Poznań University of Technology
Faculty of Computing and Telecommunications
Institute of Multimedia Telecommunications

Doctoral Dissertation

Modeling of Codecs for 3-Dimensional Video

YASIR ABDULRAHEEM AHMED AL-OBAIDI

Supervisor: prof. dr hab. inż. Marek Domański
Auxiliary supervisor: dr. inż. Tomasz Grajek

Poznań, May 2023

Table of Contents

Abstract

The dissertation deals with encoder modeling for multiview plus depth video coding with applications to bitrate control. The dissertation concerns three generations of video encoders (AVC, HEVC, and VVC) thus searching for a general approach for encoder modeling valid for all three generations of video compression encoders. The study comprises both simulcast coding (like basic HEVC) and multiview coding (like MV-HEVC) thus providing also some useful results for monoscopic video encoders. A new approach to bitrate control and bitrate allocation for stereoscopic video plus depth is presented in the dissertation. The proposed approach to bitrate control depends on two models: the bitrate allocation model and the encoder model. This approach is optimized to produce an output bitrate of the video encoder equal to the required bitrate.

The dissertation presents an original unified approach applicable to encoder modeling in all the abovementioned scenarios. The approach consists of the application of the universal encoder model that demonstrates reasonable accuracy of approximation of bitrate as a function of the quantization step for transform coefficients of prediction residuals. Moreover, the problem of the bitrate allocation between a pair of stereoscopic views and the corresponding depth maps is considered in the context of maximization of virtual video quality for the given total bitrate for views and depths. As the results, in the dissertation, the entire original procedure is provided for bitrate estimation for a given quantization step for transform coefficients of prediction residuals. With the use of experimental data, it is also demonstrated the proposed approach is useful for bitrate control for stereoscopic video plus depth.

In practice, the values of the quantization steps are defined by a quantization scale parameter or quantization parameter *QP* for video. Similarly, a quantization parameter *QD* for depth is used. In order to derive the bitrate allocation model for stereoscopic video plus depth, the optimum *QP*-*QD* pairs are calculated. These optimum *QP*-*QD* pairs are the pairs of the quantization parameters for video (*QP*) and depth (*QD*) that achieve the best quality of the virtual view for a given bitrate for multiview video plus depth (MVD) sequences. Then, the bitrate allocation model is derived depending on optimum *QP*-*QD* pairs. The proposed models are used to estimate the quantization parameter for depth based on the quantization parameter for video components in multiview video and depth maps compression. The proposed models are compared to the models presented in the previous studies: the straightforward approach ($QP = QD$) and the approach presented by the ISO/IEC MPEG group. The efficiency of the proposed method for bitrate allocation between videos and depth maps is also presented in the dissertation.

In the study of the bitrate allocation issue, some sequences present unexpected and surprising behavior in some bitrate range, as an increase in the quality of the virtual views produced from decreasing the bit allocation for the depth component under the video bitrate constancy condition. Therefore the influence of depth map quality on virtual view quality is studied in the dissertation and the respective explanations of the phenomenon are given.

In the dissertation, the encoder model for stereoscopic video plus depth is derived. The proposed model is used to estimate the bitrate or frame size of stereoscopic video plus depth depending on the quantization step size for the video (R-Q model). The accuracy of the encoder

model is carefully studied and demonstrated by the results of extensive experiments with the respective test video sequences.

For the 2D video, this dissertation also presents a new method to compute rate control for HEVC and VVC based on AVC data. This method aims to calculate the proposed models' parameters for HEVC and VVC codecs based on the model parameters for the AVC codec to reduce the required time for bitrate estimation. The effectiveness of the proposed method of rate control for HEVC and VVC is studied in the dissertation.

In order to verify the accuracy of the proposed models to control the bitrate, the relative approximation errors are computed between the experimental data and approximate data calculated by the proposed models. The experiments are performed for a set of well-known and wildly accepted test sequences approved by ISO/IEC MPEG experts for the evaluation of new compression techniques. The results prove that the accuracy of the models is sufficient for bitrate control tasks.

Streszczenie

Rozprawa dotyczy modelowania kodeków dla zastosowań w sterowaniu prędkością bitową w kodowaniu wizji wielowidokowej wraz z mapami głębi. Rozprawa traktuje o trzech generacjach koderów wizyjnych (AVC, HEVC i VVC) i poszukuje się w niej ogólnego podejścia do modelowania koderów przydatnego dla wymienionych trzech generacji koderów wizyjnych. Studium obejmuje kodowanie wielowidokowe (jak za pomocą koderów MV-HEVC), a także kodowanie niezależne widoków i głębi (jak za pomocą podstawowych koderów HEVC), i w ten sposób prezentuje wyniki użyteczne także dla kodowania wizji monoskopowej. Rozprawa przedstawia nowe podejście do sterowania prędkością bitową i alokacji bitów pomiędzy widokami i mapami głębi. Zaproponowane podejście wykorzystuje dwa modele: model alokacji bitów i model kodera. Takie podejście jest zaplanowane tak, by mona było uzyskiwać założoną prędkość bitową na wyjściu kodera.

Rozprawa prezentuje ogólne podejście do modelowania koderów w wymienionych wyżej scenariuszach zastosowań. To podejście wykorzystuje uniwersalny model kodera, który zapewnia wystarczającą dokładność szacowania prędkości bitowej w zależności od kroku kwantowania współczynników transformaty błędów predykcji. Ponadto problem alokacji bitów pomiędzy widokami pary stereoskopowej oraz odpowiadającymi im mapami głębi jest rozpatrywany w kontekście maksymalizacji jakości syntetycznej wizji wirtualnej dla danej łącznej prędkości bitowej widoków i głębi. W ten sposób rozprawa przedstawia całą procedurę estymacji prędkości bitowej dla zadanego kroku kwantowania współczynników transformaty błędów predykcji. Z wykorzystaniem danych eksperymentalnych pokazuje się, że zaproponowane podejście jest użyteczne dla sterowania prędkością bitwą dla wizji stereoskopowej wraz z głębią.

W praktyce wartości kroków kwantowania są definiowane poprzez parametr skali kwantyzatorów albo parametr kwantyzacji *QP* dla wizji. Analogicznie wykorzystuje się parametr kwantyzacji *QD* dla głębi. Aby uzyskać model alokacji bitów dla stereoskopowych sekwencji wizyjnych z mapami głębi w pracy wyznacza się optymalne pary *QP*-*QD*. Optymalna para *QP*-*QD* to taka para wartości parametrów kwantyzacji dla sekwencji wizyjnych (*QP*) i map głębi (*QD*), dla której uzyskuje się najlepszą jakość widoku wirtualnego dla danej prędkości bitowej dla sekwencji wielowidokowych z mapami głębi (MVD). Następnie, na podstawie optymalnych par *QP*-*QD*, wyznacza się model podziału bitów pomiędzy widoki i mapy głębi. Zaproponowane w pracy modele służą do szacowania wartości parametru kwantyzacji dla map głębi na podstawie parametru kwantyzacji dla sekwencji wizyjnych. Zaproponowane modele są porównane z modelami zaprezentowanymi w literaturze: modelem prostym (*QP*=*QD*) oraz modelem zaprezentowanym przez grupę ekspertów MPEG afiliowaną przy ISO/IEC. W pracy oceniono także efektywność zaproponowanych modeli.

Podczas badań nad problemem alokacji bitów zaobserwowano, że niektóre sekwencje testowe wykazują nieoczekiwane i zaskakujące zachowanie w pewnym zakresie prędkości bitowych, jak np. poprawa jakości widoków wirtualnych uzyskana w wyniku zmniejszenia liczby bitów dla map głębi w trybie stałej prędkości bitowej. W związku z tym, w pracy

zbadano wpływ jakości map głębi na jakość widoków wirtualnych i przedstawiono analizę zjawiska.

W pracy zaproponowano także model kodera dla stereoskopowych sekwencji wizyjnych z mapami głębi. Model ten wykorzystywany jest do oszacowania prędkości bitowej lub rozmiaru ramki w zależności od kroku kwantyzacji dla sekwencji wizyjnej (model *R*-*Q*). Dokładność modelu kodera została zbadana w trakcie obszernych eksperymentów z sekwencjami testowymi.

Dla dwuwymiarowych sekwencji wizyjnych, niniejsza rozprawa przedstawia również nową metodę sterowania prędkością bitową dla koderów HEVC i VVC w oparciu o dane dla kodera AVC. Metoda ta ma na celu obliczenie parametrów proponowanych modeli dla kodeków HEVC i VVC w oparciu o parametry modelu dla kodeka AVC, co pozwala na znaczą redukcję czasu estymacji parametrów. Skuteczność proponowanej metody sterowania przepływnością dla HEVC i VVC została wykazana w rozprawie.

Aby zweryfikować dokładność proponowanych modeli do sterowania prędkością bitową, obliczono względny błąd aproksymacji pomiędzy danymi eksperymentalnymi a danymi oszacowanymi przez proponowane modele. Eksperymenty przeprowadzono dla zestawu dobrze znanych i powszechnie akceptowanych sekwencji wizyjnych zatwierdzonych przez ekspertów ISO/IEC MPEG do oceny nowych technik kompresji. Wyniki eksperymentów pokazują, że dokładność zaproponowanych modeli jest wystarczająca dla zadań związanych ze sterowaniem koderami wizyjnymi.

List of Abbreviations and Symbols

$\emptyset$	vector of parameters that depend on sequence content
B	number of bits per frame
D	Distortion
PSNR	Peak signal-to-noise ratio
Q	Quantization step size
QD	Quantization parameter for depth map
QP	Quantization parameter for video
R	Bitrate
λ	The slope of the R-D curve (Lagrange Multiplier)
ρ	The percentage of zeroes among quantized transform coefficients
Relative error(Q, $\emptyset$)	relative approximation error
2D	Two-dimensional
3D	Three-dimensional
3D-HEVC	Three-dimensional high efficiency video coding
AVC	Advanced Video Coding
bpp	Bit per pixel
CTC	Common Test Condition
CU	Coding Unit
FTV	Free-viewpoint television
GOP	Group of Pictures
HEVC	High Efficiency Video Coding
HM	High Efficiency Video Coding test model
HTM	MV- and 3D-HEVC test model
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
JCT-3V	Joint Collaborative Team on 3D Video Coding Extension Development
JCT-VC	Joint Collaborative Team on Video Coding
JM	Advanced Video Coding test model
MAD	Mean of absolute differences
MPEG	Moving Picture Experts Group
MVD	Multiview Video plus Depth
MV-HEVC	Multiview high efficiency video coding
R-D	Rate-Distortion
ROI	Region Of Interest
VSRS	View Synthesis Reference Software
VTM	Versatile Video Coding test model
VVC	Versatile Video Coding

Chapter One

Introduction

1.1 Scope of the Dissertation

Video transmission represents a massive portion of internet traffic. Nowadays, video traffic on the Internet constitutes about 80% of all internet traffic, according to [Cisc_18]. Most videos are recorded by a single camera, which means a single-view video is acquired. Over the past few years, multiview videos, which are recorded using many synchronized cameras around a scene, have gained more popularity. Therefore, the dissertation focuses mainly on issues related to the multiview video.

Virtual navigation [Smol_09, Doma_16a, Stan_18, Miel_20], virtual reality [Heid_19, Cao_20, Kune_20, Sett_22, Fuxi_23], and Free Viewpoint Television (FTV) [Tani_12c, Lee_15, Dzie_18a, Stan_18, Yan_22] are the most important applications of multiview video [Lafr_16]. The most commonly used representation for the mentioned above applications is multiview video and depth (MVD) representation [Mull_11]. As is well known, MVD representation uses depth information combined with a view for each viewpoint. A depth map contains information about the geometry of a scene. Also, a depth map is an image that includes information related to the distance between the camera and the objects in the scene. Fig. 1.1 presents an example of MVD representation for a *Ballet* sequence. MVD representation makes it possible to create a synthetic view, as seen by a virtual camera. Therefore, the number of views that are sent to the decoder can be significantly reduced (e.g., three views with corresponding depth maps are sent instead of all N views). Consequently, reduced throughput is required to send a multiview video.

In many practical applications, two views combined with the corresponding depth maps are used to produce a synthetic view [Doma_16c, Stan18]. Therefore, the idea of the dissertation is to use this use case (two views and two depth maps) for the studies on the compression of multiview video sequences. It is observed that such visual content is more advanced than just a stereoscopic video that consists of only two views. Additional depth information enables, for example, stereoscopic vision with adjustable depth. An advanced stereoscopic video system is presented in Figure 1.2.

The synthetic views, also known as virtual views, are produced on the decoder side by using view synthesis that depends on the available views and depth maps [Chan_07, Mull_11]. For the execution of view synthesis, the views and respective depth maps must be sent to the receiver. Due to the limited communication channel throughput, views and depth maps cannot be sent to the decoder uncompressed because of their huge data size. Consequently, many compression methods have been proposed to compress views and associated depth maps. One of the approaches involves special techniques that have been included in international standards, such as 3D-AVC [Chen_14], MVC [Vetro_11], MV-HEVC [Tech_16], and 3D-HEVC [Tech_16]. Special coding is joint coding for all views to exploit the redundancy between the views. Another method to compress MVD employs simulcast techniques (such as AVC [Wieg_03], HEVC [Sull_13], VVC [Bros_20], AV 1[Riza_18], AVS [Rao_14], etc.).

Simulcast coding of the multiview video plus depth is to encode, send, and decode each view and each depth map independently without exploiting inter-view redundancy and inter-component redundancy for video and depth. Simulcast coding is often used in applications such as MPEG Immersive Video (MIV) [Jung_20, MPEG_20] because it can be decoded using existing decoders.

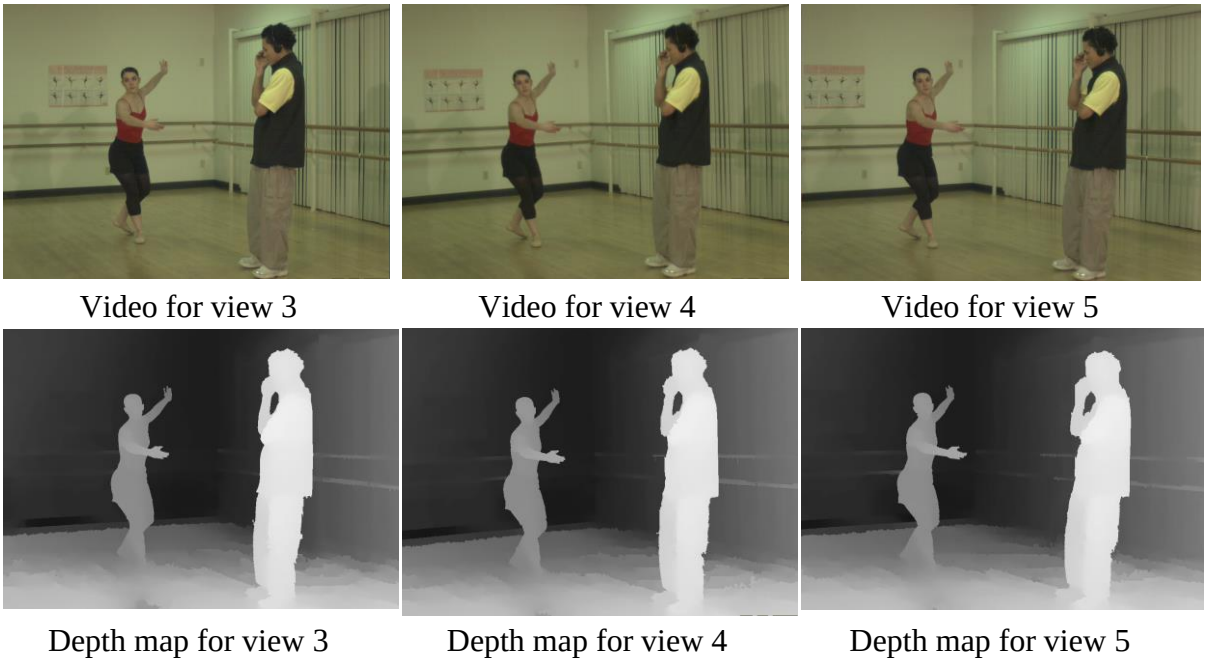


Fig. 1.1. An example of MVD representation for the *Ballet* sequence [Zitn_04].

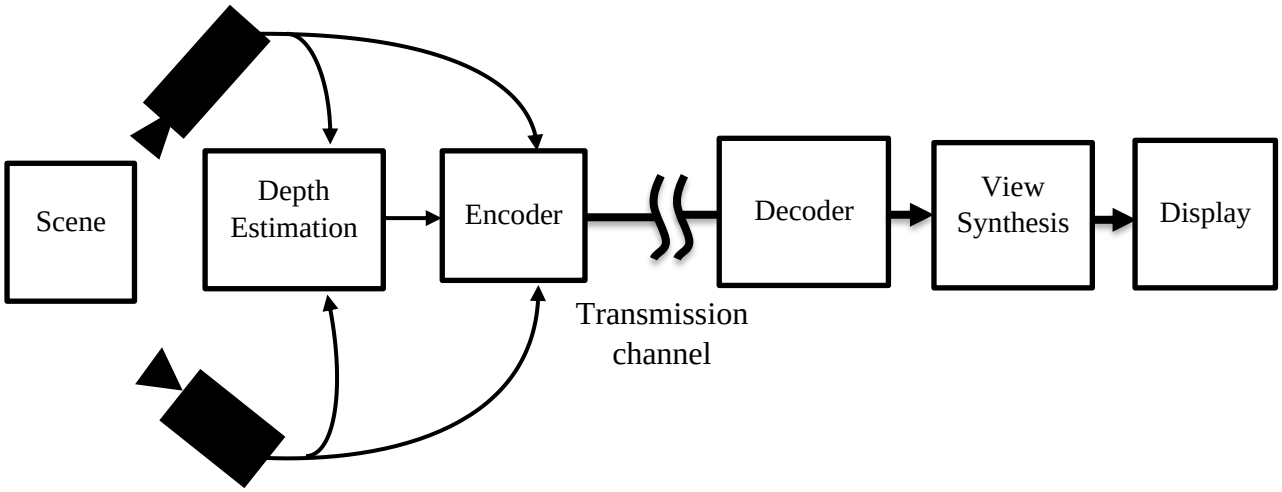


Fig. 1.2. An advanced stereoscopic video system.

Obviously, the changes in a scene's activities lead to changes in the output bitrate of the video encoder. Thus, rate control should be used to adjust the bitrate of the compressed video to meet the throughput limitation. Therefore, rate control is one of the essential tools to determine the total performance of the encoder. Two scenarios can be used for rate control. In the first scenario, the rate control algorithm selects the quantization step to obtain the maximum quality of the encoded video sequence at the assumed bit rate (constant bit rate, CBR) [Luo_05, Liu_14, Hyun_20, Li_20c]. In the second scenario, the rate control algorithm aims to maintain constant quality and minimize the total number of bits representing the video sequence

(variable bit rate, VBR) [Son_01, Anse_10, Lee_12, Guot_20, Zhou_23a]. Algorithms of rate control can be divided into two steps: codec control and bit allocation. Codec control aims to find the relationship between the required bitrate and the quantization step, while bit allocation is the distribution of bits at different levels of encoding (e.g., GOP level, frame level, region level) [Hou_10, Groi_11, Xu_16, Qin_19, Yann_20, Zhou_23b].

In modern encoders, the quantization step (Q) is the main parameter that allows for changing the bitstream of data at the output of the encoder. Therefore, Q is primarily used [Riba_99, Lim_07, Si_13, Bai_17, Cai_20] for bitrate control modeling in modern video encoders. In the encoders, the bitrate is controlled by the quantization step, as in the following formula:

$$R = f(Q), \tag{1.1}$$

where:

R - represents the bitrate,

Q - represents the quantization step size.

As is well known, the quality and bitrate of videos are controlled by using the quantization step in video coding. In two-dimensional video coding, a single Q value is used to obtain the best relationship between the quality of the encoded video sequence and the assumed bitrate. However, the relationship between the quality and bitrate is more complicated in MVD coding because of sending two components (views and the associated depth maps). Thus, two Q values are used to control the quality and bitrate for views, one Q value for videos and the other for depth maps. As the two views of the same scene need similar bitrate, the same quantization steps are assumed for the two views [Klim_14a]. Similarly, the same quantization step can be assumed for all depth maps [Stan_13a]. The bitrate allocation between videos and depth maps affects the compression efficiency, which is measured as the quality of the synthesized virtual views versus the total bitrate of the real videos and the corresponding depth maps transmitted. The virtual view quality depends on the quality of views and the quality of corresponding depth maps in the view synthesizing process [Bani_13, Wang_15]. The quality of the synthesized virtual view is measured by comparing the virtual view and the real view in the same position. Many methods have been proposed to measure the quality of the virtual view, such as PSNR [Salo_07] and IV-PSNR [Dzie_20]. Therefore, the important question is how to distribute the bitrate between videos and depth maps to obtain maximum quality at a required bitrate.

As a consequence, the aim of the dissertation is to establish rules for bitrate allocation and rate control for the multiview video plus depth map. Apart from that, the dissertation deals with the effect of depth map fidelity on the quality of the virtual view.

1.2 The Goal and the Thesis of the Dissertation

The goal is to model the video codecs in multiview-plus-depth applications. Therefore, the goal is to find the function $f(\cdot)$ from the formula (1.1)

$$R = f(Q).$$

These functions should be estimated for various standard video codecs. The relations between these functions for various codecs should be investigated in order to draw conclusions valid for the practical usage of such modeling. In particular, the problem of bitrate allocation between views and depth maps must be solved for practical purposes. Finally, the usefulness of the models should be demonstrated in practical bitrate control.

The thesis of the dissertation is as follows:

For MVD applications, there exist general $R = f(Q)$ models, where R is the bitrate or a number of bits, that are very similar for several standard video codecs. Similarity of the models implies that the data for a codec, e.g. AVC are usable to estimate bitrate produced by other codecs like HEVC or VVC.

The models may be simplified to have only one parameter that depends on content. Such models are useful for bitrate control.

1.3 Overview of the Dissertation

The dissertation is organized into nine chapters; they cover the theoretical aspects and the implementation of the theories involved in this research work.

In Chapter 1, the scope of the dissertation is described along with the introduction to a multiview video system. Also, the goal and purpose of the dissertation are presented.

Chapter 2 presents a literature survey for compression technologies, rate control, bitrate allocation for 2D and 3D sequences, and the impact of depth map quality on synthesized view quality. Moreover, the view synthesis methods are summarized.

The methodology of experiments is described in Chapter 3. Additionally, video quality assessment methods, the test sequences, and the video codecs used in the experiments are shown.

