1998), pp. 23–50 (cit. on p. 12).

[OS+12] J. Ohm, G. J. Sullivan, H. Schwarz, K. Tan, and T. Wiegand. "Comparison of the Coding Efficiency of Video Coding Standards – Including High Efficiency Video Coding (HEVC)". In: *IEEE Transactions on Circuits and Systems for Video Technology* 22.12 (2012), pp. 1669–1684 (cit. on p. 13).

[Pen12] Y. Peng. "Quality of Experience-Driven Low-Delay Error-Resilient Video Communication". PhD thesis. Technische Universität München, 2012 (cit. on pp. 25, 28, 29, 34).

[PF98] K. Pearson and L. N. G. Filon. "Mathematical contributions to the theory of evolution. IV. On the probable errors of frequency constants and on the influence of random selection on variation and correlation". In: *Philosophical Transactions of the Royal Society of London. Series A.* (1898), pp. 229–311 (cit. on p. 34).

[PM14] R.P. Pantos and W.M. May. *RFC Draft – HTTP Live Streaming*. 2014 (cit. on pp. 19, 89).

[Pos80] J. Postel. *RFC768 – User Datagram Protocol*. 1980 (cit. on p. 16).

[Pos81] J. Postel. *RFC793 – Transmission Control Protocol*. 1981 (cit. on p. 17).

[PS11] Y. Peng and E. Steinbach. "A novel full-reference video quality metric and its application to wireless video transmission". In: *IEEE International Conference on Image Processing (ICIP)*. Brussels, Belgium, 2011 (cit. on pp. 23, 27, 29, 35, 46, 63).

[PW02] M. Pinson and S. Wolf. *NTIA Technical Report TR-02-392: Video Quality Measurement Techniques*. 2002 (cit. on p. 27).

[PW03] M. Pinson and S. Wolf. "Comparing subjective video quality testing methodologies". In: *SPIE Video Communications and Image Processing* (2003), pp. 573–582 (cit. on p. 21).

[Rah09] M. Rahmani. "A Resource-Effcient IP-based Network Architecture for In-Vehicle Communication". PhD thesis. Technische Universität München, 2009 (cit. on pp. 38, 39, 41).

[Rei11] K. Reif. *Bosch Autoelektrik und Autoelektronik*. Wiesbaden: Vieweg+Teubner, 2011, p. 595 (cit. on p. 36).

[RG05] R. H. Rasshofer and K. Gresser. "Automotive Radar and Lidar Systems for Next Generation Driver Assistance Functions". In: *Advances in Radio Science* 3 (2005), pp. 205–209 (cit. on p. 37).

[Ric03] I.-E. Richardson. *H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia*. Wiley, 2003 (cit. on pp. 7, 19, 20, 26).

[RP+10] D.-M. Rouse, R. Pepion, P. Le Callet, and S.-S. Hemami. "Tradeoffs in subjective testing methods for image and video quality assessment". In: *Proceeings of SPIE* 7527 (2010) (cit. on p. 22).

[RR97] M. J. Riley and I. E. G. Richardson. *Digital Video Communications*. Artech House, Inc., 1997 (cit. on p. 9).

[SC+03] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. *RFC3550 - RTP: A Transport Protocol for Real-Time Applications*. 2003 (cit. on p. 16).

[Sch05] M. Schneider. "Automotive radar – status and trends". In: *German microwave conference (GeMiC)*. Ulm, Germany, 2005, pp. 144–147 (cit. on p. 37).

[Sch09] T. Schaller. "Stauassistenz – Längs- und Querführung im Bereich niedriger Geschwindigkeit". PhD thesis. Technische Universtität München, 2009 (cit. on p. 36).

[SD14] C. Sommer and F. Dressler. *Vehicular Networking*. Cambridge University Press, 2014 (cit. on pp. 39, 41).

[SE+13] D. Schroeder, A. Essaili, E. Steinbach, D. Staehle, and M. Shehada. "Low-Complexity No-Reference PSNR Estimation for H.264/AVC Encoded Video". In: *International Packet Video Workshop (PV)*. San Jose, CA, USA, 2013 (cit. on pp. 60–62).

[SE+14] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hossfeld, and P. Tran-Gia. "A Survey on Quality of Experience of HTTP Adaptive Streaming". In: *IEEE Communications Surveys Tutorials* 99 (2014) (cit. on p. 89).