Chapter 4 deals with rate control for two-dimensional video. The author proposes a new method to calculate rate control for HEVC and VVC codecs based on AVC data. The results of the proposed method are compared to experimental data, and the results illustrate the effectiveness of the proposed method.

Chapter 5 deals with bitrate allocation for stereoscopic video plus depth. The author presented a new method to allocate bitrate between videos and depth maps to obtain the maximum of virtual view quality at a given bitrate. The results of the experiments show the efficiency of the proposed models compared to the reference approach and the methods shown in the literature.

The impact of depth map fidelity on virtual view quality is studied in Chapter 6.

In Chapter 7, the encoder model for stereoscopic video plus depth for many codecs is presented. Additionally, simplified models are presented to calculate rate control depending on the results of the proposed basic model. The results of all proposed models are compared to experimental data, and the results demonstrate the effectiveness of these proposed models.

Chapter 8 presents a method of bitrate control for stereoscopic video plus depth for many compression techniques. Target data are compared to the results of the proposed method, and the comparison results show the efficiency and accuracy of the proposed method.

Finally, a summary of the presented dissertation in Chapter 9 is given. This chapter offers the primary and secondary achievements of the dissertation, a discussion of the results, and future works.

Chapter Two

Literature Survey

2.1 Compression Technology

Video compression plays an important part in communication systems by sending video data with the lowest possible number of bits while preserving quality [Beac_08, Gao_14, Shi_19]. Thus, the compression process will result in reducing the bandwidth for the communication channel/storage memory space to transmit/store the video.

Any video coding can be characterized by:

1. Bitrate,
2. Quality of the reconstructed video,
3. Encoding/ decoding delay,
4. Encoder/decoder complexity.

In this dissertation, we focus on the first two aspects, as we deal with algorithmic features whereas the other two aspects are more related to specific configurations and implementations of encoders mostly.

Many video codecs have been proposed to encode/decode the video. For practical use, the video coding technology has to be standardized, such as AVC [Rich_03a, Wieg_03], HEVC [Sull_13, Sze_14, Wien_15], VVC [Sull_18, Bros_20], AV-1 [Riza_18], and AVS [Zhu_13, Rao_14, AVS_15, Choi_20]. Figures 2.1 and 2.2 describe the encoder and decoder process for modern video compression methods. These video coding standards are categorized as so-called hybrid video codecs because they use various tools of eliminating spatial and temporal redundancy. Also, these codecs use the tradeoff between the bitrate and the reconstructed video quality depending on the quantization step. There is an inverse relationship between the quantization step (Q) and the bitrate/quality of the reconstructed video, e.g., increasing Q leads to a decrease in bitrate and quality.

Due to the limited usage of AVS in certain applications in some parts of China, AVS will not be considered in the dissertation. Also, AV-1 is a relatively new technology (2018), but its efficiency is not much different from HEVC, and it will not be considered in the dissertation. Therefore, the dissertation focuses only on AVC, HEVC, and VVC which are internationally standardized according to the documents [AVC_std, HEVC_std, VVC_std], respectively.

As is well known, VVC outperforms HEVC, and HEVC outperforms AVC by roughly halving the bitrate and preserving, at the same time, the subjective quality of the decoded video [Mans_20, Siqu_20, Bros_21]. This improvement of the rate-distortion performance is obtained at the cost of significantly increased complexity. For example, a VVC encoder is about 5-10 times more complex than an AVC encoder [Topi_19].

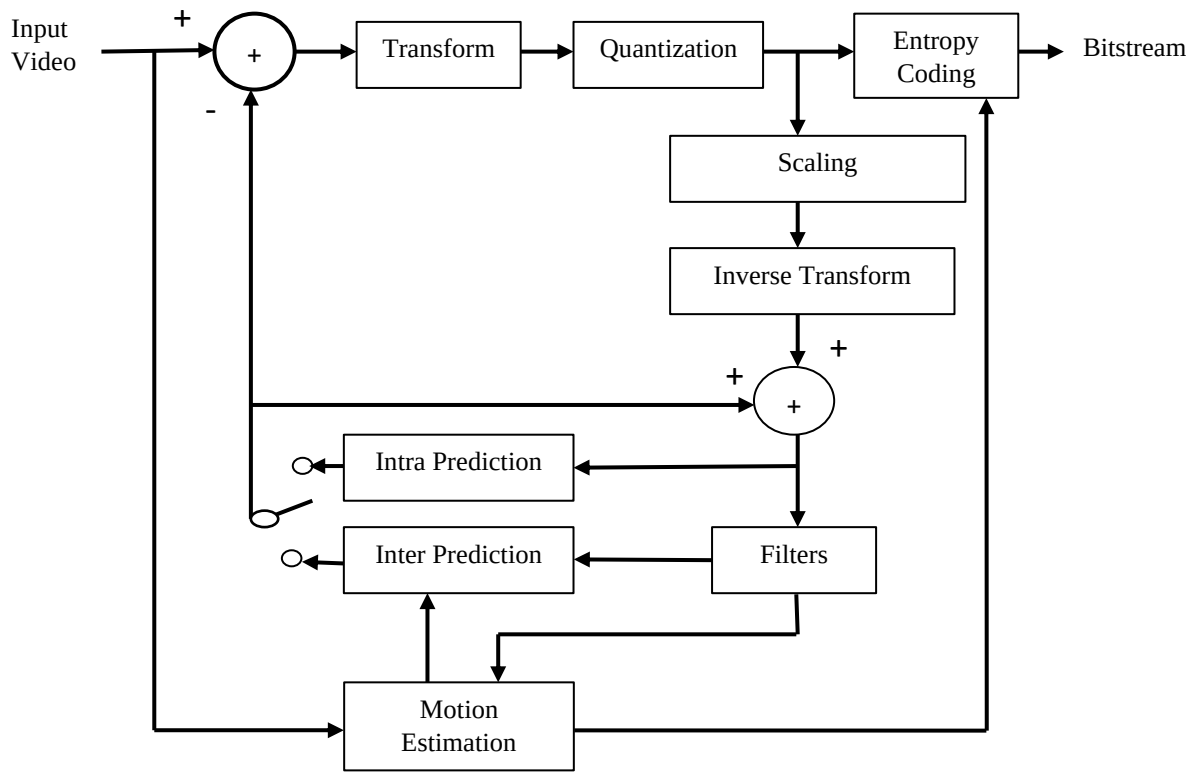


Fig. 2.1. The general structure of a video encoder.

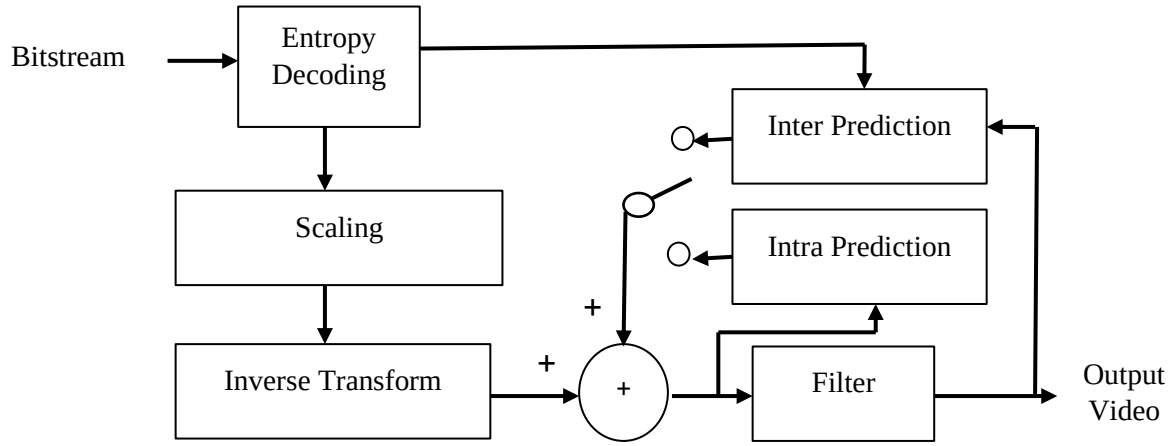


Fig. 2.2. The general structure of a video decoder.

The standard video codecs (e.g., AVC, HEVC, VVC) produce various bitrates and quality levels of the reconstructed video by using the same quantization steps, due to using different tools in these codecs. For example, in intra prediction, AVC uses fewer prediction directions than HEVC and VVC (AVC uses 9 prediction directions, while VVC and HEVC use over 65 and 35 directions, respectively). This approach provides very high compression efficiency of intraframe compression of HEVC and VVC codecs by improving the possible reference blocks

for the block matching algorithm. Inter prediction estimates the relationship between neighboring frames by using motion compensation prediction (MCP) to remove temporal redundancy, hence, enabling higher compression rates [Flie_02]. Inter prediction is one of the most complex and time-consuming processes of video encoding. VVC uses affine compensation prediction [Meue_20, Jin_22, Muno_23] in inter prediction, while AVC and HEVC use translational compensation prediction [Rich_03b, Haro_10, Yu_13, Wien_15, Sair_22]. The affine model can describe most of the motions (translation, rotation, and zoom) in video sequences, and this model employs two or three motion vectors to allow movement using four or six degrees of freedom (DOF) for a block [Ghaz_19]. In contrast, the translational model can only describe translational motion by a single vector [Wedi_03, Ugur_13]. Therefore, inter-frame prediction for VVC is more accurate but also more computation-costly than AVC and HEVC. HEVC and VVC use more transforms than AVC (AVC applies a discrete cosine transform (DCT) while HEVC and VVC use DCT as well as a discrete sine transform (DST)). In quantization, VVC uses a wider range in Q than AVC and HEVC (e.g., AVC uses Q from 0.625 to 224, and HEVC uses Q from about 0.63 to 228.07, while VVC uses Q from about 0.63 to 912.28). HEVC and VVC use the same coding in the entropy coding section, unlike AVC (VVC and HEVC use only Context Adaptive Binary Arithmetic Coding (CABAC) while AVC employs CABAC and CAVLC (Context Adaptive Variable Length Coding)). In the filter section, VVC and HEVC employ more filters than AVC (e.g., VVC and HEVC apply a deblocking filter and sample adaptive offset (SAO) filter, while AVC only uses the deblocking filter).

In AVC, HEVC, and VVC [Rich_03b, Ueda_07, Sjob_12, Wien_15, Chen_19, Weng_19, Wang_21], the coded bitstream can be divided into a series of Network Abstraction Layer (NAL) units. Fig. 2.3 presents an example of a bitstream structure. Each NAL unit includes a NAL header and raw byte sequence payload (RBSP). RBSP may be a video parameter set (VPS), sequence parameter set (SPS), picture parameter set (PPS), or encoded slice. The parameters (VPS, SPS, and PPS) contain general video parameters. These parameters give a robust mechanism for transporting data that are necessary for the decoding process. These parameters can be either a part of a bitstream or can be stored separately. The encoded slice includes a slice header and a series of blocks; these blocks are called coding tree units (CTUs) in HEVC and VVC, while in AVC they are called macroblocks (MBs). In HEVC and VVC, the blocks can be divided into coding units (CU) and corresponding prediction units (PU), and transform units (TU). In picture partitioning, VVC employs block size larger than AVC and HEVC (e.g., VVC uses a code tree unit of up to 256×256 pixels with a more variable sub-partition structure, whereas AVC and HEVC utilize blocks of up to 16×16 and 64×64 pixels, respectively).

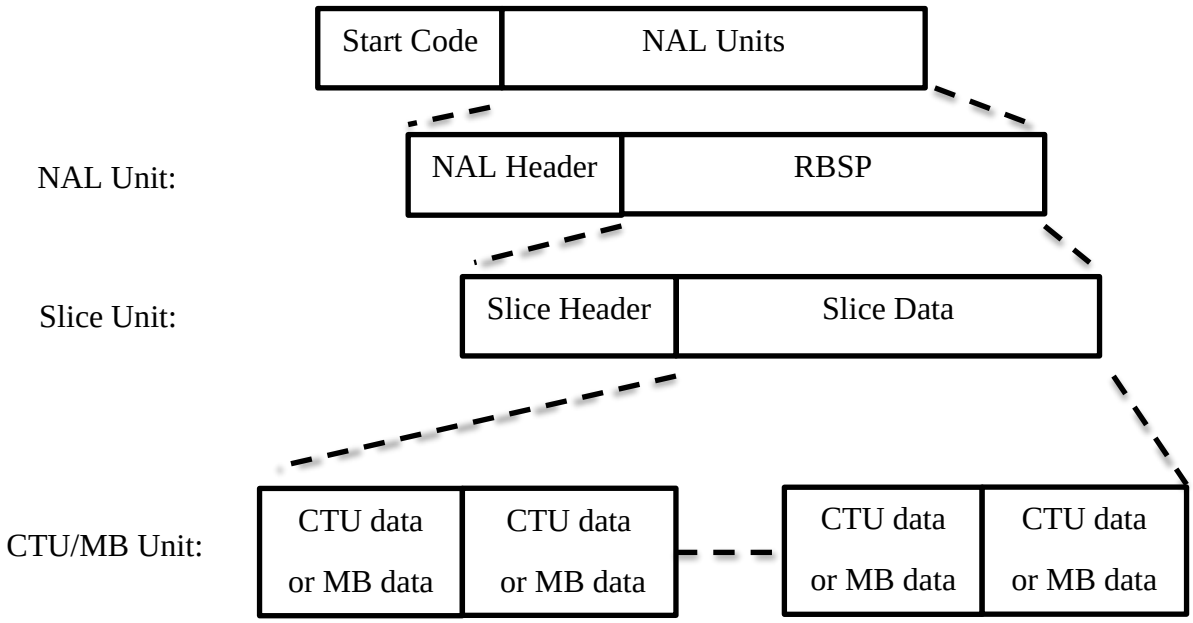


Fig. 2.3. Bitstream structure for byte stream format.

In AVC, HEVC, and VVC, a picture is coded as one or more slices. Each slice can be decoded individually from the other slices in the same picture, which means no prediction is performed from one slice to the other. There are different types of encoded slices, e.g. AVC uses five types of the encoded slice: I, P, B, SP (Switching P), and SI (Switching I) [Rich_03a], while HEVC and VVC use three types: I, P, and B. In an I slice, MBs or CTUs are coded by using only intra prediction. A slice of type P may contain intra prediction or inter prediction. In a P slice, MBs or CTUs can be coded from a single reference picture in a reference picture list with a single motion vector per prediction partition. A B slice may contain intra prediction or inter prediction. MBs or CTUs in a B slice can be coded from one or two reference pictures in two reference picture lists with one or two motion vectors per prediction partition.

In Fig. 2.4, an example of R-D curves for different codecs (AVC, HEVC, and VVC) for the *BQTerrace-1920x1080* sequence is shown. R-D curves in Fig 2.4 have been obtained using reference software for HEVC, VVC, MV-HEVC, and 3D-HEVC, as shown in Section 3.3.

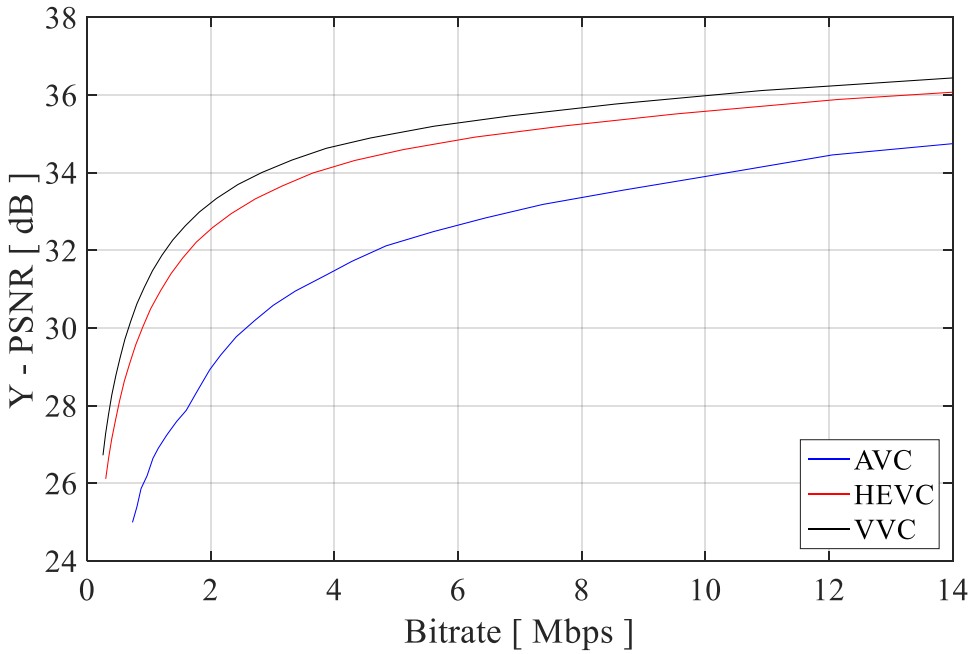


Fig. 2.4. An example of R-D curves for different codecs [AVC, HEVC, VVC] for the *BQTerrace- 1920x1080* test video sequence [the results by the author].

Based on Fig. 2.4, the fact that VVC outperforms both HEVC and AVC is confirmed because VVC uses more advanced video coding techniques than AVC and HEVC, as mentioned previously. Additionally, it has been observed that the shapes of R-D curves for AVC, HEVC, and VVC codecs are approximately similar.

2.2 Rate Control

The goal of video compression is to produce the optimum video quality i.e. to minimize distortion under certain requirements, such as channel throughput or storage limitations. Therefore, video compression plays a very significant role in applications that require the transmission and storage of video. Video coding systems perform compression by reducing redundancies, especially spatial and temporal ones. As known, the amount of redundancy in sequences is variable; therefore, video encoders produce output streams of variable bitrate, mainly due to the changes in the activities in the sequences. In the case of using only intra-frame coding, the number of bits spent by each frame will vary due to the scene complexity. Complicated scenes require a much larger number of bits than simple scenes. Inter-frame coding is another factor in changing the bitrate of the compressed video. In inter-frame coding, the encoded data includes motion vectors and residual coefficients. When the scene contains only small and simple movements (such as the translational motion of rigid objects), block-based motion estimation can be effective in predicting the movement. Consequently, the motion vector has a relatively high portion of the number of bits. If the scene contains complex motion (such as rotation, zoom, etc.), block-based motion estimation has difficulty predicting the movement. Thus, the motion vectors constitute a lesser portion of the number of bits [Rich_03a, Sze_14, Sald_22].

In video coding, the frame coding type is another reason that impacts the number of bits produced by an encoder. I-frame only employs intra-frame prediction, so the rate of bits is usually very high, which means the compression ratio is meager. P-frame employs inter-frame prediction (unidirectional), and its compression efficiency is normally higher than the I-frame. B-frame utilizes more efficient bi-directional inter-frame prediction; therefore, the compression ratio is very high [Lee_13, Pate_15, Wien_15, Taja_17, Fili_19, Hsie_20, Jin_23].

As the channel throughput or storage capacity is restricted, all bitrate variations should be well controlled before transmission/storage. Several networks and storage memories are working at a constant bit rate (CBR). Even if they operate at a variable bit rate (VBR), the maximum stream rate fluctuations will have corresponding constraints. Consequently, the compressed video should be adjusted to meet the channel throughput and storage memory space requirements [He_08, Tian_18, Gong_19, Guo_20, Guot_20, Li_20c, Zhou_23a].

Rate control is concerned with budget-constrained bit-allocation issues that aim to decide the number of bits to be utilized in various parts of the video in order to maximize the quality of the decoded frames. A common approach to deal with these problems is to look at the R-D (rate-distortion) trade-offs in bit allocation. As a consequence, a video encoder uses rate control as a method of organizing the changing bitrate of the encoded bitstream to produce high-quality decoded frames at a required bitrate.

In video coding, rate control is executed by using a set of steps. The first step is to update the required average bitrate for each short time interval. The next step is to determine the frame coding type (I -, P-, or B-frame) and the bit budget needed for each frame to be encoded. The last step is to determine the coding type and *Q* for each block in a frame to meet the required rate for this frame [Rama_17, Cao_18].

2.3 Encoder Modeling

Rate control is usually adopted in the video coding system to accomplish a required bit rate by adjusting the quantization parameter (*QP*) which integer values correspond to some quantized values of the quantization step. The relationships between the quantization parameter and the equivalent quantization step size for the transform coefficient are as follows:

1. For HEVC [Sze_14] and VVC [VVC],

$$Q = f(QP) = (2^{1/6})^{QP-4}\,, \tag{2.1}$$

where:

QP - quantization parameter,

Q - quantization step size.

From equation (2.1), the quantization step size equals 1 for the quantization parameter equal to 4.

2. For AVC, Table 2.1 shows the values of *Q* and the corresponding values of *QP*.

Table 2.1: The width of quantization intervals *Q* and the corresponding values of *QP* in the AVC codec [Rich_03a].

QP	0	1	2	3	4	5	....	46	47	48	49	50	51
Q	0.625	0.6875	0.8125	0.875	1	1.125		128	144	160	176	208	224

In AVC, the quantization step size doubles in size for every increment of 6 in the quantization parameter [Rich_03a].

Fig. 2.5 presents the relationship between the quantization parameter and the equivalent quantization step size for AVC, HEVC, and VVC codecs.

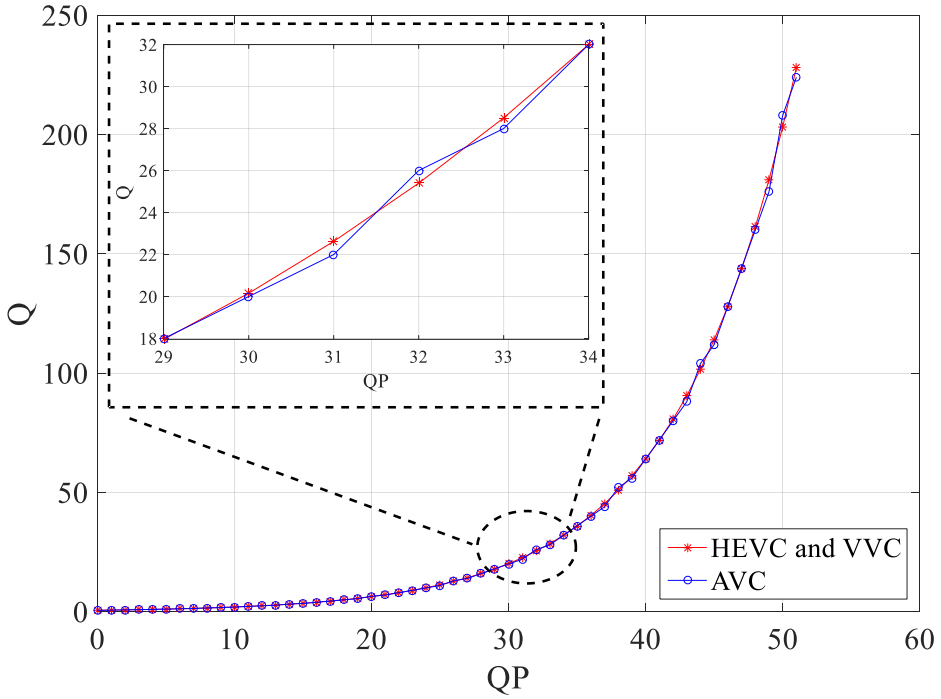


Fig. 2.5. Quantization step (*Q*) as a function of the quantization parameter (*QP*).

Once the bit budget is allocated, the following step is to estimate the encoding parameters that enable the required bitrate to be reached. Changing *Q* is the first approach to reach a required bitrate, which leads to modeling functions linking bitrate and quantization step (R-Q) [Chia_96, Chia_97]. The R-Q model can be very complex to derive. Therefore, other models have been proposed, such as ρ-domain (R-ρ) [Kim_01, He_02, He_08]. The R-ρ model consists of passing through an intermediate and more simple linear model to avoid the R-Q model. The λ-R model was used to control the bitrate by finding the relationship between the bitrate and the Lagrangian multiplier (λ) (λ is the slope of the R-D (rate-distortion) curve) in [Li_12, Li_20a, Li_20b, Yang_20, Chen_22]. In [Gao_19], a learning-based initial *QP* method was presented to replace the traditional calculative method, which works to improve the performance of rate control. A proposed neural network-based approach was presented in e.g. [Li_17, Kian_20] to control the bitrate by estimating the model parameters to improve the efficiency.

2.3.1 R-Q Approach

In [Choi_13, Tan_17, Cao_18, Mao_22], an R-Q model-based rate control scheme was proposed. The R-Q model depends on the relationship between the bitrate and the quantization step. The R-Q relationship is based on two main models: the quadratic R-Q model and the simple R-Q model.

For the quadratic R-Q model, which was used with AVC [Hu_10, Xiao_11] and HEVC [Lian_13, Wu_15] coding, the relationship between *Q* and *B* (the number of bits) was written as:

$$B(Q) = a \cdot \frac{MAD}{Q} + b \cdot \frac{MAD}{Q^2}, \tag{2.2}$$

where:

a and *b* - model parameters that depend on the video content,

MAD - mean absolute difference between the reconstructed and the original image,

Q - quantization step size,

B - number of bits per frame.

In a simple R-Q model, used with AVC [Dong_07c, Dong_09, Lu_14] and HEVC [Cen_14, Wang_18] coding, the simple relationship between *Q* and *B* was expressed as:

$$B(Q) = \frac{\varphi_1}{Q} + \varphi_2 \tag{2.3}$$

where $\varphi_1$and φ_2 model parameters that depend on the video content and can be determined by using a linear regression method [Wei_05, Yan_09] between R-Q points.

The authors in [Graj_10] proposed a model to find the relationship between *B* and *Q*; the following relationship was proposed:

$$B(Q) = \frac{a}{Q^b + c}, \tag{2.4}$$

where:

B - number of video bits per frame,

$a, b, and\ c$ - model parameters that depend on sequence content,

Q - quantization step size.

A comparison between the quadratic R-Q model and the simple R-Q model was presented in [Dong_07a]. The authors in [Dong_07a, Dong_07b] found the quadratic R-Q model consistently gives smaller errors, and thus, it is more accurate than the simple R-Q model. As is well known, quadratic R-Q models are based on complex functions; thus, it has been found that few applications are using the quadratic R-Q models because of their high computational complexity [Chia_97, Dong_07b]. Due to the computational complexity of the quadratic R-Q

model and the low accuracy of the simple R-Q model, the proposed model was presented in [Graj_10]. The authors in [Graj_10] declared that model (2.4) mostly fits very well with the experimental data in a wide range of bitrates. Besides, the authors in [Graj_10] claimed that the proposed model (2.4) outperforms the model from the reference implementation of an MPEG-4 AVC/H.264 encoder. Consequently, model (2.4) will be used in this dissertation to control the bitrate for sequences.

2.3.2 R-ρ Approach

The linear model was applied in MPEG-2 [Lee_98, Kim_01], AVC [Lim_07, Zhan_11], and AVS-1 [Wang_09], and a linear relationship between bitrate and ρ was assumed, where ρ is the percentage of zeroes among quantized transform coefficients. It is observed that the value of ρ increases with increasing Q; therefore, it is possible to find an individual mapping of the value of ρ on Q, and so the required bitrate to represent the compressed image in function ρ is expressed. Generally, the R-ρ model was described in the following equation:

$$R_{TC}(\rho) = \theta \cdot (1 - \rho), \tag{2.5}$$

where,

R_{TC} - bitrate for transform coefficients,

θ - a model parameter related to the video content,

ρ - a percentage of zeroes among quantized transform coefficients.

From equation (2.5), R_{TC} depends on θ and ρ. According to [Ser_11], θ is constant. Therefore, the value of ρ can be determined according to the target bitrate. In [He_02, Wang_13a, Wang_13b], a mapping scheme between ρ and Q was presented to determine an appropriate Q to meet the target bitrate.

Modern video coding standards (HEVC and VVC) present the skip prediction method and a flexible quad-tree coding unit partition scheme, leading to a significant difference in the distribution of zeros after transformation and quantization. Besides, the linear relationship between ρ and R_{TC}, assumed, is inaccurate. As a result, the linear ρ-R model is rarely used in modern video coding standards (e.g., HEVC and VVC). Because of this problem, the proposed approach was presented to convert the ρ-R model to the R-Q model in [Wang_13a, Wang_13b] as follows:

$$R_{TC} = \theta \cdot (1 - \rho) = \theta \cdot N_{non\_zero}, \tag{2.6}$$

$$N_{non\_zero} = N + a \cdot Q + b \cdot Q^2, \tag{2.7}$$

where,

N_{non_zero} - number of nonzero transform coefficients,

N - total number of the frame,

Q - quantization step size,

a and *b* - model parameters that depend on the video content, obtained via the linear regression scheme.

Finally, the proposed model in [Wang_13a, Wang_13b] can be obtained by

$$R_{TC} = \theta \cdot N_{non\_zero} = \theta \cdot (N + a \cdot Q^2 + b \cdot Q). \quad (2.8)$$

From the previous works [Wang_13a, Wang_13b], it is noticed that the ρ-R model cannot be used with modern video coding standards due to its inaccuracy in determining the bit rate.

2.3.3 R-λ Approach

Many R-λ models have been proposed to find the relationship between bitrate and λ, where λ represents the slope of the R-D (rate-distortion) curve. The R-λ model was used with AVC [Hu_12, Li_14a], HEVC [Li_14b, Cord_16, Wang_16, Lei_18, Abol_19, Li_19] and VVC [Chen_20, Li_20c, Zhao_22] to calculate the bitrate control. In equation (2.9), the relationship between distortion (*D*) and bitrate (*R*) was modeled as a hyperbolic function.

$$D(R) = C \cdot R^{-K}, \quad (2.9)$$

where *C* and *K* are model parameters relevant to the video content (these parameters are estimated based on the video content). As is well known, λ is the slope of the R-D curve, which can be calculated by

$$\lambda = -\frac{\partial D}{\partial R}. \quad (2.10)$$

Then, a relationship between λ and bits per pixel (*bpp*) is shown in

$$\lambda = \alpha \cdot bpp^{\beta}, \quad (2.11)$$

where α and β are content-related parameters. This approach depends on the accurate approximation of the Lagrange multiplier (λ). Once the λ for the required bitrate is determined, all the coding parameters, including the quantization parameter (*QP*), can be chosen by the Rate-distortion optimization process [Sull_98]. In [Chiu_12, Li_12], it was observed that the relationship between *QP* and *ln(λ)* is a linear one, regardless of the coding level and coding structure. Therefore, the relationship between *QP* and *ln(λ)* is:

$$QP = a_0 \cdot \ln(\lambda) + b_0, \quad (2.12)$$

where a_0 and b_0 are constant parameters that are estimated by using the least-squares fitting to the optimum *QP-ln(λ)* pairs for sequences.

2.4 Bit Allocation

Bit allocation is one of the primary issues in video coding. The basic idea of bit allocation is to reduce overall distortions within a required bitrate. Bit allocation can be divided into three levels: GOP level, frame level, and block level. Much research was presented to calculate the bit allocation [Rama_17].

A video sequence is divided into groups of pictures (GOPs). A GOP can contain the following frame types: I frame, P frame, and B frame. As is known, GOP-level bit allocation is the first step of rate control and has a significant effect on rate control performance [Song_17, Wang_20a, Zhou_23b]. The GOP-level rate control includes determining the total number of bits for each GOP. In [Sanz_13, Cheng_19], GOP-level rate control algorithms for video coding were presented. In [Lian_13, Cao_18], R-Q model-based rate control for GOP-level was proposed. Also, GOP-level rate control with the R-λ model was introduced in [Tang_19, Zhan_19].

Many algorithms have been proposed for frame-level rate control [Guo_17, Chen_18, Guo_19, Liu_22]. The aim of frame-level rate control for video coding is to obtain approximately constant frame quality along with the time when each frame should get a portion of the target GOP number of bits proportional to frame type complexity. The authors in [Lin_08, Zhan_11] proposed ρ-domain rate-frame based rate control. In [Lu_14, Wang_18], frame-level rate control with the R-Q model was suggested. λ domain rate control based frame-level was presented in [Li_18b, Sanc_18].

Many block-level rate control algorithms for video coding have been presented in [Medd_14, Medd_15, Shen_16, Zhan_17, Lei_18, Yan_20a, Yan_20b, Liu_21, Lin_22, Wang_22]. Block-level rate control with the R-Q model was introduced in [Shi_08, Wu_14]. In [Yang_14, Maun_16], R- λ rate control based on rate-block was proposed.

Conclusions: the GOP-level bit allocation approach takes into consideration the number and types of frames being encoded, along with any error in encoding previous GOP sequences [Wu_11]. In contrast, the frame-level bit allocation approach takes into account separate models to estimate parameters according to the picture position in hierarchical GOP. The block-level bit allocation approach consists of considering each block in a specific picture as a rival component of participation in the available frame-level budget.

2.5 Immersive Video

Immersive videos have gained great popularity recently, because they give the audience a new viewing experience by deeply involving them in the content, meaning the ability to absorb the user entirely into a visual scene. Immersive videos may be linked to both original and computer-generated content. Also, immersive video content is occasionally described as high-realistic [Doma_17, Wien_19, Vada_22].

Immersive videos are video recordings where a scene is recorded in all directions at the same time. They are usually shot using an omnidirectional camera (also known as a 360°

camera), or a collection of separated cameras, which are connected and mounted in a spherical array. In the immersive video, a head-mounted display (HMD) can be used to give users the option to select their field and direction of view by head movement, to simulate a real-world viewing experience [Jeon_19, Jung_22].

2.6 Multiview Video Representations

Many ways of representation are used to recreate a scene. The type of representation depends on the content that camera systems can produce on their own or by processing original data. Many approaches have been proposed to represent three-dimensional scenes, like image-based representation, geometry-based representations, and intermediate representations [Ozak_07, Smol_09].

Image-based representations usually require a large number of cameras to achieve good-performance rendering [Smol_09]. Views are generated using interpolation from the available camera views without using any geometrical model. The main advantage of image-based representations is the possible high quality of virtual view synthesis, avoiding any three-dimensional scene reconstructions. In image-based representation, complexity is a primary problem, as the quantity of data to be processed is huge. Examples of this kind are ray-space and light-field [Fuji_94, Ozak_07, Tani_12a].

Geometry-based representations may require a lower number of cameras but rely on complicated image processing algorithms (e.g., geometry estimation algorithms, etc.). View generation is usually expensive and requires human assistance. Examples of this approach are point-cloud and 3D meshes [Ozak_07, Nata_11]. This approach is used in movies, computer games, etc.

Another approach is called intermediate representations, which include both a geometrical model and standard camera views. Multiview Video plus Depth (MVD) representation is one example of intermediate representations [Mull_11, Tani_12c].

Some ways to represent multiview video are presented in this section as multiview video plus depth, point cloud, and ray-space.

2.6.1 Multiview Plus Depth

Multiview plus depth (MVD) is the most commonly used representation for applications like Free Viewpoint Television (FTV) [Tani_12c, Stan_18] and 3-dimensional Television (3DTV) [Kubo_07] [Ozak_07] [Lafr_16]. MVD includes a number of views – very diversified, sometimes less than 10, sometimes more than 100 – and depth maps. In Fig. 2.6, an example of a multiview video plus depth maps is shown. A depth map represents the geometric information about a scene. Depth maps can be estimated by using algorithms of depth estimation [Seno_15, Du_19, Miel_20, Schr_22].

Due to limited communication channel throughput, the transmission of all videos from cameras to the decoder is not possible, because it requires too high bitrate – even with the use of the latest technology. MVD provides the ability to produce a synthesized view as viewed by a virtual camera. Such a virtual camera can be put in the position of another real camera or an arbitrary position. Thus, MVD requires sending a limited number of views and depth maps to the decoder. The remaining (unsent) views will be produced at the decoder side based on transmitted views [Puri_16, Ceul_18]. The views which are produced at the decoder side, are called virtual views. The virtual views can be generated by view synthesis [Dzie_16, Li_18a, Nam_22]. Thus, the generation of virtual views permits users to alter the viewpoint within the scene freely [Tani_12c].

Because the required data of multiview video plus depth video is huge, many methods have been presented to compress multiview video plus depth maps, and it will be mentioned in Section 2.7.

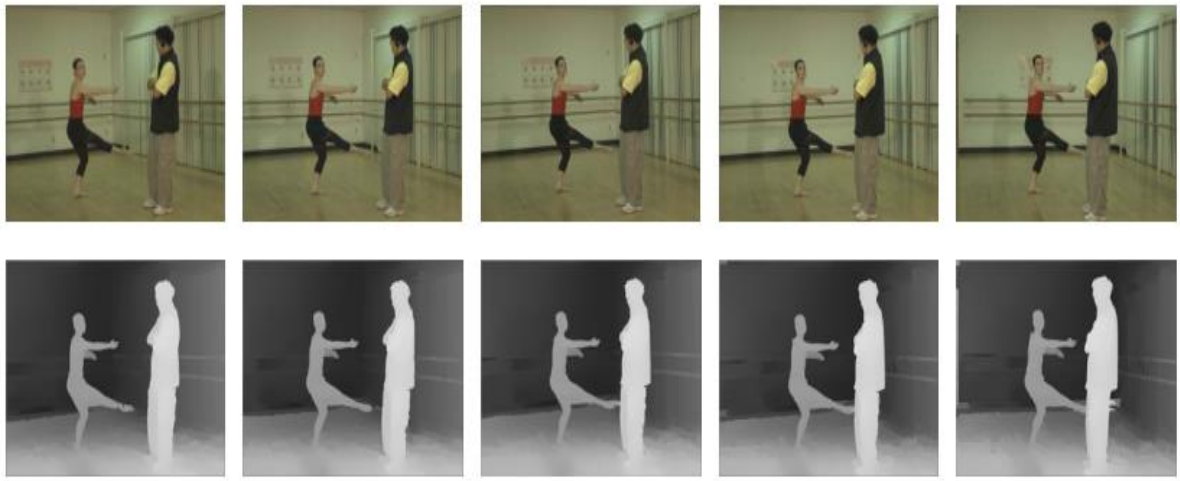


Fig. 2.6. An example of multiview video plus depth maps for the *Ballet* test sequence.

2.6.2 Point Cloud

Point clouds are groups of points representing 3D objects, as shown in Fig. 2.7. A point cloud consists of a set of coordinates indicating the location of each point, along with one or more attributes, such as color, transparency, and material properties associated with each point. Three-dimensional point clouds can be captured utilizing many cameras and depth sensors in different setups. Also, point clouds are used to create 3D meshes and other models used in 3D modeling for various fields, including architecture, geographic information systems, medical imaging, manufacturing, 3D gaming, and various virtual reality (VR) applications [Lins_01, Ozak_07, Schw_19, Fran_22].

Point clouds are distinguished by simplicity and versatility. Point clouds are flexible to noise, as there are no suppositions on the structure, such as the smooth manifold assumption needed for meshes [Lins_01]. Also, this approach does not require preprocessing and thus is suited to real-time applications. However, point clouds are disorganized and there is no

correspondence or correlation between frames. This problem creates a challenge to exploiting temporal redundancies for compression. Point clouds often consist of millions to billions of points that require significant storage space and/or transmission bandwidth. Therefore, the compression of data is important in point cloud-based applications [Tulv_16]. A point cloud can be compressed in two ways [Graz_20]. The first way is called video-based point cloud compression (V-PCC) which consists of projecting the three-dimensional space onto a set of two-dimensional patches and encoding them using 2D video techniques. The second way (geometry-based point cloud compression (G-PCC)) is to cross the three-dimensional space directly to generate the predictors [Schw_19].

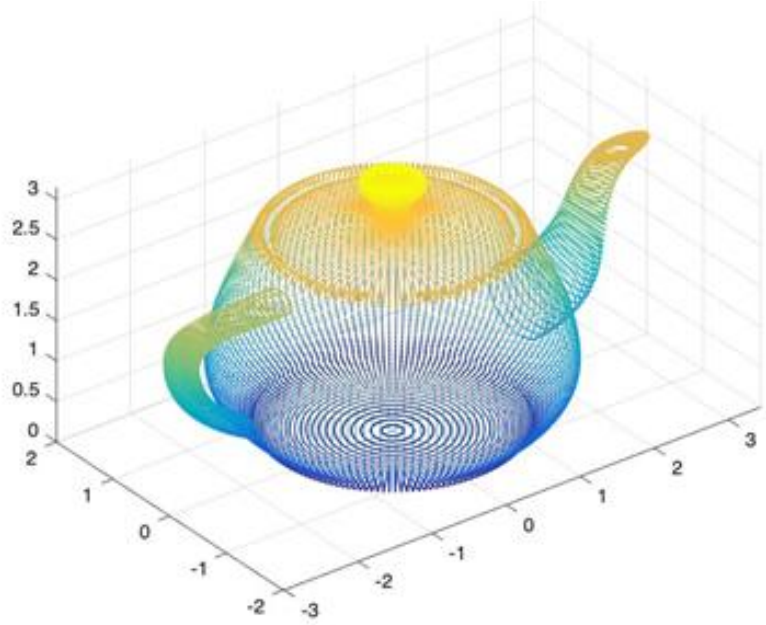


Fig. 2.7. An example of point cloud representation with MATLAB.

2.6.3 Ray-Space

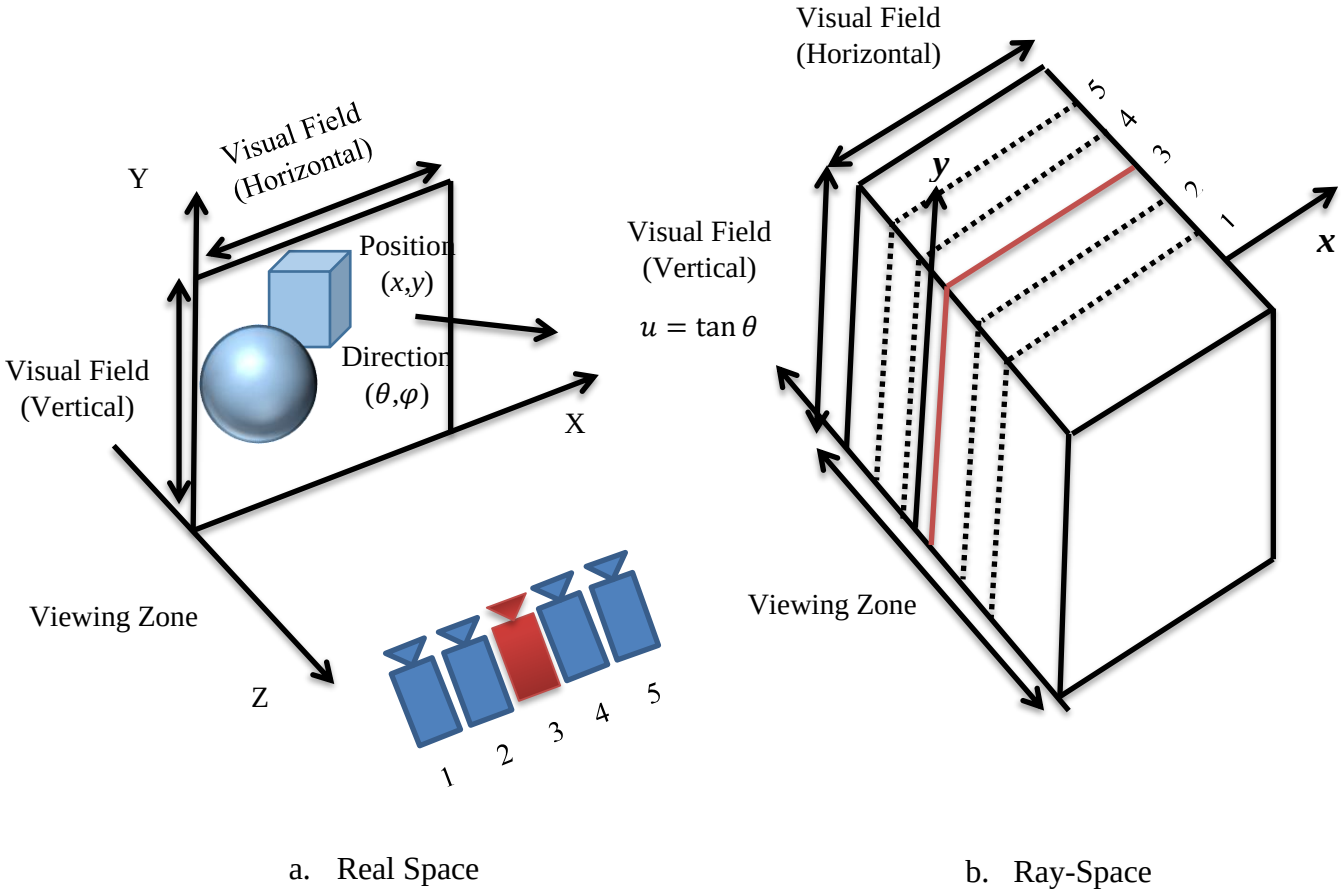
Ray-Space representation is one of the ways used to represent 3D scenes by converting the original multiview images to "ray" parameters [Fuji_94, Tani_12a]. In addition, ray-space is a virtual space, but it is straightforwardly joined to real space.