[Seq] YUV Video Sequences. URL: `http://trace.eas.asu.edu/yuv/`. Accessed May 15, 2015 (cit. on pp. 9, 26, 47).

[SF68] I. Sobel and G. Feldman. "A 3x3 Isotropic Gradient Operator for Image Processing". Presented at a talk at the Stanford Artificial Project. 1968 (cit. on p. 23).

[SFR07] P. Seeling, F. Fitzek, and M. Reisslein. *Video Traces for Network Performance Evaluation – A Comprehensive Overview and Guide on Video Traces and Their Utilization in Networking Research*. Springer, 2007 (cit. on p. 10).

[Sha48] C. E. Shannon. "A mathematical theory of communication". In: *Bell System Technical Journal* 27 (1948), pp. 379–423 (cit. on p. 12).

[SMW07] H. Schwarz, D. Marpe, and T. Wiegand. "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard". In: *IEEE Transactions on Circuits and Systems for Video Technology* 17.9 (2007), pp. 1103–1120 (cit. on pp. 14, 15).

[SO+12] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand. "Overview of the High Efficiency Video Coding (HEVC) Standard". In: *IEEE Transactions on Circuits and Systems for Video Technology* 22.12 (2012), pp. 1649–1668 (cit. on pp. 10, 13).

[Sod11] I. Sodagar. "The MPEG-DASH Standard for Multimedia Streaming Over the Internet". In: *IEEE MultiMedia* 18.4 (2011), pp. 62–67 (cit. on p. 18).

[SRL98] H. Schulzrinne, A. Rao, and R. Lanphier. *RFC2326 - Real Time Streaming Protocol (RTSP)*. 1998 (cit. on p. 16).

[SRS15] D. Schroeder, P. Rehm, and E. Steinbach. “Block structure reuse for multi-rate High Efficiency Video Coding”. In: *IEEE International Conference on Image Processing (ICIP)*. Québec City, Canada, 2015 (cit. on p. 15).

[SS02] D. Strelow and S. Singh. “Optimal motion estimation from visual and inertial measurements”. In: *IEEE Workshop on Applications of Computer Vision (WACV)*. Orlando, FL, USA, 2002 (cit. on p. 68).

[STL04] G. J. Sullivan, P. N. Topiwala, and A. Luthra. “The H.264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions”. In: *Proceedings SPIE on Applications of Digital Image Processing* 27 (2004), pp. 454–474 (cit. on p. 13).

[Sto11] T. Stockhammer. “Dynamic Adaptive Streaming over HTTP – Standards and Design Principles”. In: *ACM Conference on Multimedia Systems (MMSys)*. San Jose, CA, USA, 2011 (cit. on pp. 17, 18, 58, 85).

[Sun00] M.-T. Sun. *Compressed video over networks*. CRC Press, 2000 (cit. on p. 9).

[TAP+14] L. Toni, R. Aparicio-Pardo, G. Simon, A. Blanc, and P. Frossard. “Optimal Set of Video Representations in Adaptive Streaming”. In: *ACM Conference on Multimedia Systems (MMSys)*. Singapore, Singapre, 2014 (cit. on pp. 62, 86, 89, 97).

[TL13] G. Tian and Y. Liu. “On Adaptive HTTP Streaming to Mobile Devices”. In: *International Packet Video Workshop (PV)*. Klagenfurt, Austria, 2013 (cit. on pp. 88, 96, 103–105).

[TQ+] A. Tirumala, F. Qin, J. Dugan, J. Ferguson, and K. Gibbs. *Iperf: The TCP/UDP bandwidth measurement tool* (cit. on p. 96).

[VE03] A. Vahidi and A. Eskandarian. “Research advances in intelligent collision avoidance and adaptive cruise control”. In: *IEEE Transactions on Intelligent Transportation Systems* 4.3 (2003), pp. 143–153 (cit. on p. 36).

[Vid] VideoLAN. *x264 Project*. URL: `http://www.videolan.org/developers/x264.html`. Accessed May 15, 2015 (cit. on pp. 9, 30, 48).

[VQE03] VQEG. *Final report from the Video Quality Experts Group on the validation of objective models of video quality assessment*. 2003 (cit. on p. 26).