In a three-dimensional scene, a ray is represented by using a set of parameters, which can describe all rays of 3D space. The ray parameters indicate the direction of the light ray and the coordinates of the intersection of the ray and x-y plane [Adel_91, Mcmi_95, Levo_96]. In Fig. 2.8, a ray in a 3D scene is represented by four parameters; two parameters for the direction (θ, φ) of the light ray and the other two for the position (x, y) that represents the intersection of the ray and the reference plane.

In ray-space representation, the image information from a specific viewpoint is represented as one subspace of all the ray-space. The data captured by the organization presented in Fig. 2.8a can be shown as a ray-space shown in Fig. 2.8b. The five views in Fig. 2.8a correspond to

the 1st to 5th vertical sections in Fig. 2.8b. Thus, an arbitrary viewpoint image can be created by reading and transforming the corresponding data in the ray-space. Therefore, ray-space representation can be used in applications like Free Viewpoint Television [Tani_12b] and 3-dimensional television [Shao_05, Tani_12b].

Since ray-space is composed of a large number of 2D real images, it has a huge amount of data, and thus, must be compressed before storage or transmission. Since most of the ray-space compression methods only use two-dimensional intra-frame redundancies, they are similar to the still image compression techniques, leading to low compression efficiency. Therefore, compression is one of the important issues that must be overcome in ray-space representation.



a. Real Space b. Ray-Space

Fig. 2.8. Definition of Ray-Space [Fuji_94, Tani_12a].

2.7 Compression of Multiview Video Plus Depth Maps

As already mentioned in Section 2.6.1, MVD includes multiple videos and associated depth maps, which means that the volume of MVD data is huge. Therefore, MVD must be compressed before transmission and storage. Many methods have been proposed to compress multiview video plus depth maps: simulcast coding, multiview video coding, and 3D video coding.

2.7.1 Simulcast Coding of Multiview and Multiview Plus Depth Video

A straightforward method to compress multiview video plus depth is to use 2D coding, usually known as simulcast coding. Simulcast coding consists of independently encoding the views and depth maps, meaning that each view is coded separately, and each depth map is coded separately. For example, standard video encoders such as AVC [Rich_03b], HEVC [Sull_13, Sze_14], VVC [Sull_18, Bros_20, Hami_22], or AV-1 [Riza_18, Trow_20] can be utilized for this purpose. Fig. 2.9 shows the compression of stereoscopic video plus depth by using simulcast coding. An example of the prediction structure of a simulcast coding algorithm is explained in Fig. 2.10, where each view is independently coded, and compression only exploits temporal redundancy [Merk_07]. The arrows in Fig. 2.10 indicate the direction of prediction in GOP (all arrows from a reference picture to the prediction target picture).

An advantage of simulcast coding is that standard video encoders can be used for multiview plus depth coding. This approach is the simplest solution, compared to later solutions, as it only exploits temporal redundancy and does not exploit spatial redundancy between views, keeping the computational cost and the coding delay on levels achieved by state-of-the-art schemes in video coding. Also, it is a suitable solution for the current technology, as each of the views can be decoded using existing decoders.

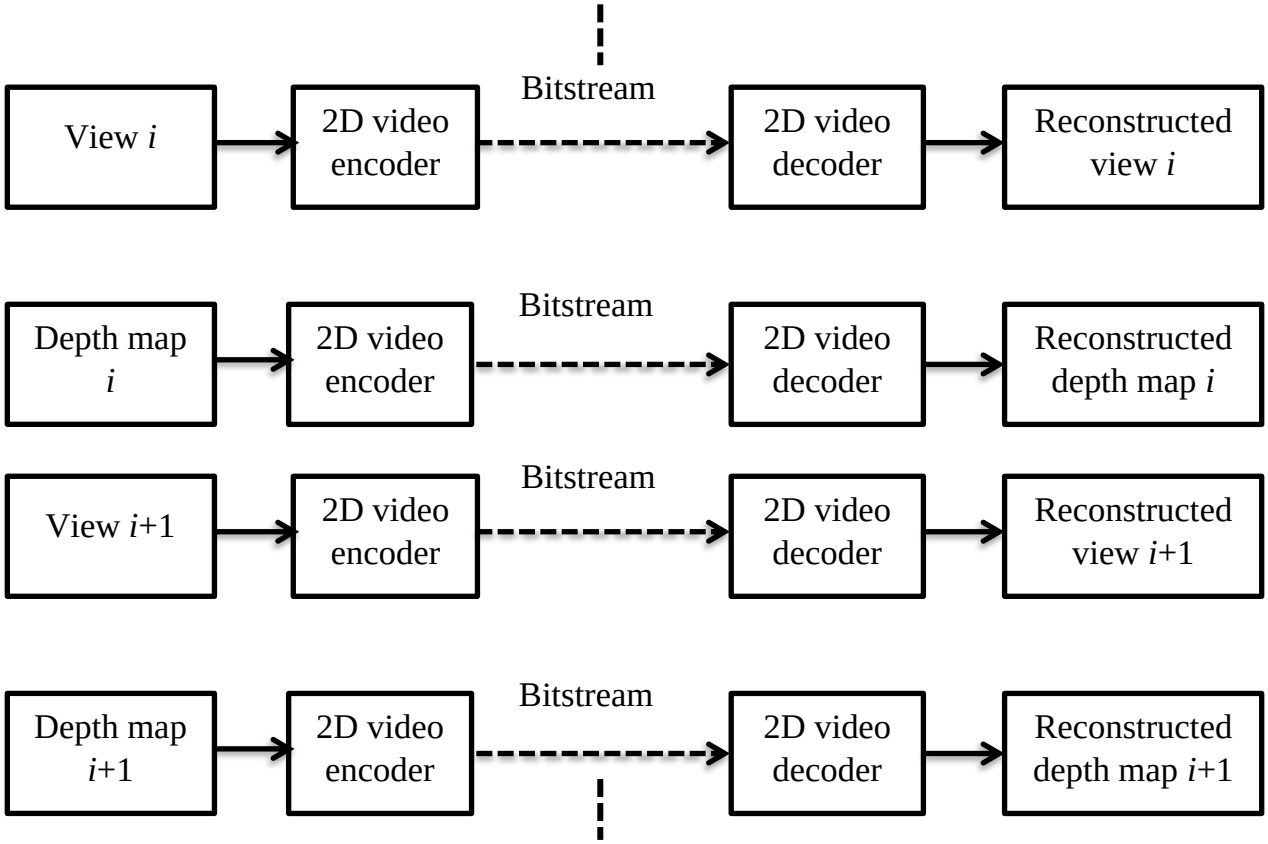


Fig. 2.9. Simulcast coding of stereoscopic video plus depth.

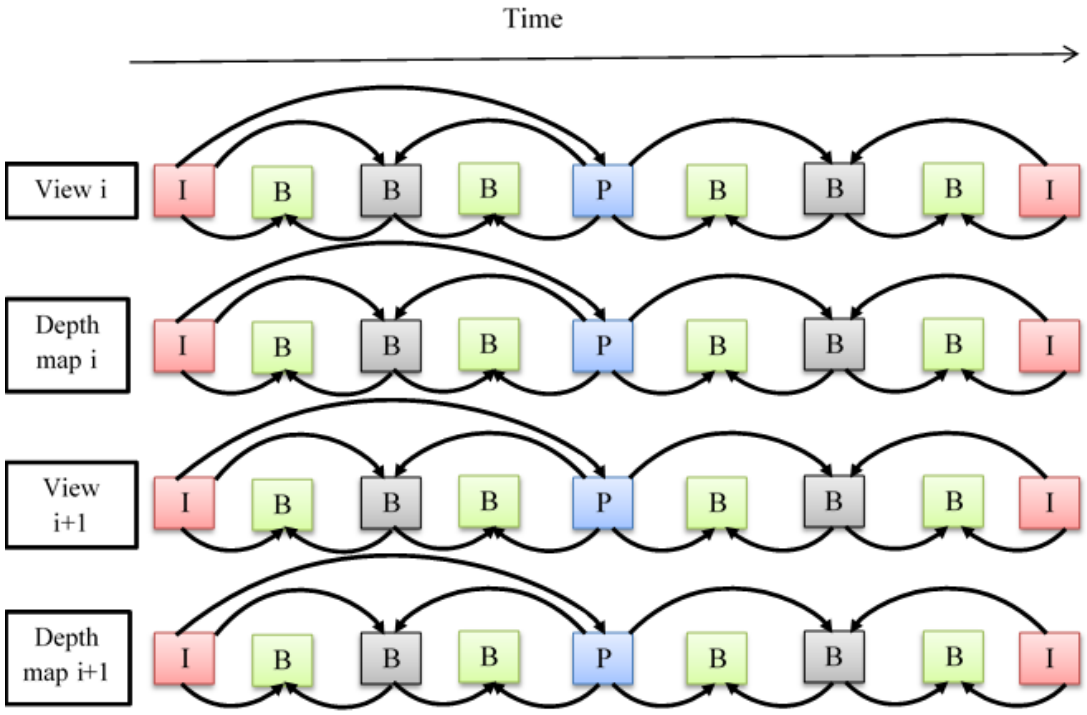


Fig. 2.10. An example of prediction structure for stereoscopic video plus depth in simulcast coding.

2.7.2 Multiview Video Coding

Multiview video coding involves coding two or more views together and then sending them in a single bitstream. Fig. 2.11 presents a block diagram of the compression of stereoscopic video plus depth by using multiview coding. Multiview video coding techniques (e.g., MV-HEVC, MVC) [Vetro_11, Tech_16, Chou_22, Huv_23, Deng_23] provide a more effective method to code multiview video plus depth than simulcast coding by taking advantage of inter-view redundancy. The approach exploiting the temporal redundancies can be used to eliminate inter-view redundancy by disparity-based techniques. The disparity between various views is treated as a movement in the temporal direction, and the same methods utilized to model motion fields are used to model disparity fields [Hann_13, Chen_15]. Fig. 2.12 explains an example of a multiview coding structure with both temporal and inter-view prediction for stereoscopic video plus depth. The arrows in Fig. 2.12 show the direction of prediction from a reference picture to the prediction target picture.

One of the essential features of multiview video coding is that the primary block-based coding and the decoding process of 2D coding stay fixed. In addition, the primary principle of multiview video coding is to reuse the two-dimensional coding tools, with modifications made only to high-level syntax on the slice header level and above. Multiview video coding uses one of the views as a base view, while the rest of the views as dependent views, as shown in Fig. 2.12. The base view is coded by using standard 2D video coding. In contrast, the dependent views are created by inter-view prediction to achieve high compression performance.

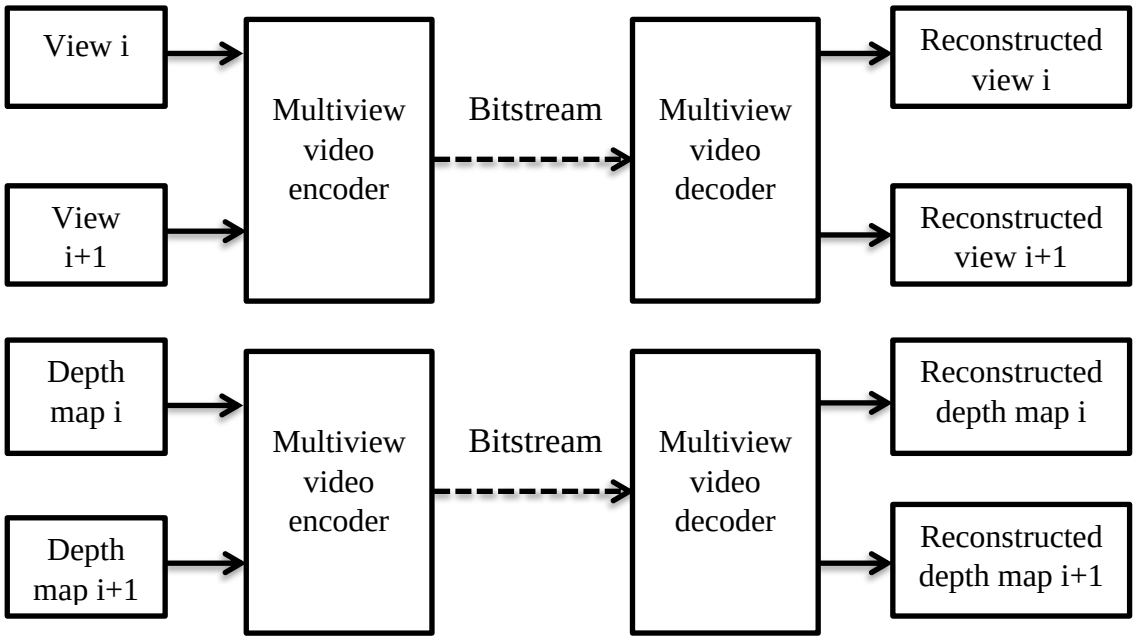


Fig. 2.11. Compression of stereoscopic video plus depth by using multiview coding.

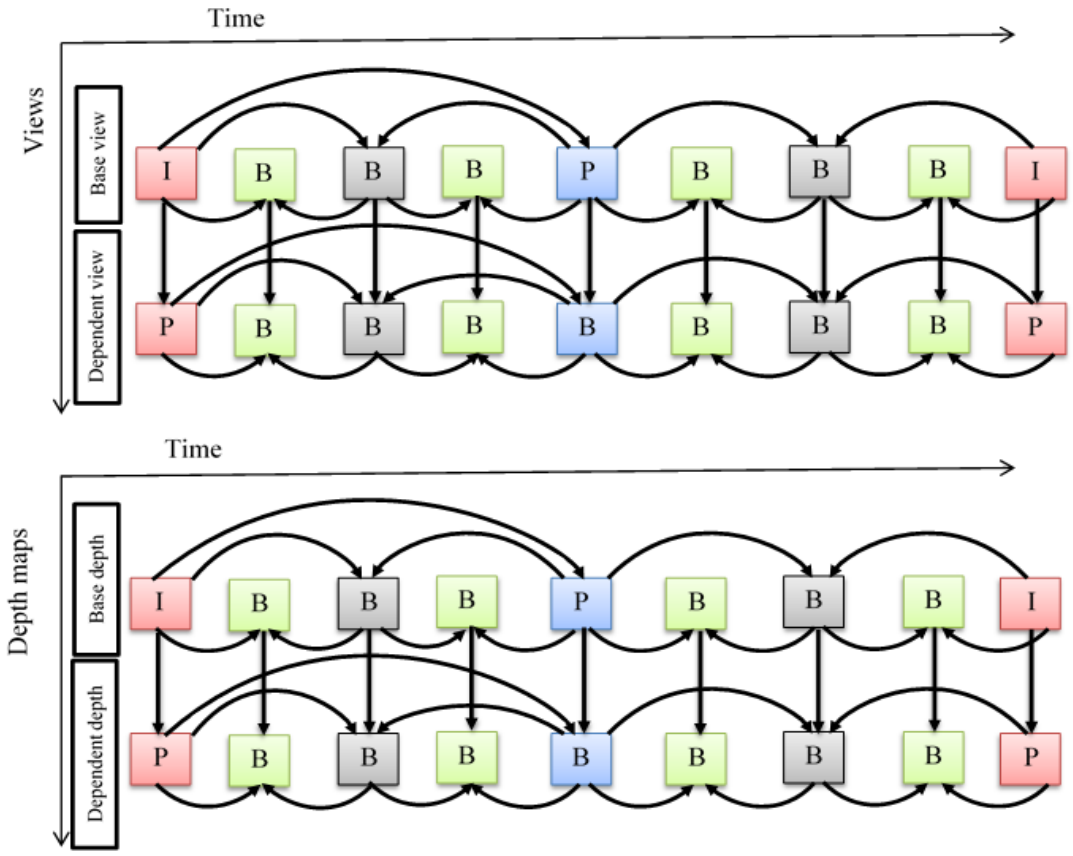


Fig. 2.12. Example of prediction structure with both temporal and inter-view predictions for stereoscopic video plus depth in multiview video coding.

2.7.3 3D Video Coding

3D video coding consists in joint compression of all videos and depth maps, which means exploiting the correlations between views and between video and depth components [Doma_13, Tech_16]. 3D video coding does not exploit only temporal and inter-view redundancies, but also the inter-component redundancy for video and depth.

3D video coding (e.g., 3D-AVC, 3D-HEVC) is an extension of the standards of video coding (e.g., AVC or HEVC) that are being used by adding extra coding tools and inter prediction techniques between components, as indicated by the red arrows in Fig. 2.14. One of the views, which is called the base view or independent view, is coded independently of the other views using 2D HEVC video coding. The other views are called dependent views because they may be coded depending on the data of the other views. 3D video coding uses quantization parameters for views and depth maps to find a trade-off between the quality of views and bitstream size. Also, camera parameters are additionally included in the bitstream for view synthesis [Doma_13, Chen_15]. 3D video coding is used in many practical applications, such as advanced three-dimensional television (3DTV), free-viewpoint television (FTV), and 3D digital cinema applications. Fig. 2.13 presents a block diagram of the compression of stereoscopic video plus depth by using 3D video coding. In addition, Fig. 2.14 shows a 3D video coding structure.

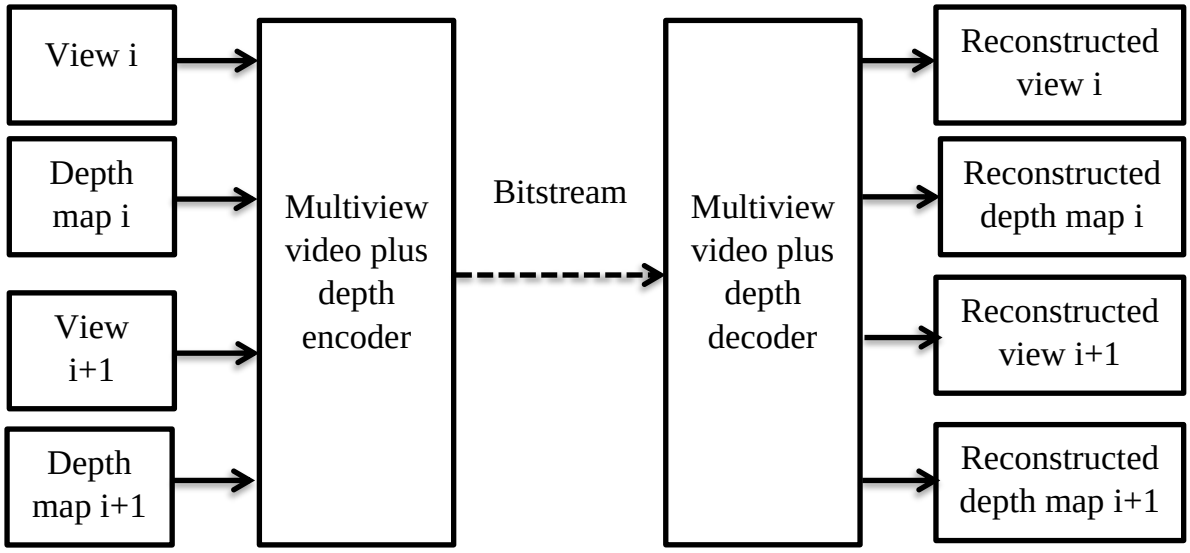


Fig. 2.13. Compression of stereoscopic video plus depth by using 3D video coding.

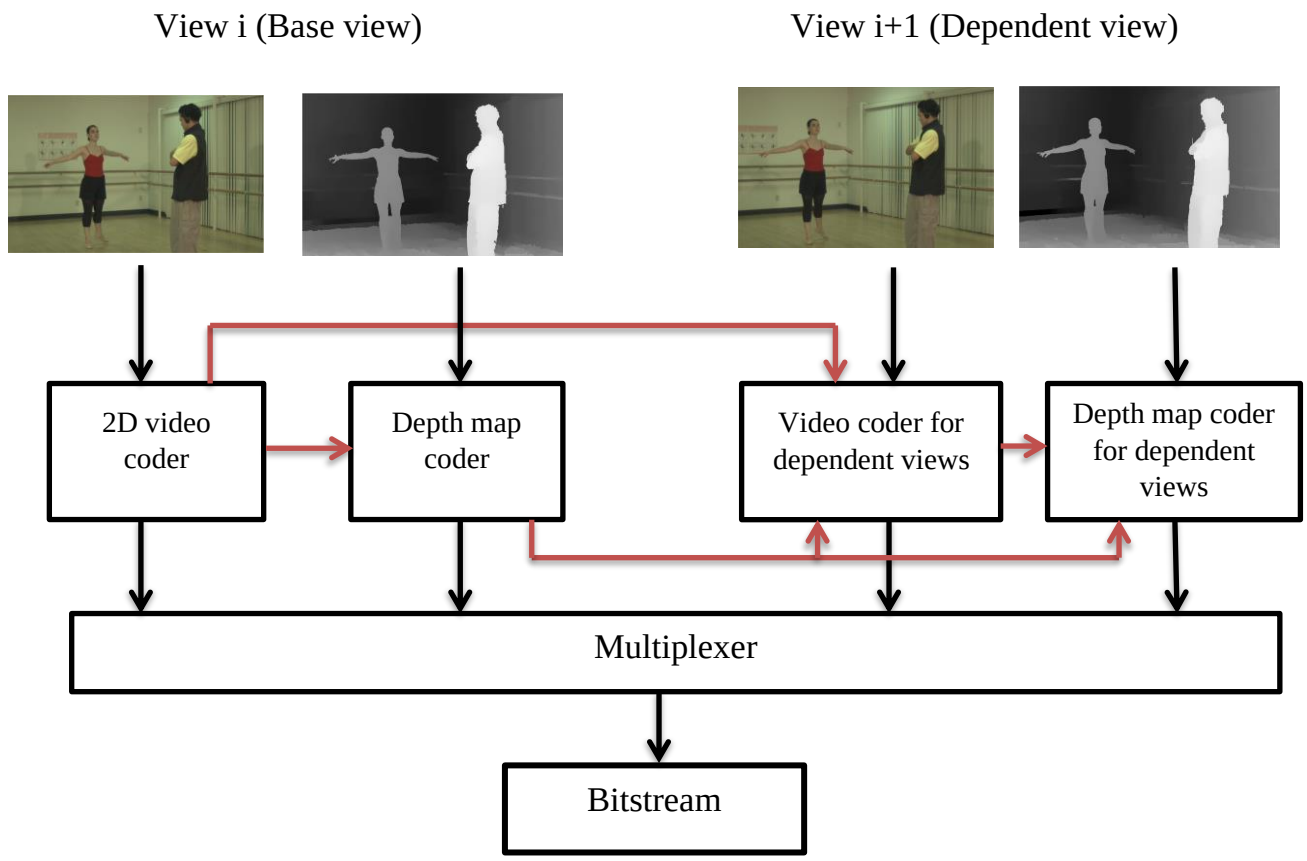


Fig. 2.14. 3D video coding structure with the inter-component prediction (red arrows) for stereoscopic video plus depth.

2.8 View Synthesis

In three-dimensional video applications (e.g., FTV, 3DTV, etc.), it is necessary to use the view synthesis process to create virtual views. Virtual views are generated by methods of view synthesis from videos and depth maps obtained by a camera system. Thus, the virtual view quality depends on the quality of views and depth map information in the view synthesis process [Dzie_16, Puri_16].

Many virtual view synthesis methods were proposed in research [Tani_09, Dzie_16, Li_22, Zhan_22]. Therefore, virtual view synthesis methods can be classified according to several aspects. The first one is the number of real views that are used to synthesize the virtual view. Most methods allow using two real views to create a virtual view [Tani_09, Li_13, Jin_16a], while other methods of synthesizing a virtual view depend on three or more real views [Dzie_16, Ceul_18]. Secondly, the existing methods can be divided according to the camera arrangement in multiview systems, such as systems with linear camera arrangement [Akin_15, Jun_15] or nonlinear camera arrangement [Doma_14, Yu_16]. Thirdly, the synthesis methods can be categorized according to the type of camera used in a multiview system: light-field cameras [Levo_06, Kim_13, Yao_16], or omnidirectional cameras [Wegn_17, Doma_18]. Fourthly, the methods of synthesis view can also be classified according to the computation time. Methods that allow the synthesis of virtual views in real-

time can also be classified as designed for FPGA (Field Programmable Gate Array) [Wang_12], graphics cards [Rogm_09, Do_11, Yao_16], or CPU (Central Processing Unit) [Dzie_18b].

2.9 Encoder Modeling and Bit Allocation for Multiview Video Plus Depth

The issue of bit allocation for multiview video plus depth has already been studied in many previous works [Fehn_04, Mull_09, Stan_13a]. As is well known, the aim of bit allocation is to obtain the highest possible quality of synthesized views based on the decoded views and depth map data at the assumed bitrate. Therefore, many methods have been proposed to find the bit allocation for multiview video plus depth maps; let us present a review of the previous studies.

A study on the influence of bitrate allocation for video and depth data based on the synthesized view quality was presented in [Bosc_11, Bosc_13], while no formula was proposed that satisfies this requirement. In similar research [Mull_09], the authors did not propose any formula to solve the bit allocation issue, like in [Bosc_11, Bosc_13].

In [Morv_07], the effect of sequence coding on the virtual view quality has been studied, but the results obtained in this research were based on very limited assumptions, where only intra-image coding was used. Also, the study was based on the analysis of the results of using only two sequences of a very similar nature.

Another study discussed the bit allocation issue [Fehn_04], using an approach depending on a fixed ratio (5:1) to allocate bits between views and depth maps, meaning that the approach did not consider the influence of the content factor in the views and the depth maps on the quality of the virtual view. Therefore, this approach is rarely used to calculate bitrate allocation for multiview video plus depth.

A model was proposed by [Klim_14a] to calculate bit allocation by finding a nonlinear relationship between the quantization parameter for views (*QP*) and the quantization parameter for depth maps (*QD*). Furthermore, the authors did not calculate ΔPSNR and ΔBitrate between coding with their proposed approach against coding with other approaches. Unlike the previous study [Klim_14a], the authors in [Stan_13a, Klim_14b] used linear models to describe the relationship between the quantization parameters of the video data and depth data. In this study, the coding performance with algorithmically optimized *QP*-*QD* quantization parameter pairs was compared with a reference (*QP* = *QD*).

Some researchers did not provide a formula to represent the relationship between *QP* and *QD* in order to calculate bit allocation for multiview plus depth, while other studies introduced some models to describe this relationship. All models proposed in the previous studies are only suitable for finding bitrate allocation with linear multiview sequences. Therefore, the issue of bitrate allocation for MVD video remains open.

2.10 Influence of the Depth Map on Virtual View Quality

One of the most important issues in multiview video plus depth is to know the influence of depth map fidelity on the quality of the synthesized virtual view [Mull_11, Puri_16]. Therefore, many studies have examined this issue, as will be further presented.

In [Nur_10], the authors studied the impact of the spatial resolutions of depth maps encoded in various qualities on 3D video quality and depth perception. Additionally, the authors concluded that the higher the spatial resolution of depth maps was, the more did the video quality and depth perception increase.

The authors in [Yama_10] proposed using the luma PSNR average of the synthesized view produced from uncompressed views and compressed depth maps relative to the synthesized view from uncompressed views and depth maps as a measurement of the effect of depth map quality.

Another research studied the relationship between depth map quality and synthesized video quality measured by perceptual quality metrics [Bani_13]. Two possible depth map artifacts, which are only generated in depth map compression with quantization parameters 25 and 45, were applied to the depth map sequence corresponding to the left view of a stereo video pair. The synthesized views were created depending on the original views and the distorted depth map sequence. Subjective evaluations showed that the synthesized video quality depends on the depth map quality.

The influence of a virtual depth map synthesized from neighboring views on virtual view quality was studied in [Lee_11]. The authors found that the boundary regions in the synthesized depth image only led to producing visual artifacts in the virtual image. Furthermore, the authors suggested an edge-based depth map coding method, which only encodes the boundary areas, including the edge blocks, and skips the remaining areas without encoding.

In previous studies, the effect of depth map quality on virtual view quality has been studied, but the outcomes obtained in these studies were based on certain assumptions, meaning that these studies were not based on comprehensive research on this issue. Therefore, such a comprehensive study will be presented in the dissertation.

2.11 Conclusions

This chapter presents the scientific and technical challenges of multiview video. One of these challenges is how to represent a 3D scene by using multiview videos. Many ways were presented to represent a 3D scene like multiview video plus depth, point cloud, and ray-space. Multiview video plus depth is the most popular and widely used representation in research and applications – such as virtual navigation, free-viewpoint television, and virtual reality – related to 3D video. Multiview video plus depth will be considered in this dissertation because this format will allow any intermediate view within a particular range to be produced using view synthesis. Consequently, it can reduce the number of transmitted views. Additionally, the video

plus depth representation is suitable for rendering and compression [Merk_07]. In contrast, the point cloud and ray-space formats will not be considered in the dissertation because they lack effective and reliable compression methods.

Due to the required data of multiview video plus depth, the video is still very large, which is why many ways have been presented to compress multiview video plus depth maps, which are simulcast coding, multiview video coding, and 3D video coding. The 3D video coding is the most effective compared to other codecs, especially with MVD sequences which have been acquired using cameras with parallel optical axes and densely distributed in a line (linear camera arrangement). Simultaneously, the efficiency of 3D video coding is significantly lower with MVD sequences which have been obtained by using cameras sparsely located around a scene, and their optical axes convergent with the parallax of 10-20 degrees of arc even (nonlinear camera arrangement) [Doma_16d, Same_16]. Therefore, all MVD compression methods will be considered in the dissertation.

In multiview video plus depth, the bitrate allocation between views and depth maps affects the compression efficiency. Although many methods of bit allocation for multiview video plus depth were presented in Section 2.9, these presented methods correspond only to linear multiview sequences, meaning that they are not compatible with nonlinear multiview sequences. Therefore, a new bit allocation method for multiview video plus depth was presented to meet this requirement in the dissertation.

As is well known, rate control plays an important part in 2D and 3D video coding to satisfy different communication channel throughput and limited storage memory space. All video codecs need rate control to deal with variable bitrate properties of the coded bitstream and to produce good video quality at a given bitrate. Rate control algorithms usually depend on encoder modeling and bit allocation. The bit allocation can be performed on three levels: GOP-, Frame-, and block-level. In contrast, encoder modeling can be performed with three approaches: the R-Q approach, the R-ρ approach, and the R-λ approach. The R-Q model will be considered in the dissertation due to this model's accuracy in determining the bitrate for any video coding.

Chapter Three

Methodology of Experiments

3.1 Goals of Experiments

Rate control is not a part of the video coding standard; yet, it is an essential part of the encoding process. The encoder has to control many parameters to reach a target bitrate, and also at the same time, provide good quality of decoded video.

The main goals of the experiments are as follows:

1. Assessment of models for bitrate or number of bits as functions of the quantization step or quantization parameter.
2. Estimation of rate-distortion (R-D) characteristics for various parameters of encoders. That way, the optimum or near-optimum parameters of the encoders will be estimated.
3. Assessment of the rate control techniques based on the models derived.

3.2 Test Sequences Used in Experiments

The dissertation mainly deals with multiview plus depth video (MVD). For the sake of limited complexity, the experiments are limited to a "two views plus two depth maps" video. The experiments are executed using video test sequences recommended by the Moving Picture Experts Group (MPEG) affiliated with the International Organization for Standardization (ISO). In all experiments, two sets of test sequences are used, which are training and verification sets. The training set is used to estimate the parameters of the proposed models, while the verification set is used to assess the proposed models. The test sequences differ in their content and resolution.

The 2D sequences used in the experiments are summarized in Table 3.1. Table 3.2 shows the multiview plus depth (MVD) sequences used in the experiments. For MVD video, the quality is measured for a synthetic view. Therefore, the column "Synthesized view" in Table 3.2 provides the view number that defines the geometrical location of the synthetic view. The reference views for synthesis are fixed for each sequence (cf. "Used views" column) in Table 3.2. Fig. 3.1 presents examples of frames for one view of each MVD sequence.

Table 3.1: Test 2D sequences used in experiments

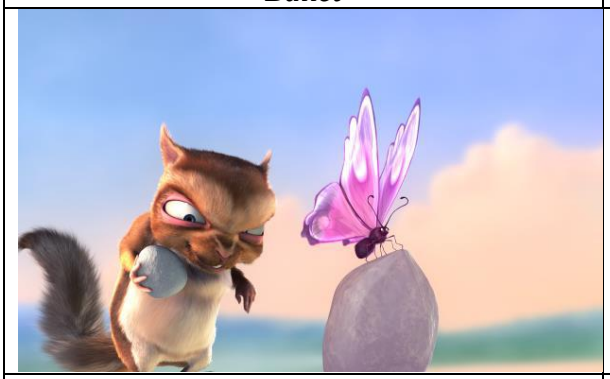
Sequence name	Resolution	Frame rate
Training set of 2D sequences		
PeopleOnStreet [JCT]	2560×1600	30
Traffic [JCT]	2560×1600	30
Kermit [Sala_19]	1920×1080	25
Poznan_Block2 (view 2) [Doma_16b]	1920×1080	25
Poznan_Fencing (view 2) [Doma_16b]	1920×1080	25
BBB.Butterfly (view 49) [Kova_15]	1280×768	25
BBB.Flowers (view 39) [Kova_15]	1280×768	25
Ballet (view 3) [Zitn_04]	1024×768	25
Breakdancers (view 2) [Zitn_04]	1024×768	25
Keiba [JCT]	832×480	30
RaceHorses [JCT]	832×480	30
Basketball_Drill [JCT]	832×480	50
Basketball_Pass [JCT]	416×240	50
BQSquare [JCT]	416×240	60
Verification set of 2D sequences		
Poznan_CarPark (view 3) [Doma_09]	1920×1088	25
FourPeople [JCT]	1280×720	60
ChinaSpeed [JCT]	1024×768	30
BQMall [JCT]	832×480	60
BlowingBubbles [JCT]	416×240	50

Table 3.2: Test MVD sequences used in experiments

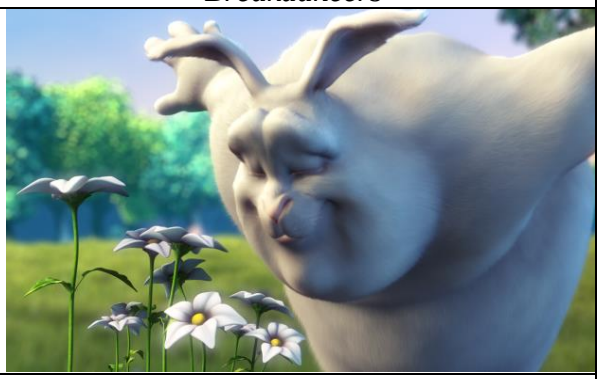
Sequence name	Resolution	Used views	Synthesized view
Training set of MVD sequences			
Ballet [Zitn_04]	1024×768	3, 5	4
Breakdancers [Zitn_04]	1024×768	2, 4	3
BBB.Butterfly [Kova_15]	1280×768	49, 51	50
BBB.Flowers [Kova_15]	1280×768	39, 41	40
Kermit [Sala_19]	1920×1080	5,7	6
Poznan_CarPark [Doma_09]	1920×1088	3,5	4
Verification set of MVD sequences			
Poznan_Block2 [Doma_16b]	1920×1080	2, 6	4
Poznan_Fencing [Doma_16b]	1920×1080	2, 6	4

Ballet

Breakdancers



BBB.Butterfly

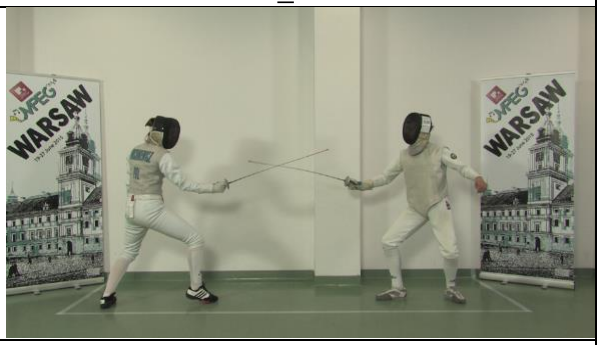


BBB.Flowers

Kermit

Poznan_CarPark

Poznan_Block2



Poznan_Fencing

Fig. 3.1. Examples of views from test sequences used in experiments.

3.3 Test Video Codecs

The experiments are executed using the basic standard video codecs (mentioned in Section 2.7), as shown in Table 3.3.

Table 3.3: Video codecs used in experiments

Video coding standard		Reference software	Common test conditions
AVC (Advanced Video Coding) (ISO/IEC 14496-10, ITU-T Rec. H.264) [AVC_std]		JM codec in version 19.00 [AVC]	[Tour_09]
HEVC (High Efficiency Video Coding) (ISO/IEC 23008-2, ITU-T Rec. H.265) [HEVC_std]	Single view	HM codec in version 16.18 [HEVC]	[Boss_12]
	Multiview	HTM codec in version 16.3 [MVHEVC]	[MVHEVC]
	3-D	HTM codec in version 16.3 [3DHEVC]	[3DHEVC]
VVC (Versatile Video Coding) (ISO/IEC 23090-3, ITU-T Rec. H.266) [VVC_std]		VTM codec in version 7.3 [VVC]	[Boss_19]

The encoders are configured according to the MPEG common test conditions for 2D and 3D sequences in the experiments, as shown in Table 3.3. The data are obtained for *QP* (the quantization parameter) values from 15 to 50, which corresponds to the used bitrates in practice. The GOP structure used in the experiments for all codecs is I B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3 P B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3, i.e. a GOP that comprises 32 frames.

3.4 View Synthesis

The state-of-the-art synthesis software called View Synthesis Reference Software (VSRS 3.5) is often used in the literature (e.g., [Rana_10, Luca_13, Oh_14, Liu_15, Miel_18a, Raha_19, Miel_20, Li_22, Zhan_22]). This software was developed and is continuously improved by the ISO/IEC MPEG community [Miel_18b]. Therefore, View Synthesis Reference Software (VSRS 3.5) [Stan_13b] will be used in all experiments in this dissertation to synthesize the virtual views.

3.5 Video Quality Assessment

Fig.3.2 shows the virtual view quality assessment flow chart. Initially, views and depth maps were encoded and then decoded using proposed compression techniques (shown in Section 3.3). Then, decoded views and depth maps were used to create a required virtual view. This virtual view was compared via video quality assessment with the view acquired by a real camera in exactly the same position in the 3D space as the created virtual view.

Video quality assessment is an essential matter in a free-viewpoint television system, where the quality of virtual views determines the total quality of the system. Many quality measures have been used to assess the quality of the virtual view [Lin_14, Fari_15]. Therefore, quality evaluation methods can be divided into subjective and objective ones. Subjective assessment of content is considered the most reliable assessment method, but it is the most expensive, difficult, and time-consuming method of content assessment [Wink_05, Koni_12]. On the other hand, objective assessment (like PSNR) is considered a more straightforward and uncomplicated assessment method than a subjective one. The objective assessment usually provides a well-made content quality evaluation. Also, objective and subjective metrics in most cases produce similar results [Papa_11, Pino_14, Jung_17]. Therefore, objective assessment (PSNR) is used in many studies (e.g., [Merk_07, Noor_13, Oh_14, Doma_15, Jin_16b]) to measure the quality of the virtual view. Due to the above reasons, the objective measure (PSNR, IV-PSNR) has been chosen for virtual view quality evaluation in the dissertation.

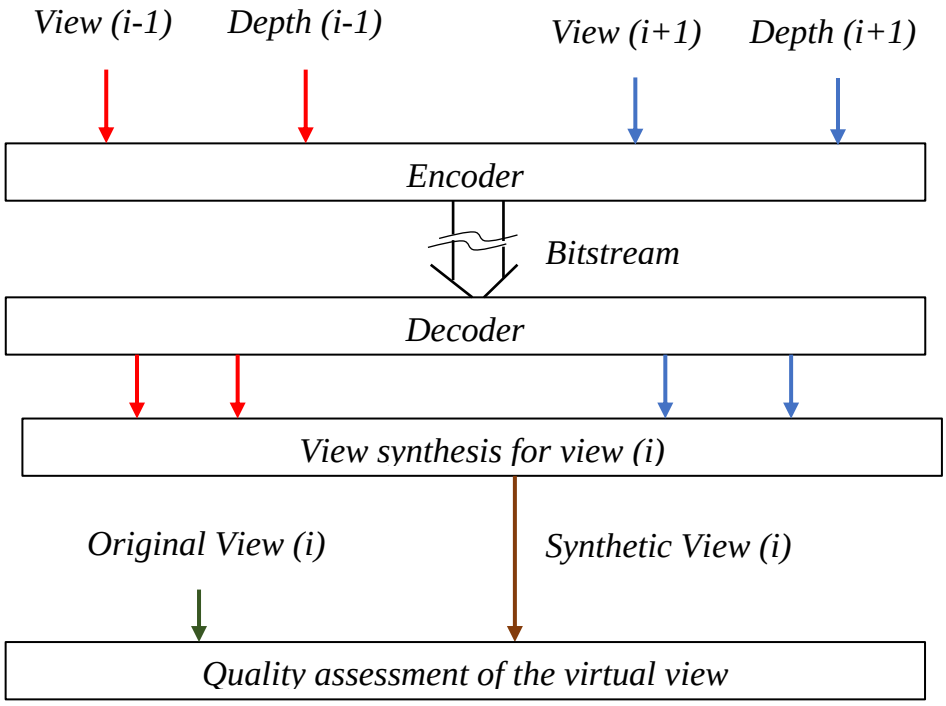


Fig. 3.2. Virtual view quality assessment flow chart.

3.5.1 PSNR

The term *peak signal-to-noise ratio* (PSNR) [Wink_05, Akra_14, Joshi_15, Likh_22, Wali_22] is an expression for the ratio of the maximum signal power to the power of distortion that affects the quality of its representation. PSNR is expressed in decibels according to formula (3.1). PSNR is defined as

$$PSNR[dB] = 10 \cdot log_{10} \frac{M \cdot N \cdot (2^n - 1)^2}{\sum_{I=1}^{N} \sum_{J=1}^{M} (f(i,j) - g(i,j))^2} \tag{3.1}$$

Legend:

$f(i,j)$ - sample value for the original image pixel with coordinates *i, j*,

$g(i,j)$ - sample value for the synthesized image pixel with coordinates *i, j*,

M - number of rows of pixels of the images; *i* represents the index of that row,

N - number of columns of pixels of the image; *j* represents the index of that column,

n - number of bits per pixel - in all cases described in this work, *n* equals 8.

The value of PSNR is usually measured only for the luminance component of the picture because the distortions in chrominance components of the image are less visible [Sull_98]. Also, determining the virtual view quality only for luminance is a common practice, often used in literature (e.g., [Jun_15, Dzie_16, Ceul_18, Fach_18, Li_18a]). Therefore, PSNR will be calculated only for the luminance component (Y component) of the picture.

In a virtual view, the PSNR measure doesn't always reflect the quality measured using a subjective method. For example, a very small object edge shift that often occurs in the synthesized view does not change the viewer's impression but decreases the quality of the virtual view measured by PSNR [Puri_15]. Although PSNR measurement is not accurate in all cases, PSNR is characterized by simplicity. Thus, in most publications in the field of virtual view synthesis, the quality of synthesized views is determined by PSNR. Therefore, Y-PSNR (PSNR for luminance component) will be used in this dissertation to calculate the virtual view quality.

Recently, a new method has been proposed, which is IV-PSNR, to measure the quality of synthesized views to solve some cases of the inaccuracy of PSNR mentioned above.

3.5.2 IV-PSNR

IV-PSNR [Dzie_19, Dzie_20, Dzie_22] is an objective quality metric proposed for stereoscopic, multiview, and immersive video applications and it is PSNR with some modifications. The corresponding pixel shift and global color shift are the substantial modifications that were added to regular PSNR. The corresponding pixel shift aims to take out the effect of a little shift of object edges produced by the re-projection error. Whereas, the

global color difference intends to lessen the impact of various color characteristics of diverse input views.

IV-PSNR value is calculated as:

$$IV - PSNR = \frac{\sum_{C=0}^{L-1} IV-PSNR(C)\cdot CCW(C)}{\sum_{C=0}^{L-1} CCW(C)} \tag{3.2}$$

Legend:

IV-PSNR(C) - IV-PSNR for component *C*,

CCW(C) - color component weight for each color component *C* (e.g., $CCW = 1$ for the luminance component, and $CCW = 0.25$ the chrominance components [Dzie_19, Dzie_20]),

L - number of components (e.g., *L* for YUV is 3).

IV-PSNR(C) is calculated by using equation (3.3).

$$IV - PSNR(C) = 10 \cdot \log\left(\frac{W\cdot H\cdot MAX^2}{\sum_{y=0}^{H-1}\sum_{x=0}^{W-1} \underset{\substack{x_R\in\left|x-\frac{B_k}{2},x+\frac{B_k}{2}\right| \\ y_R\in\left|y-\frac{B_k}{2},y+\frac{B_k}{2}\right|}}{min} \left(f(x,y,C)-g(x_R,y_R,C)+GCD(C)\right)^2} \right) \tag{3.3}$$

Legend:

W - width of the image,

H - height of the image,

MAX - maximum value of luma (a color component),

f - original image,

g - synthesized image,

B_k - size of the analyzed block in the original view,

GCD(C) - global color difference for component *C*.