[VQE08] VQEG. *Final report from the video quality experts group on the validation of objective models of multimeda quality assessment, phase 1*. 2008 (cit. on p. 34).

[WB+04] Z. Wang, A.-C. Bovik, H.-R. Sheikh, and E.-P. Simoncelli. “Image quality assessment: from error visibility to structural similarity”. In: *IEEE Transactions on Image Processing* 13.4 (2004), pp. 600–612 (cit. on p. 27).

[WCK06] H. Wu, M. Claypool, and R. Kinicki. “Guidelines for Selecting Practical MPEG Group of Pictures”. In: *International Conference on Internet and Multimedia Systems and Applications (IASTED)*. Innsbruck, Austria, 2006 (cit. on p. 51).

[WDS09] H. Winner, B. Danner, and J. Steinle. “Adaptive Cruise Control”. In: *Handbuch Fahrerassistenzsysteme*. Ed. by H. Winner, S. Hakuli, and G. Wolf. Vieweg+Teubner, 2009, pp. 478–521 (cit. on pp. 36, 38).

[Weba] Netflix Website. URL: `http://www.netflix.com`. Accessed May 15, 2015 (cit. on p. 15).

[Webb] Periscope Website. URL: `http://www.periscope.tv`. Accessed May 15, 2015 (cit. on p. 1).

[Webc] YouTube Website. URL: `http://www.youtube.com`. Accessed May 15, 2015 (cit. on pp. 1, 15).

[Wen03] S. Wenger. "H.264/AVC over IP". In: *IEEE Transactions on Circuits and Systems for Video Technology* 13.7 (2003), pp. 645–656 (cit. on p. 13).

[Win15] S. Winkler. "Perceptual Video Quality Metrics - A Review". In: *Digital Video Image Quality and Perceptual Coding*. Ed. by H.-R. Wu and K.-R. Rao. Spinger, 2015 (cit. on p. 27).

[WM08] S. Winkler and P. Mohandas. "The Evolution of Video Quality Measurement: From PSNR to Hybrid Metrics". In: *IEEE Transactions on Broadcasting* 54.3 (2008), pp. 660–668 (cit. on p. 27).

[WMO09] Y. Wang, Z. Ma, and Y.-F. Ou. "Modeling rate and perceptual quality of scalable video as functions of quantization and frame rate and its application in scalable video adaptation". In: *International Packet Video Workshop (PV)*. Seattle, WA, USA, 2009 (cit. on pp. 3, 12, 28, 45, 60).

[WOZ02] Y. Wang, J. Ostermann, and Y. Q. Zhang. *Digital Video Processing and Communications*. Vol. 2002. Prentice Hall, 2002 (cit. on pp. 8–11, 15, 25).

[WS+03] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. "Overview of the H.264/AVC video coding standard". In: *IEEE Transactions on Circuits and Systems for Video Technology* 13.7 (2003), pp. 560–576 (cit. on pp. 9, 10, 13).

[WSB03] Z. Wang, H. Sheikh, and A. Bovik. "Objective video quality assessment". In: *The handbook of video databases: design and applications* (2003), pp. 1041–1078 (cit. on p. 27).

[WT06] J. Wietkze and M. T. Tran. *Automotive Embedded Systeme: Effizfientes Framework – Vom Design zur Implementierung*. Vol. 2006. Springer, 2006 (cit. on p. 41).

[YW98] M. Yuen and H.-R. Wu. "A Survey of Hybrid MC/DPCM/DCT Video Coding Distortions". In: *Signal Processing: Image Communication* 70.3 (1998), pp. 247–278 (cit. on p. 19).

[Zam09] A. Zambelli. *IIS Smooth Streaming Technical Overview*. Tech. rep. Microsoft, 2009 (cit. on pp. 19, 89).

[Zha14] F. Zhang. "Quality of Experience-driven Multi-Dimensional Video Adaptation". PhD thesis. Technische Universität München, 2014 (cit. on p. 20).

[ZL+14] C. Zhou, C.-W. Lin, X. Zhang, and Z. Guo. "A Control-Theoretic Approach to Rate Adaption for DASH Over Multiple Content Distribution Servers". In: *IEEE Transactions on Circuits and Systems for Video Technology* 24.4 (2014), pp. 681–694 (cit. on p. 88).