According to [Dzie_19, Dzie_20], $B_k = 5$ is chosen in the experiments, which corresponds to a 2-pixel shift of object edges.

The global color difference is determined as follows:

$$\mathrm{GCD(C)} = max\left(\frac{1}{W\cdot H}\sum_{y=0}^{H-1}\sum_{x=0}^{W-1}\left(f(x,y,C)-g(x,y,C)\right), MUD(C)\right), \tag{3.4}$$

where *MUD(C)* is the maximum unnoticeable difference for color component *C*. In the experiments, *MUD* =1% was assumed for all the color components, according to [Dzie_19, Dzie_20].

3.6 Bjøntegaard Metrics

Bjøntegaard metrics are widely utilized in research to compare the coding efficiency of the codecs [Hann_13, Stan_13a, Yu_13, Klim_14b, Oh_14, Same_16, Stan_18, Topi_19, Mans_20, Meue_20, Siqu_20, Bros_21, Merk_22, Kim_23]. Therefore, Bjøntegaard metrics will be used in experiments in this dissertation to compare the coding performance with the proposed approach against the coding performance with other approaches.

Bjøntegaard metrics can be divided into two metrics. The first metric (represented by ΔBitrate [kbps]) calculates the average bitrate difference by comparing the bitrate of the streams produced by each of the analyzed codecs for the same quality of decoded images, as shown in Fig.3.3a. Whereas the second metric (represented by ΔPSNR [dB] or ΔIV-PSNR [dB]) computes average quality differences by comparing the quality of decoded images done for the same bitrate, as shown in Fig.3.3b. In Bjøntegaard metrics, the comparison results are usually calculated by four points generated with different *QP* values, but it can be easily extended to an arbitrary number of points when needed. A detailed explanation of the method for calculating Bjøntegaard metrics can be found in [Bjon_01].

The Bjøntegaard rate is computed between the two R–D curves, as shown in Fig.3.3. ΔPSNR and ΔBitrate are calculated in the following equations:

$$\Delta PSNR = \frac{\int_{Min_a}^{Max_a} [PSNR_1(Bitrate) - PSNR_2(Bitrate)]\, dBitrate}{Max_a - Min_a}, \tag{3.5}$$

$$\Delta Bitrate = \frac{\int_{Min_b}^{Max_b} [Bitrate_1(PSNR) - Bitrate_2(PSNR)]\, dPSNR}{Max_b - Min_b}, \tag{3.6}$$

where, Max_a, Min_a, Max_b, and Min_b are the integration bounds, as shown in Fig.3.3.

In Fig.3.3, the coding efficiency of Codec 1 is better than Codec 2 because it achieves a smaller bitrate at the same quality of the decoded image. Besides, Codec 1 accomplishes a better quality of decoded image for the same bitrate than Codec 2.

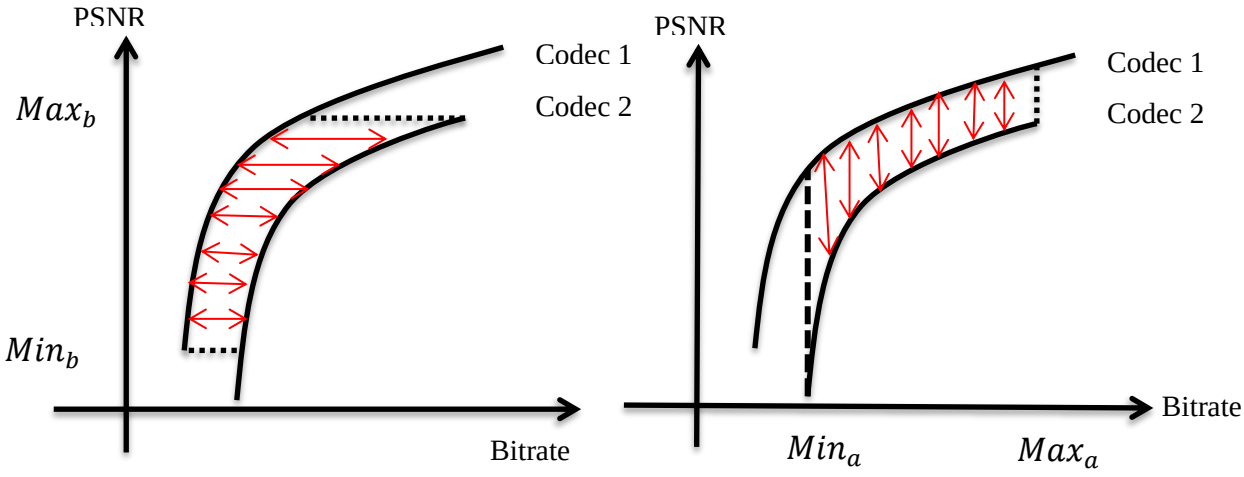


Fig. 3.3. Bjøntegaard metrics: (a) ΔBitrate, (b) ΔPSNR.

3.7 Trust-Region Method

The trust-region method is one of the optimization methods used in solving systems of nonlinear equations. The trust-region method can use non-convex approximate models due to the limitations of the trust region. This is one of the advantages of the trust-region method compared to the line search method. The trust-region method is reliable, efficient, and robust, it has powerful convergence characteristics, and it can be applied to unconditional problems. But, the trust-region method may tend to calculate slower than other methods (for example, the line search method) in each iteration. The trust-region method requires more computational effort than other methods in some cases (such as Hessian remains bounded) [Byrd_87, Alex_98, Conn_00, Mess_15, Bute_17].

To estimate the parameters of the encoding model (i.e., the parameters of the R-Q model), the trust-region optimization method can be used. Because the trust-region optimization method is often used in the literature of nonlinear systems (such as [Rodr_11, Mone_13, Wang_20b]), it will be used in the dissertation to calculate the encoder models' parameters. In experiments, this method is implemented by using the optimization toolbox of MATLAB [Math]. The number of iterations and tolerance value used in experiments is 2×10^{6} and 1×10^{-6}, respectively.

The trust-region method is based on decreasing the objective function (the real-valued function whose value must be reduced over the set of possible alternatives) within a predefined space. The first step is to determine the radius of the trust region around the current best solution. Inside this radius, the objective function is decreased to produce a vector, which is the minimization direction. The iterative modification of the trust-region radius is essential for the trust-region method. The step is rejected, the radius is decreased, and a new, more local solution is computed when the objective function cannot be reduced inside the trust-region area. Whereas if a step succeeds in reducing the objective function as approximated, then the trust-region radius is increased. In general, the direction alters whenever the trust-region radius is changed [Mess_15, Bute_17].

3.8 Conclusions

In this chapter, the virtual view quality assessment procedure for stereoscopic video plus depth was presented. Also, two-dimensional videos and MVD sequences used in the work have been shown. The modern versions of codecs of successful video coding standards have been selected to execute the main goals of the dissertation. Also, PSNR and IV-PSNR have been chosen to calculate the quality of the synthesized virtual view. The trust-region method has been chosen to estimate the parameters of the R-Q model used in the dissertation.

Chapter Four

VVC and HEVC Video Encoder Modeling Using AVC Data

4.1 Main Idea and Motivation

In this chapter, the first part of the original research results is presented. The chapter deals with two-dimensional (2D) video coding, as multiview video coding may be implemented by simulcast coding using any of the 2D video codecs.

This chapter relates to solving the problem of rate control for two-dimensional video. As mentioned in Section 2.1, several internationally standardized two-dimensional video compression technologies were developed like AVC [Rich_03b, Tour_09], HEVC [Sull_13], and VVC [Bros_20]. AVC coding is the least complex compression technology among those three [Topi_19].

In a video encoder, the output bitrate depends on the complexity of the content. Rate control is used to adjust the compressed video bitrate so that the channel throughput constraint can be met. Thus, rate control is an essential task to ensure the successful transmission of compressed video data through tight-band or time-varying channels in communication systems. Many methods have been proposed to implement rate control, as shown in Section 2.3, and the R-Q approach is one of the more efficient approaches to rate control. The quantization step size (Q) controls the bitrate and the number of bits assigned to individual frames that much depend on video content features. For example, Fig. 4.1 and Fig. 4.2 show R-Q curves for the two test sequences for different codecs.

In Fig. 4.1 and Fig. 4.2, it is observed that the shapes of R-Q characteristics for different codecs (AVC, HEVC, and VVC) are roughly similar for the given content. As it is well known, a common aim of rate control is to adjust encoder parameters to achieve a target bitrate. An AVC encoder is much faster than HEVC and VVC encoders, as mentioned in Section 2.1. Therefore, the complexity of the estimation of HEVC and VVC models can be significantly reduced by deriving the relationship between the HEVC and VVC model parameters and the AVC model parameters that can be estimated much faster.

In this chapter, we consider the practical usage of the similarity of the R-Q characteristics for AVC, HEVC, and VVC codecs. In particular, we consider the estimation of HEVC and VVC model parameters from the AVC model parameters. The goal is to use such parameters in rate control. The rate control is dealt with both for the GOP-level and for the frame-level in this chapter.

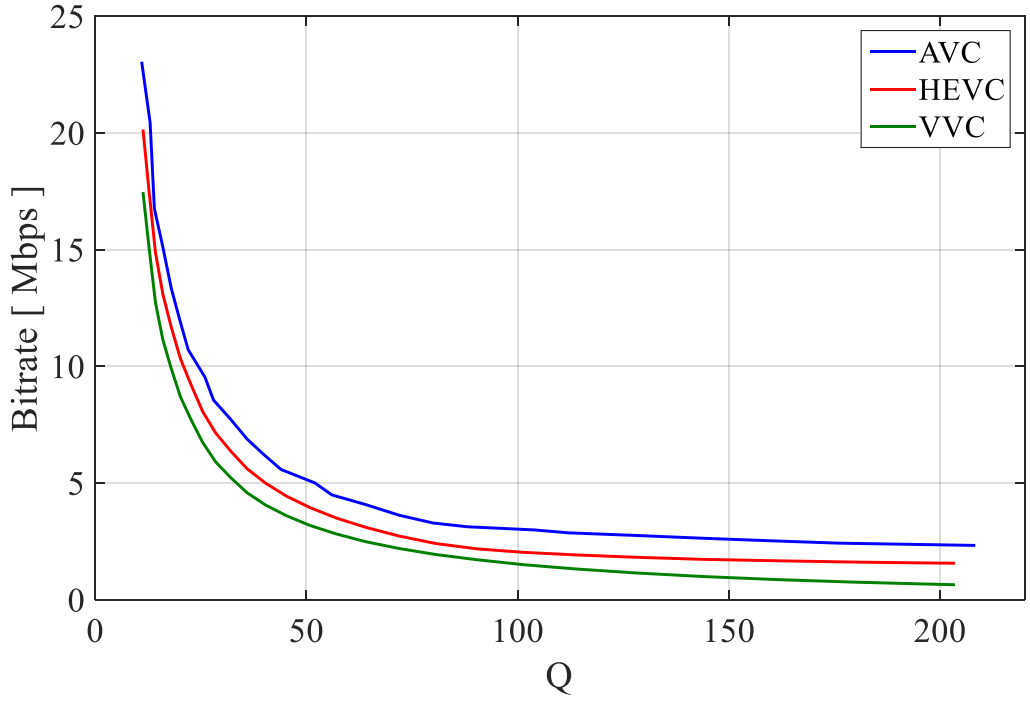


Fig. 4.1. Experimental curves for the *PeopleOnStreet* sequence for different codecs.

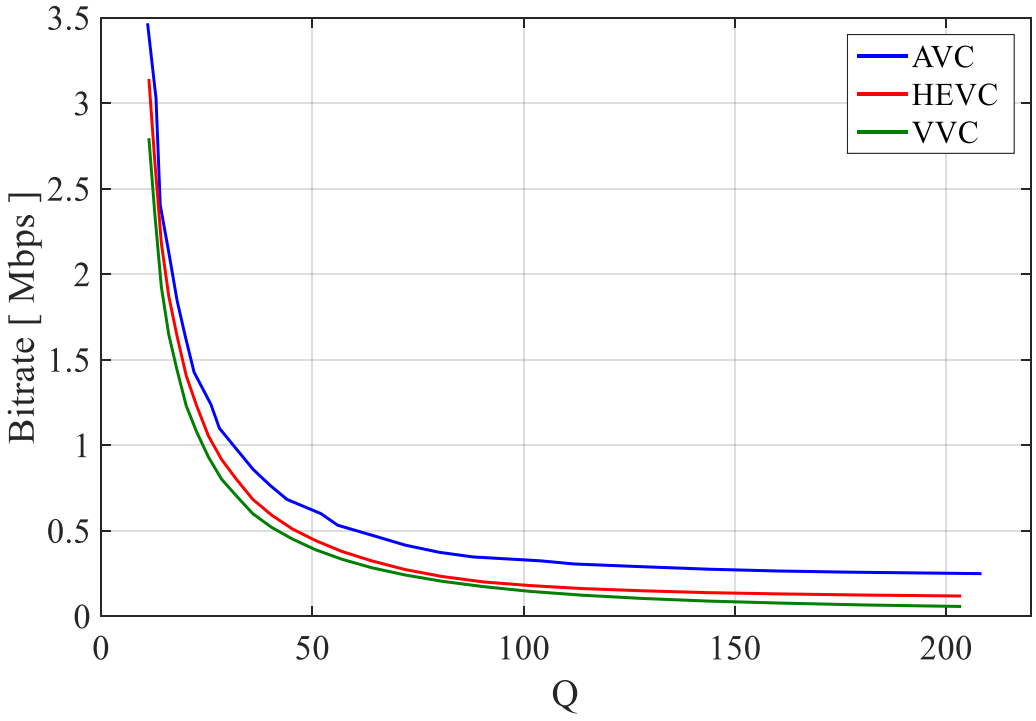


Fig. 4.2. Experimental curves for the *RaceHorses* sequence for different codecs.

4.2 R-Q Model Used

Several studies have described the relationship between the number of bits and quantization step size using R-Q models, as shown in Subsection 2.3.1. For AVC, the model used in [Graj_10] was experimentally compared to the standard model [Li_03]. For the MPEG test sequences defined for AVC, for typical broadcast bitrates, the approximation error for the model used in [Graj_10] was, on average, one-third of the error obtained with the technique used in the AVC reference software, as mentioned in [Graj_10]. Therefore, the model from [Graj_10] is used in this dissertation.

In this model, the relationship between the number of video bits and quantization step size for a given frame type in the 2D sequences is proposed. For matching the experimental data, the R-Q model is used as follows [Graj_10]:

$$B_{est}(Q, \emptyset) = \frac{a}{Q^b + c}, \tag{4.1}$$

where:

B_{est} - bitrate or frame size (the number of bits per frame) estimated using the model,

$\emptyset = [a\ b\ c]$ - parameters that depend on sequence content,

Q - quantization step size.

The model parameters can be estimated by minimizing the error between the experimental curves and the curves obtained from the model. The best estimate of the model parameters means getting the smallest error between these curves. The accuracy of the approximation of the experimental data is measured as the following [Graj_10, Choi_13, Lian_13, Guo_15]:

$$Relative\ error\ (Q, \emptyset) = \frac{|B_X\ (Q) - B_{est}(Q, \emptyset)|}{B_X(Q)} \times 100\%, \tag{4.2}$$

where:

$Relative\ error\ (Q, \emptyset)$ - relative approximation error,

$B_X\ (Q)$ - bitrate or frame size of the video,

$B_{est}(Q, \emptyset)$ - approximate bitrate or frame size of the video estimated using the model.

In this chapter, it will be demonstrated that the model from Eq. 4.1 is valid also for HEVC and VVC. Moreover, astonishing relations between the model parameters of the encoders of different types are demonstrated. Obviously, the total number of bits or the bitrate for a given quality of the decoded video is very different for AVC, HEVC, and VVC and it is roughly in the ratio of 4:2:1, respectively [Sull_13, Bros_20]. Nevertheless, when considering the R-Q model, the similarity of the R-Q curves is striking.

4.3 Model Parameters for AVC, HEVC, and VVC Codecs

The R-Q model (Eq.4.1) has been derived independently for each frame type and bitrate. Also, the R-Q model used has three parameters that depend on the sequence content. As mentioned in the previous section, the parameters' values can be estimated by approximation error minimization over the interval of the quantization step (*Q*).

The model parameters are obtained using the trust-region optimization method with the number of iterations equal to 2×10^6 and with the tolerance value of 1×10^{-6} (see Section 3.7). Nevertheless, the choice of the nonlinear optimization method is not crucial for the dissertation, and other methods can be used.

For example, Table (4.1) shows the model parameters and the average relative approximation error for the frame size of the I-frame type for different codecs. More detailed data may be found in Appendix A (Table A.1 to A.7).

Table 4.1: Model parameters and the average relative approximation error for the frame size of I-frame type for different codecs.

Sequence	Codecs	*a*	*b*	*c*	Relative error [%]	
					mean	std. dev.
Ballet	AVC	1061.70	0.87	-1.69	3.94	2.74
	HEVC	765.02	0.88	-1.55	1.09	0.89
	VVC	590.02	0.76	-2.25	1.50	1.11
Poznan_Block2	AVC	8328.46	1.03	-2.52	3.09	2.35
	HEVC	5200	0.93	-3.13	1.63	1.31
	VVC	4200	1.01	-3.35	1.62	1.42

In Table 4.1, it is noticed that parameters *b* and *c* of the model used for all codecs have approximately similar values for the given content. Thus, it has been considered that parameters *b* and *c* for the HEVC and VVC models are similar to parameters *b* and *c* for the AVC model in all experiments. Accordingly, the rate control model for HEVC and VVC based on AVC data can be proposed by discovering the relationship between parameter *a* of the HEVC and VVC models and parameter *a* of the AVC model. This is the rationale for the estimation of HEVC and VVC model parameters using the AVC model parameters.

For the experiments, reference software for AVC, HEVC, and VVC was used, as shown in Section 3.3. A training set of 14 standard MPEG/JVET test sequences (diverse resolutions and frame rates) was used, shown in Table 3.1.

The data were obtained for *QP* values from 25 to 50, which corresponds to practically used bitrates. This *QP* interval corresponds to the interval of the quantization step *Q* from about 11 to 208. The GOP structure used in the experiments for all codecs is I B3 B2 B3 B1 B3 B2 B3

B0 B3 B2 B3 B1 B3 B2 B3 P B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3. The *QP* and *Q* values are the basic values for GOP. The *QP* offsets were set according to the MPEG common test conditions for individual frame types, shown in Table 3.3.

In the experiments, the trust-region optimization method was used. Nevertheless, the choice of the nonlinear optimization method is not crucial for the dissertation, and other methods can be used.

In this section, the approximated curves are estimated using the model parameters (*a*, *b*, and *c*), where the values of parameter *a* of AVC, HEVC, and VVC models were estimated directly from experimental data of the training set (Table 3.1) for AVC, HEVC, and VVC, respectively. While, the values of parameters *b* and *c* of AVC, HEVC, and VVC models were estimated directly from experimental data for AVC, as mentioned above. For the training set of video sequences, as shown in Table 3.1, the mean relative approximation errors for individual frames of various types, as well as for total bitrate, are summarized in Tables 4.2 and 4.3, with more details in Appendix A.

Table 4.2: Mean relative approximation error for the frame size of different frame types for the training set.

Frame type	Relative error [%]					
	AVC		HEVC		VVC	
	mean	std. dev.	mean	std. dev.	mean	std. dev.
I	3.47	2.55	1.85	1.48	2.31	1.97
P	4.58	3.46	2.99	2.15	3.83	3.29
B0	5.30	3.67	8.99	7.82	9.90	9.11
B1	6.28	3.92	8.27	5.92	10.23	8.01
B2	4.28	2.97	16.69	9.79	24.17	15.92
B3	3.61	2.54	17.30	10.04	27.64	17.49

Table 4.3: Mean relative approximation error for the bitrate for the training set.

Bitrate	Relative error [%]					
	AVC		HEVC		VVC	
	mean	std. dev.	mean	std. dev.	mean	std. dev.
GOP	3.99	3.13	13.21	7.24	20.77	14.54

Table 4.2 shows that the accuracy of the model is higher for frames with larger numbers of bits (I and P frames), but the efficiency decreases for frames with smaller numbers of bits (B0, B1, B2, and B3 frames). For the approximation of bitrate, the accuracy is satisfactory for approximate estimation, as shown in Figs. 4.3 and 4.4 and in Figs. B.1 - B.12 in Appendix B.

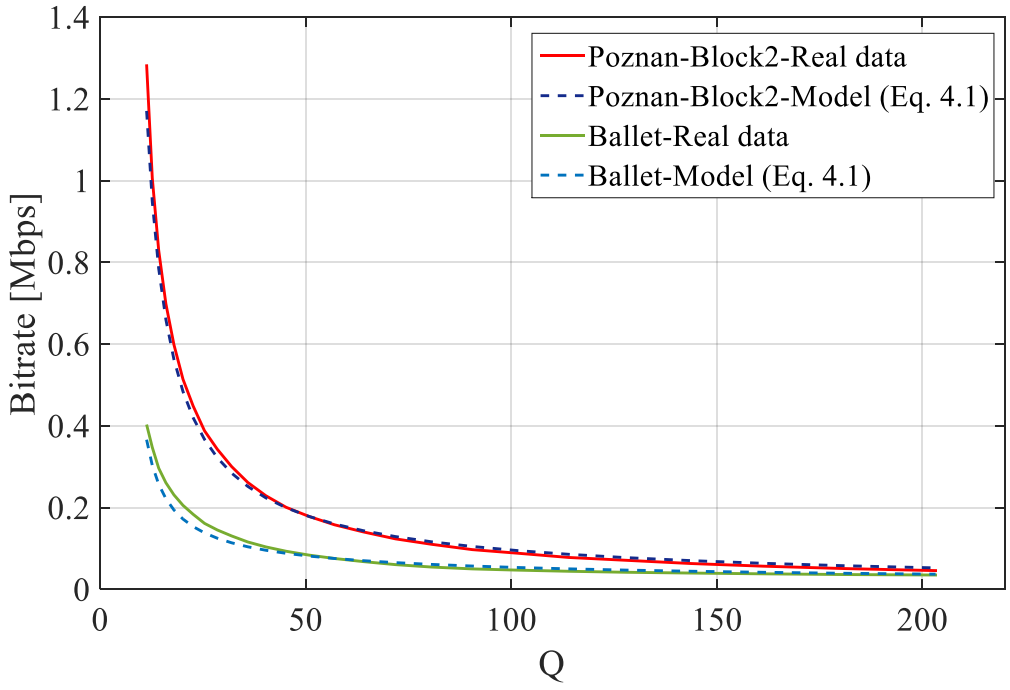


Fig. 4.3. Experimental and approximated curves for bitrate for *Poznan_Block2* and *Ballet* sequences for the HEVC codec.

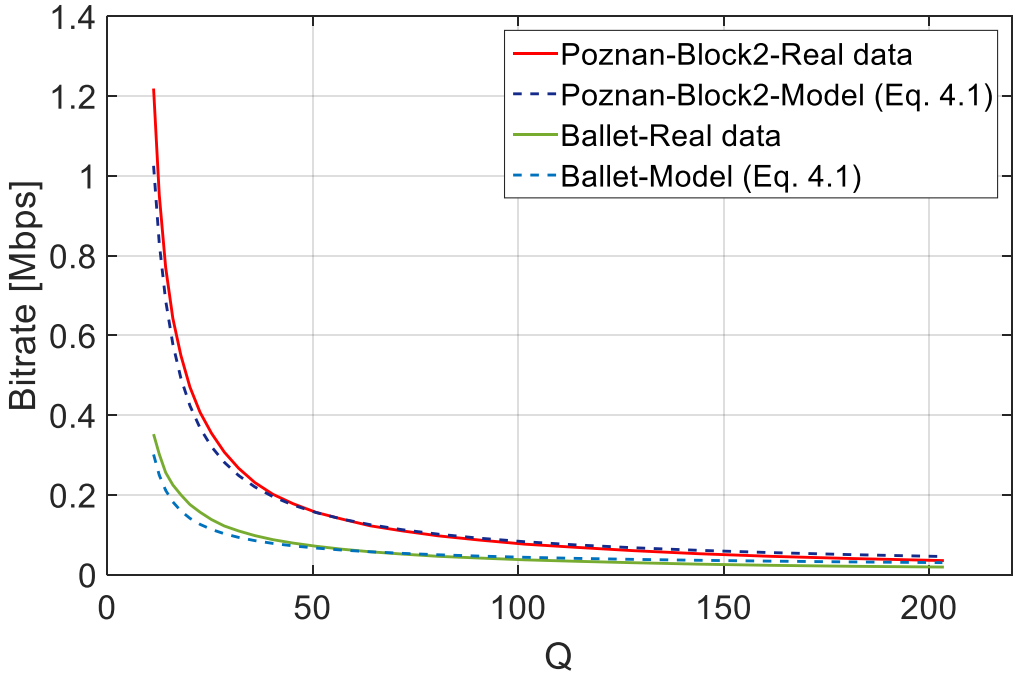


Fig. 4.4. Experimental and approximated curves for bitrate for *Poznan_Block2* and *Ballet* sequences for the VVC codec.

Based on the results obtained in the experiments (Appendix A: Table A.1 to Table A.14), Figs. 4.5 and 4.6 show all *a(AVC)- a(HEVC)* and *a(AVC)-a(VVC)* pairs for the bitrate for all training sequences. Furthermore, Table 4.4 shows the relationship between the values of

parameter *a* of the AVC model and the values of parameter *a* of the HEVC and VVC model for the bitrate for training sequences.

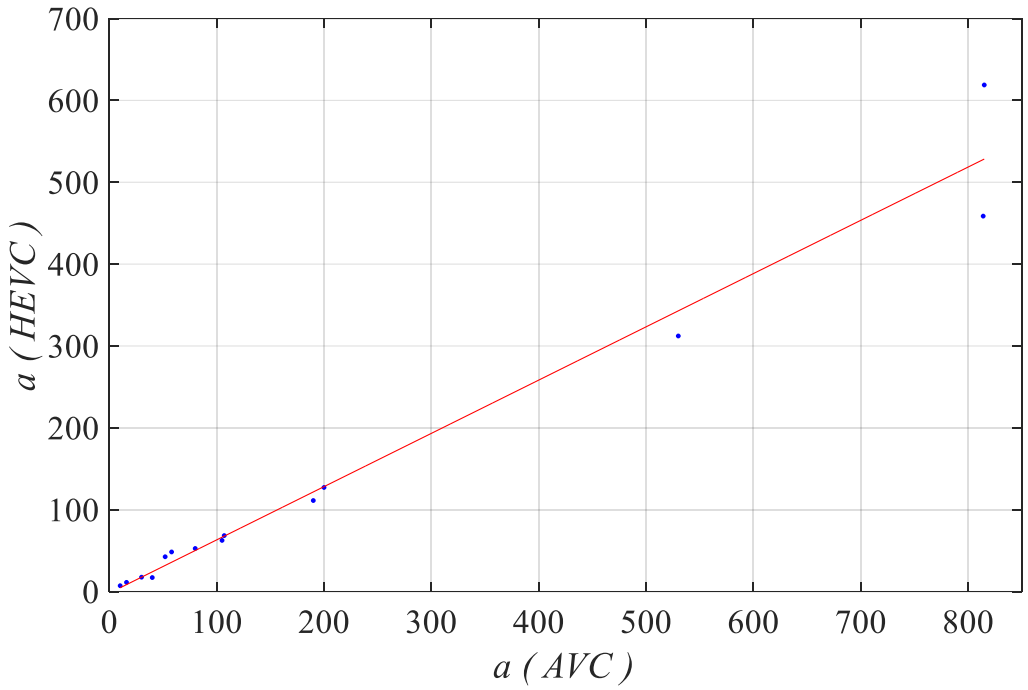


Fig. 4.5. All pairs of *a(AVC)- a(HEVC)* for the bitrate for all training sequences.

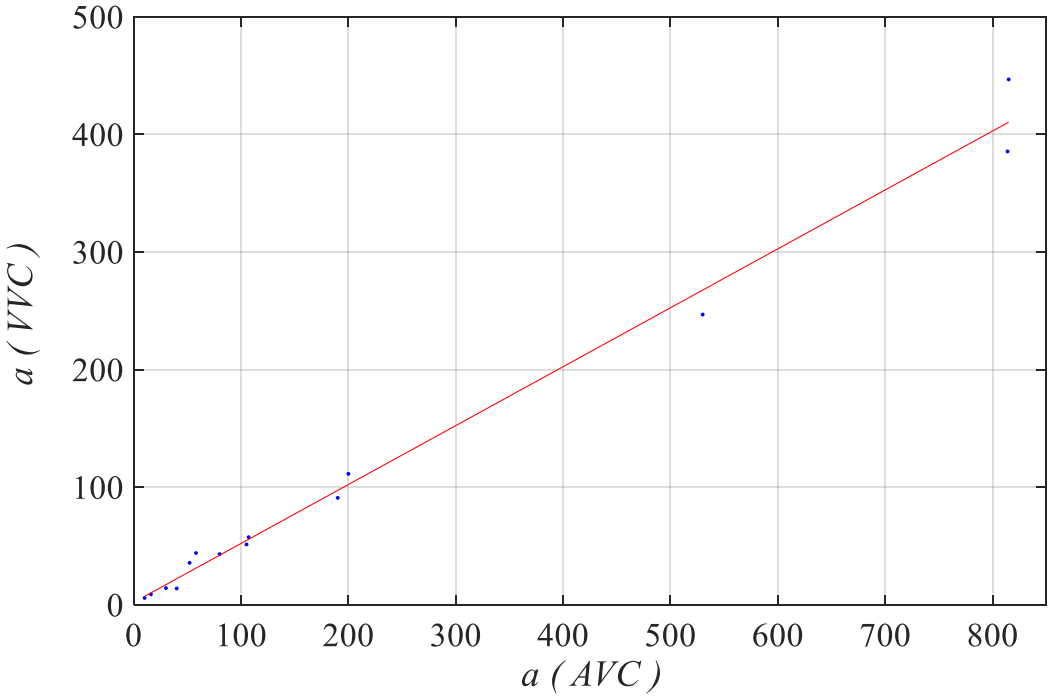


Fig. 4.6. All pairs of *a(AVC)- a(VVC)* for the bitrate for all training sequences.

Table 4.4: The relationship between parameter a of the AVC model and parameter a of the HEVC and VVC models for bitrate (for a GOP).

Sequences	*a(AVC)*	*a(HEVC)*	*a(VVC)*	$\frac{a(HEVC)}{a(AVC)}$	$\frac{a(VVC)}{a(AVC)}$
PeopleOnStreet	815.011	618.851	446.706	0.76	0.55
Traffic	814.018	458.627	385.493	0.56	0.47
Kermit	530	312.190	246.836	0.59	0.47
Poznan_Block2	200.000	127.140	111.384	0.64	0.56
Poznan_Fencing	107.024	68.334	57.497	0.64	0.54
BBB.Butterfly	30.000	17.5106	14.214	0.58	0.47
BBB.Flowers	105	62.461	51.351	0.59	0.49
Ballet	10.1000	7.095	5.8471	0.70	0.58
Breakdancers	16	11.189	8.926	0.70	0.56
Keiba	40.000	17.074	13.908	0.43	0.35
RaceHorses	190.000	111.161	90.950	0.59	0.48
Basketball_Drill	80.003	52.591	43.093	0.66	0.54
Basketball_Pass	52.009	42.467	35.705	0.82	0.69
BQSquare	58	48.352	44.025	0.83	0.76

It has been found that $\frac{a(HEVC)}{a(AVC)}$ and $\frac{a(VVC)}{a(AVC)}$ are roughly similar values for different test sequences, as shown in Table 4.4. In Figures (4.5 and 4.6) and Table 4.4, it has been observed that a linear relationship is most appropriate for describing the relationship between the model's parameters for codecs. Therefore, the linear relationship will be proposed to describe the relationship between *a(AVC)* and *a(HEVC)* and between *a(AVC)* and *a(VVC)*.

4.4. Method of Estimation of the Model Parameters for HEVC and VVC from AVC Data

The goal of this section is to propose a VVC and HEVC encoder model estimated from the AVC model for the same video sequence or the same frame. The VVC or HEVC model can be obtained in the following steps [Doma_21]:

1. Estimation of *a*, *b*, and *c* parameters of Eq. 4.1 for AVC.

2. Calculation of parameter *a* of the VVC and HEVC model from the value of *a* for AVC. Use the following relations:

$$a\,(HEVC) = \alpha_{HEVC} \cdot a\,(AVC), \tag{4.3}$$

$$a\,(VVC) = \alpha_{VVC} \cdot a\,(AVC), \tag{4.4}$$

where α_{HEVC} and α_{VVC} are general constants derived from experimental data for a large training set of test video sequences.

3. For VVC or HEVC, use the following parameters: *a(HEVC)* or *a(VVC)* from Eq. 4.3 or Eq. 4.4, *b(AVC)* and *c(AVC)*.

This way, the parameters of the HEVC and VVC models can be calculated by estimating the parameters for the respective AVC model.

Through experiments with the training set of test sequences, the universal constants α_{HEVC} and α_{VVC} are estimated. Tables 4.5 and 4.6 show the values of α_{HEVC} and α_{VVC} in Eqs. 4.3 and 4.4 for the frame size of various types and bitrate.

Table 4.5: Parameters α (Eqs. 4.3 and 4.4) for the proposed model for the frame size of various types for different codecs.

Frame type	α_{HEVC}	α_{VVC}
I	0.69	0.61
P	0.69	0.59
B0	0.89	0.75
B1	0.85	0.71
B2	0.83	0.66
B3	0.39	0.30

Table 4.6: Parameters α (Eqs. 4.3 and 4.4) for the proposed model for bitrate for different codecs.

Bitrate	α_{HEVC}	α_{VVC}
GOP	0.65	0.54

4.5 Results for the HEVC and VVC Models Derived Using AVC Parameters

The goal of the research is to estimate the accuracy of the HEVC and VVC models obtained according to the procedure from Section 4.4. It means that the model is not derived directly from the experimental data for HEVC or VVC, as in Section 4.3. This time, the model of an HEVC or VVC encoder is obtained from AVC data only.

The verification set, shown in Table 3.1, is used to assess the accuracy of the proposed method in Section 4.4. The verification set includes sequences with different resolutions and diverse frame rates. For the verification sequences, the mean relative approximation errors for individual frames of various types, as well as for the bitrate, are summarized in Tables 4.7 and 4.8. The details of the results in Appendix C. Fig. 4.7 and Fig.4.8, and all figures in Appendix D show experimental and approximated curves for frame sizes of various types and bitrate for *FourPeople* and *BQMall* sequences for HEVC and VVC codecs.

Table 4.7: Mean relative approximation error for the frame size of different frame types for the verification set.

Frame type	Relative error [%]			
	HEVC		VVC	
	mean	std. dev.	mean	std. dev.
I	6.37	1.96	6.86	2.39
P	11.83	4.37	11.88	4.26
B0	15.79	5.74	18.30	6.03
B1	20.67	4.68	22.89	5.47
B2	25.41	7.80	30.15	8.70
B3	28.70	8.63	30.35	10.89

Table 4.8: Mean relative approximation error for bitrate for the verification set.

Bitrate	Relative error [%]			
	HEVC		VVC	
	mean	std. dev.	mean	std. dev.
GOP	14.77	6.61	18.96	9.36

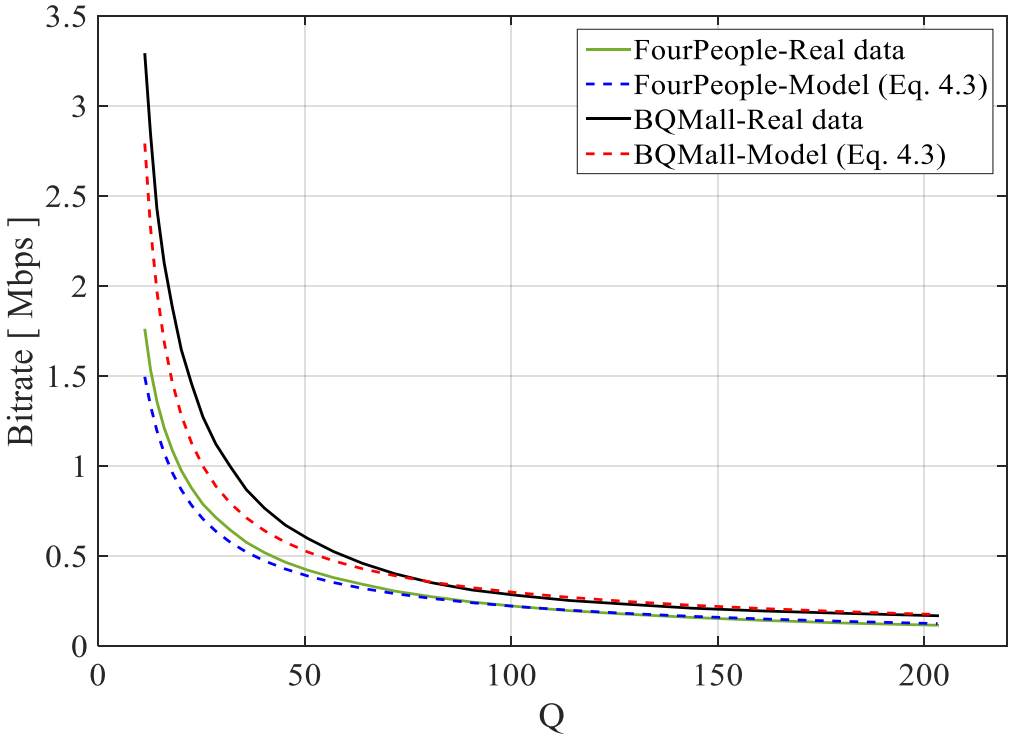


Fig. 4.7. The experimental and approximated curves for bitrate for *FourPeople* and *BQMall* sequences for the HEVC coding.

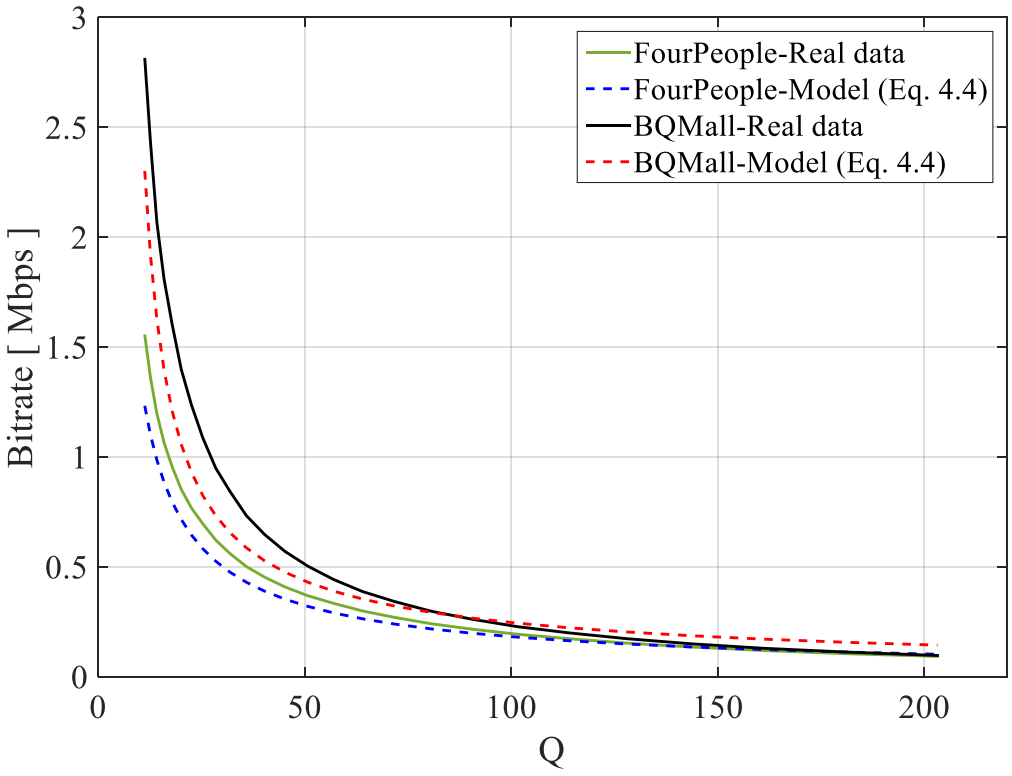


Fig. 4.8. The experimental and approximated curves for bitrate for *FourPeople* and *BQMall* sequences for the VVC coding.

Based on the experiments, the proposed method in Section 4.4 can be used for both bitrate estimation and the estimation of the frame size for a given frame. In both cases, the estimation of three parameters by curve fitting requires at least three encodings. In the proposed approach, HEVC or VVC encodings are replaced by AVC encodings. This approach significantly reduces the estimation time, as each video encoding is 5-10 times faster for AVC as compared to VVC [Topi_19]. Also, HEVC and VVC encoders were compared to the AVC encoder in the experiments, and it has been found that the AVC encoder was about 1-2 and 2-9 times simpler than HEVC and VVC encoders, respectively (shown in Table E.1 in Appendix E). The respective reduction is smaller for HEVC; therefore, the approach is more interesting for VVC than for HEVC.

4.6 Conclusions

The R-Q model is proposed to implement rate control for the codecs considered in this chapter. In addition, a new approach is presented to estimate the parameters of the rate control model for HEVC and VVC by depending on AVC data. The experimental data illustrate that the model accurately describes the bitrate as a function of the quantization step for VVC and HEVC encoders in the most frequently used intervals of quantization parameters (*QP* values from 25 to 50).

An interesting original observation is that the models for AVC, HEVC, and VVC are closely related to each other. Obviously, the bitrates for AVC are the highest, whereas the bitrates for VVC are the lowest, but the shapes of the rate-distortion curves are roughly similar. Thus, the estimation of a single model parameter for the VVC or HEVC model is sufficient to estimate the 3-parameter model knowing the AVC model for the same content.

For VVC or HEVC encoders, the next original achievement is the proposal to estimate the bitrate or the number of bits in a frame using the respective experimental data obtained for the AVC encoder. As the estimation of the three parameters of the model requires at least three encodings, such an approach is very interesting for VVC, because VVC encoding (of a sequence or a single frame) is 2-9 times more complex than AVC encoding (see Appendix E).

The accuracy of the proposed approach is high for the I frame and P frame (large frames), while it is significantly lower for B frames. As B frame sizes are small, the error can be clearly observed for any small difference between the experimental and approximated data (see Tables C.8-14 and Figs D.1-12 in Appendix C and Appendix D, respectively).

Chapter Five

Views – Depths Bitrate Allocation for Stereoscopic Video

5.1 Motivation and Goal

Compression of multiview plus depth (MVD) video (mentioned in Section 2.7) has a significant effect on the quality of synthesized views [Mull_11, Alob_18a]. As mentioned in Section 2.10, the quality of virtual views depends both on the quality of views and on the quality of depth maps. Moreover, the bitrate allocation between views and depth maps has a major impact on the quality of the virtual views synthesized from decoded views and depth maps. Therefore, in practical video encoder control, there is a need to decide how to allocate the available bit budget to views and depth maps.

In Chapter 1, the importance of advanced stereoscopic systems was mentioned. The importance of such systems implies the importance of compression of stereoscopic video plus depth. This chapter deals with the bitrate allocation in compression of stereoscopic plus depth video. Compression of two-view plus depth video may produce different qualities of a virtual view at a specified bitrate depending on the quantization steps for both video and depth. Therefore, the essential requirement in the compression of stereoscopic video plus depth is to select the pairs of the quantization steps for both video and depth for the best quality of the virtual view for a given bitrate. As discussed in Chapter 2, the quantization step in modern video encoders (HEVC and VVC) is determined by the quantization parameter (a quantization parameter is an integer number), as shown in Eq. 2.1, which means that quantization steps for video and depth are determined by the quantization parameters for video (*QP*) and depth (*QD*), respectively.

The task of bitrate allocation to views and depth maps may be solved by finding a formula for *QD* as a function of *QP* to achieve the best quality of the virtual view for a given bitrate as follows:

$$QD = f(QP). \tag{5.1}$$

In this chapter, we use a hypothesis that a reasonably accurate approximation of Eq. 5.1 exists as a general formula with the coefficients independent from the video content. Thus, this means that the formula can be derived using a training set of stereoscopic video sequences (as shown in Table 3.2). Also, for simplicity, it is assumed that the quantization parameter *QP* is constant for all views, and the quantization parameter *QD* is constant for all depth maps for each view, respectively.

The compression efficiency of codecs that use the proposed bit allocation procedure will be estimated by the quality of the views synthesized from the stereoscopic video plus depth obtained at the output of a video codec that exploits the proposed bitrate allocation procedure. These measurements use another set of stereoscopic video sequences (the verification set from Table 3.2).

For the considerations of this chapter, the quality of the virtual view will be measured by PSNR [Wink_05, Joshi_15] and IV-PSNR [Dzie_20, Dzie_22] by comparing the virtual view synthesized from decoded views and depth maps with that obtained by a real camera at exactly the same spatial position (see Section 3.5). The virtual view position to be synthesized will always be chosen in-between input views. For simplicity, PSNR is mostly calculated only for the luma component.

In order to derive the model with its parameters, the experimental data are collected for all training sequences, i.e., quality measurements and bitrates for (*QP*, *QD*) pairs are collected from reasonable intervals of *QP* and *QD*. The Pareto-optimum $(QP, QD)_{opt}$ pairs that form an optimum R-D (rate-distortion) curve are chosen. The set Π of $(QP, QD)_{opt}$ pairs is approximated by a $QD = f(QP)$ function.

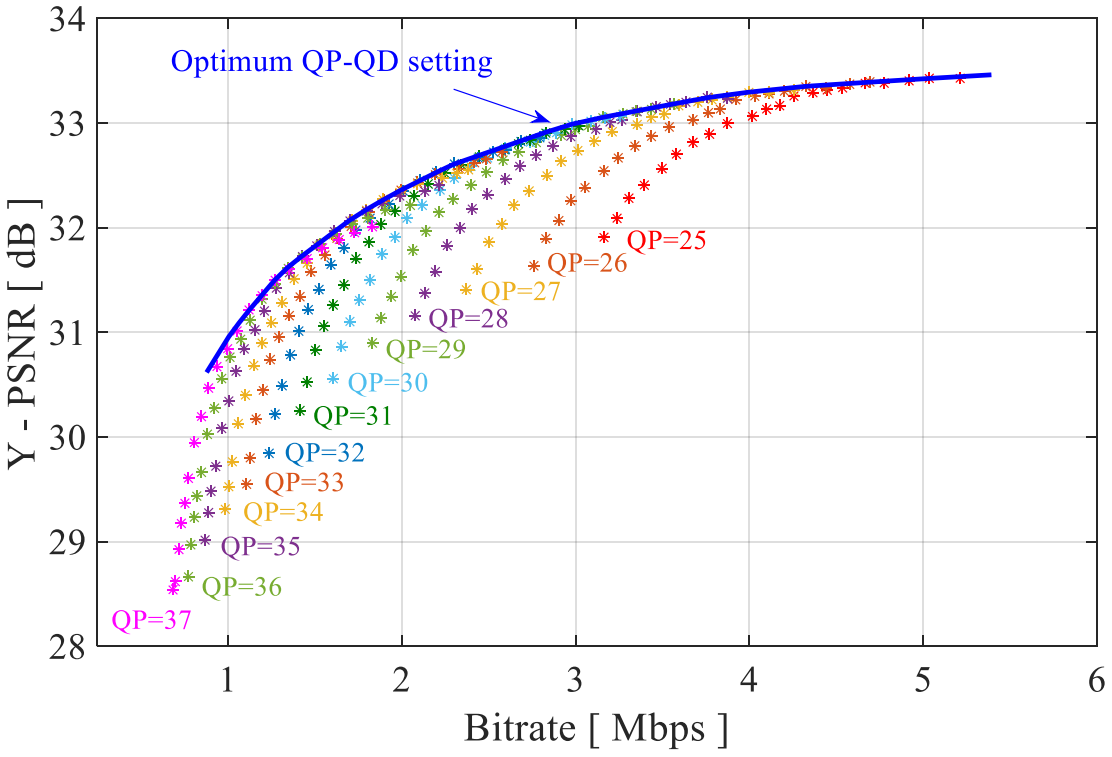
5.2 Estimation of the Set Π of the Optimum (QP, QD) Pairs

In order to derive the formula for bitrate allocation (Eq. 5.1), the optimum (*QP*, *QD*) pairs ((QP, QD) pairs that belong to Π) for all training sequences (shown in Table 3.2) should be found, as in [Alob_18b, Alob_19, Alob_20]. All possible quantization parameter pairs (QP, QD) are tested in the range of 15 to 51 in order to find the optimum *QP*-*QD* settings according to the block diagram presented in Fig 3.2, which means that two views and two depths have been encoded, decoded, and synthesized for each pair. Thus, the gathering of experimental data in order to pick up the set Π of (QP, QD) pairs such that:

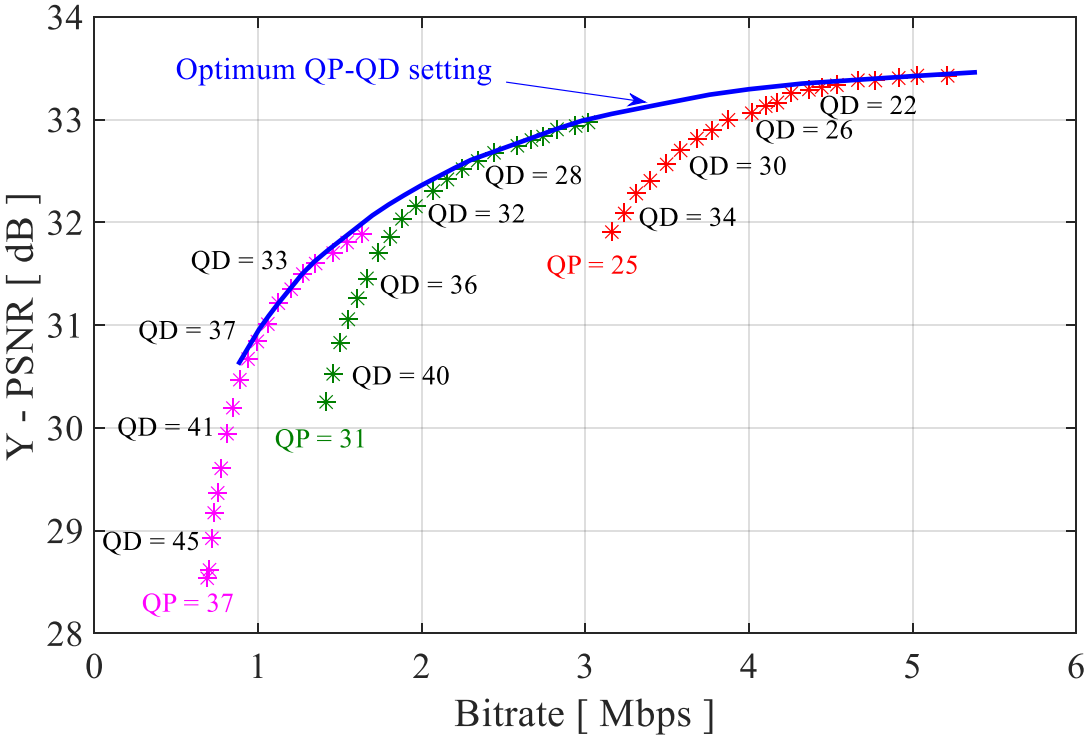
- (QP, QD) pair corresponds to bitrate $B(QP, QD)$ and quality $PSNR(QP, QD)$;
- For any $(QP, QD)_i$ and $(QP, QD)_j$ belonging to Π, there exists no other pair (QP, QD) such that it achieves:

$(PSNR(QP, QD) > PSNR(QP, QD)_i$ or $PSNR(QP, QD) > PSNR(QP, QD)_j)$ and $(B\ (QP, QD)_i \leq B\ (QP, QD) \leq B\ (QP, QD)_j)$.

In Fig. 5.1, colored points represent the results of virtual view synthesis with the use of encoded views and depth maps with all the possible combinations of quantization parameters for views and depth maps; each color represents a constant value of *QP* and different values of *QD*. In contrast, the blue line represents the optimum (*QP*, *QD*) pairs. The optimum (*QP*, *QD*) pairs belong to the envelope over the cloud of PSNR-bitrate points that form the best R-D (rate-distortion) curve (Appendix F).



(a)



(b)

Fig. 5.1. The best R-D curve obtained from the optimum (*QP* , *QD*) pairs of *BBB.Flowers* sequence for the HEVC codec for given *QP* values with different *QD* values:
(a) $QP \in [25-37] \in \mathbb{N}$., where $\mathbb{N}$ is the set of natural numbers. (b) $QP \in \{25, 31, 37\}$.

5.3 Derivation of the Analytic Model $QD = f(QP)$

In Section 5.2, the proposed method is described to derive Pareto-optimum (*QP*, *QD*) pairs in the sense of the minimum bitrate for a given quality of the synthesized view or the best quality of the synthesized view for a given bitrate. Consequently, the bitrate allocation model (Eq. 5.1) can be derived based on the optimum (*QP*, *QD*) pairs obtained in the previous section. In that way, the relation $QD = f(QP)$ can be derived according to polynomial regression:

- A first-order polynomial regression relationship between both quantization parameters is:
$$QD = \mu \cdot QP + \beta \ . \tag{5.2}$$
- A second-order polynomial regression relationship between both quantization parameters is:
$$QD = \rho \cdot QP^2 + \sigma \cdot QP + \tau \ . \tag{5.3}$$
- A third-order polynomial regression relationship between both quantization parameters is:
$$QD = \varphi \cdot QP^3 + \phi \cdot QP^2 + \chi \cdot QP + \psi. \tag{5.4}$$

The parameters of the polynomial regression [Wei_05, Roys_08, Yan_09, Cela_16] are estimated using the least-squares fitting to the optimum (*QP*, *QD*) pairs. Table 5.1 shows the parameters of the polynomial regression for the training sequences (mentioned in Table 3.2).

Table 5.1: Parameters of polynomial regression models (Eqs. 5.2, 5.3, and 5.4) for optimum (*QP*, *QD*) pairs for HEVC coding.

Sequences	First-order regression (Eq. 5.2)		Second-order regression (Eq. 5.3)			Third-order regression (Eq. 5.4)			
	μ	β	ρ	σ	τ	φ	ϕ	χ	ψ
Ballet	1.38	-18.48	0.01	0.86	-8.98	-0.003	0.37	-12.42	146.16
Breakdancers	1.27	-9.00	-0.04	4.25	-62.27	-0.002	0.14	-2.50	17.85
BBB.Butterfly	1.17	-11.96	-0.01	2.26	-32.28	0.002	-0.24	10.81	-135.68
BBB.Flowers	1.22	-12.28	-0.01	2.06	-27.53	-0.001	-0.01	1.77	-24.14
Poznan_CarPark	1.44	-18.65	-0.04	4.30	-69.89	-0.001	-0.01	3.30	-57.95
Kermit	1.13	-11.43	-0.02	2.38	-34.94	0.002	-0.23	10.51	-133.74

Fig. 5.2 shows the approximate relationship between *QP* and *QD* for the optimum pairs for two training sequences using polynomial regression.

In Table 5.1, it has been noticed that the parameters of Eq. 5.2 for the training sequences have roughly similar values, which means a general model ($QD = f(QP)$) for MVD sequences

can be derived, while the parameters of Eqs. 5.3 and 5.4 have different values (i.e., a general model cannot be derived). In Fig. 5.2, it has been observed that the curves of Eq. 5.2 are similar for all sequences, while the curves of Eqs. 5.3 and 5.4 are quite different for each sequence. The advantage of the first-order polynomial regression models is related to their simplicity. Due to the abovementioned reasons, it is suitable to use first-order polynomial regression to derive the *QD* model based on *QP* for each codec.

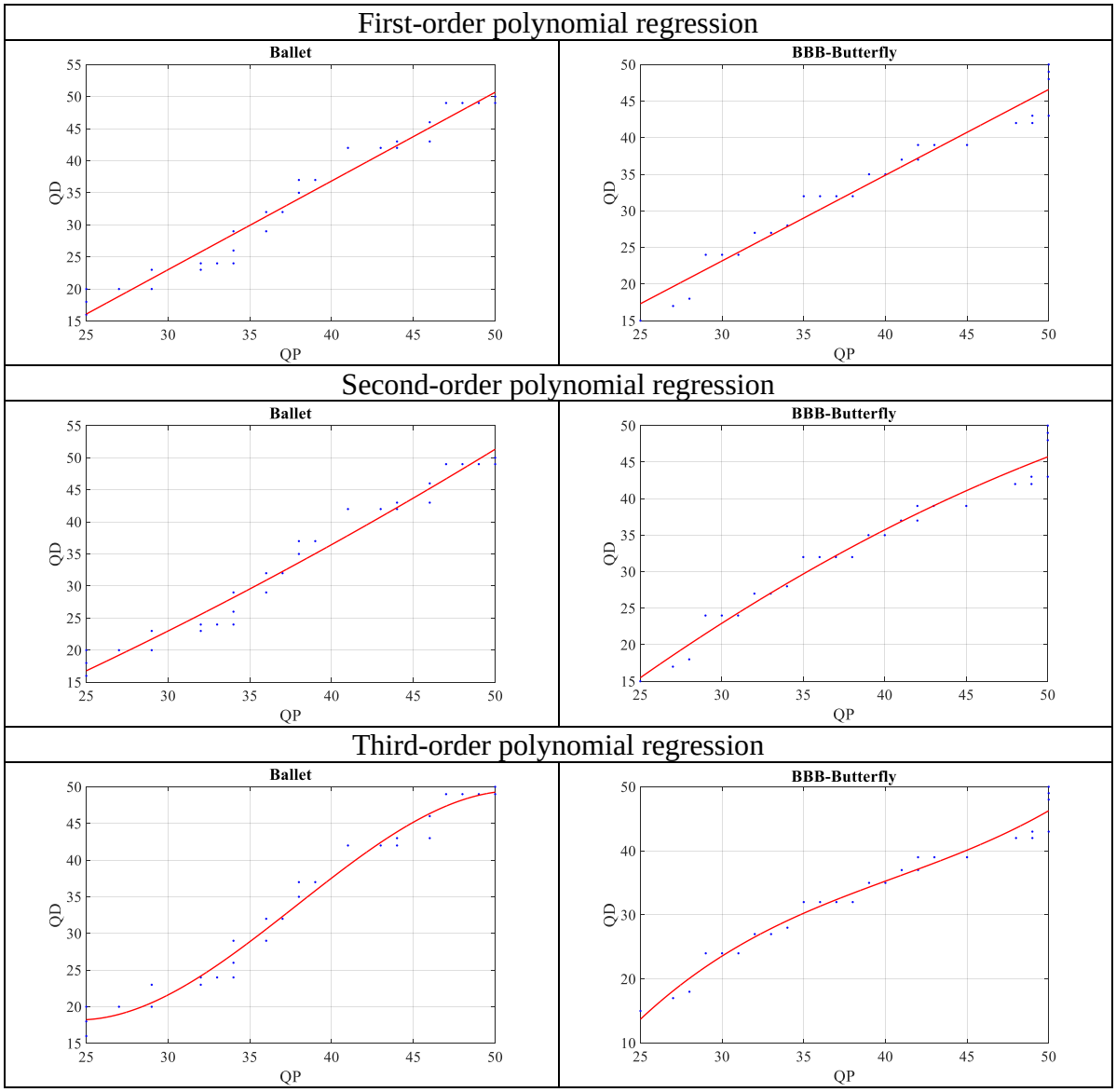


Fig. 5.2. The approximate relationship between *QP* and *QD* for the optimum pairs for *Ballet* and *BBB.Butterfly* sequences with the use of polynomial regression. The dots represent optimum (*QP*, *QD*) pairs for *Ballet* and *BBB.Butterfly* sequences. The red curves represent curves obtained from polynomial regression models.

Fig 5.2 demonstrates that the values of *QP* (the quantization parameters common for both views) and *QD* (the quantization parameter common for both depth maps) are strongly correlated for the optimum (*QP*, *QD*) pairs.

5.4 Bitrate Allocation Models

5.4.1 Introduction

In order to derive a global model $QD = f(QP)$, the experiments have been performed using several test multiview video sequences with depth maps (as in Table 3.2).

The experiments and measurements were organized according to the block diagram presented in Fig 3.2. The reference software for codecs (HEVC, VVC, MV-HEVC, and 3D-HEVC) has also been used to encode and decode views and depth maps (see Section 3.3). The reference model software VSRS v.3.5 [Stan_13b] has been used for view synthesis, as mentioned in Section 3.4.

5.4.2 Specific Models for Individual Codec Types

The parameters of the first-order model (μ and β in Eq. 5.2) are estimated using the least-squares fitting to the optimum (*QP*, *QD*) pairs for the training test sequences for different codecs (shown in Figs. G.1 - G.24 in Appendix G). The parameter values (μ and β) estimated individually for different codecs are collected in Table 5.2 for the training test sequences.

Table 5.2: Model parameters derived individually for training test sequences and various codecs.

Sequence	HEVC		VVC		MV-HEVC		3D-HEVC	
	μ	β	μ	β	μ	β	μ	β
Ballet	1.38	-18.48	1.15	-6.15	1.39	-15.28	1.13	-1.64
Breakdancers	1.27	-9.00	1.30	-10.65	1.26	-8.43	1.17	-4.19
BBB.Butterfly	1.17	-11.96	1.19	-12.86	1.10	-7.86	1.05	-3.84
BBB.Flowers	1.22	-12.28	1.18	-9.91	1.26	-11.97	1.06	-0.44
Poznan_CarPark	1.44	-18.65	1.50	-21.27	1.41	-14.29	1.25	-7.55
Kermit	1.13	-11.44	1.22	-14.58	1.15	-11.21	1.20	-11.34
Average	1.20	-11.27	1.22	-11.25	1.20	-9.41	1.11	-3.40

For a given codec, the values of parameters for the training sequences have nearly the same values. This yields a model with the parameters being constants but specific for the individual codec types. Therefore, the parameter values of the model obtained individually for different codecs can be averaged over six training test sequences as summarized in Table 5.2 (shown in

Figs. G.25 - G.28 in Appendix G). Consequently, for codec modeling, the average values of model parameters (μ and β) are used.

5.4.3 Proposed Global Model for Codecs

In Table 5.2, it has been noticed that model parameters (μ and β) for different codecs and training sequences have similar values; therefore, a global model for codecs can be proposed. Based on the least-squares fitting, the parameters (μ and β) of the global model are estimated for the optimum (*QP*, *QD*) pairs for all training sequences and all codecs, as shown in Fig. G.29 presented in Appendix G. The value of the global model parameters obtained on average over all codecs and all training test sequences are shown in Table 5.3.

Table 5.3: Global proposed model parameters for all codecs

	μ	β
Global model	1.17	-8.41

5.5 Assessment of the Proposed Models

In this section, experimental assessment is provided for the specific models (Section 5.4.2) and for the global model (Section 5.4.3). Firstly, we assess the exactness of the proposed specific models, i.e. the models that are specific to the codec type and the content. The quality of the synthesized virtual views achieved using the proposed specific models (the parameter values for Eq. 5.2 are shown in Tables 5.2 and 5.3) is compared to the reference approach and optimum (*QP*, *QD*) pairs for many different test sequences for the verification set (shown in Table 3.2). The reference approach for independent codecs is $QP = QD$, while the reference approach for joint coding is defined in CTC (Common Test Conditions) [Mull_14]. Tables 5.4 and 5.5 show the assessment of models dedicated to the given compression technology. The coding efficiency of the given technology is assessed by calculating the average difference between the curves for PSNR (*ΔPSNR*) and bitrate (*ΔBitrate*) just as an extension of the well-known Bjøntegaard metric (shown in Section 3.6) to work with more than four points.

Table 5.4: Bitrate reductions calculated by Bjøntegaard rates between the specific model for different codecs (Eq. 5.2 with average parameter values shown in Table 5.2) against coding with reference and optimum (*QP*, *QD*) pairs for verification sequences.

Codec	Specific model for different codecs vs reference approach		Specific model for different codecs vs optimum (*QP*, *QD*) pairs	
	ΔPSNR [dB]	ΔBitrate [%]	ΔPSNR [dB]	ΔBitrate [%]
Poznan_Block2				
HEVC	0.10	-15.89	-0.13	26.70
VVC	0.06	-13.97	-0.13	41.50
MV-HEVC	0.07	-8.34	-0.33	52.10
3D-HEVC	0.28	-34.36	-0.22	37.17
Poznan_Fencing				
HEVC	0.04	-17.81	-0.09	54.03
VVC	0.02	-10.40	-0.06	31.51
MV-HEVC	0.02	-7.74	-0.10	53.21
3D-HEVC	0.05	-23.05	-0.06	40.26

From the results shown in Table 5.4, it was noticed that the usage of the specific models proposed for different codecs (Eq. 5.2 with average parameter values shown in Table 5.2) resulted in a decrease in total bitrate and an improvement in the virtual view quality for sequences when compared to the reference approach. This means that the choice of the quantization parameters using the special model for different codecs outperforms the reference approach for the bitrate allocation for stereoscopic sequences. The specific models for different codecs (Eq. 5.2 with average parameter values shown in Table 5.2) with optimum (*QP*, *QD*) pairs for the verification sequences led to an increase in total bitrate and a decrease in virtual view quality.

Table 5.5: Bitrate reductions calculated by Bjøntegaard rates between the global proposed model for codecs (Eq. 5.2 with parameter values shown in Table 5.3) against coding with reference and optimum (*QP*, *QD*) pairs for verification sequences.

Codec	Global model vs reference approach		Global model vs optimum (*QP*, *QD*) pairs	
	ΔPSNR [dB]	ΔBitrate [%]	ΔPSNR [dB]	ΔBitrate [%]
Poznan_Block2				
HEVC	0.05	-7.51	-0.18	36.76
VVC	0.04	-9.31	-0.16	49.36
MV-HEVC	0.08	-9.80	-0.32	49.68
3D-HEVC	0.42	-47.10	-0.10	17.51
Poznan_Fencing				
HEVC	0.02	-12.06	-0.11	63.68
VVC	0.01	-3.71	-0.07	45.70
MV-HEVC	0.02	-6.91	-0.10	53.53
3D-HEVC	0.08	-40.16	-0.04	26.65

In Table 5.5, it was observed that the global model (Eq. 5.2 with parameter values shown in Table 5.3) led to a decrease in total bitrate and an improvement in the virtual view quality for sequences when compared to the reference approach. Based on the results shown in Tables 5.4 and 5.5, that is on the comparison of the specific models for different codecs with the reference approach and the comparison of the global model with the reference approach, it can be concluded that the specific models led to a decrease in total bitrate and an improvement in the virtual view quality for sequences when compared to the global model. The specific models obviously outperform the global model, and the global model outperforms the reference approach in the bitrate allocation for stereoscopic sequences.

In Figs. 5.3 and 5.4, the R-D (rate-distortion) curves for the *Poznan_Block2* sequence (HEVC codec) are depicted for the specific and global models. Additional plots are provided in Appendix H. It is worth to remind, that in the figures and in Appendix H, PSNR or IV-PSNR is calculated for the virtual views with the reference to the original (uncompressed) view.

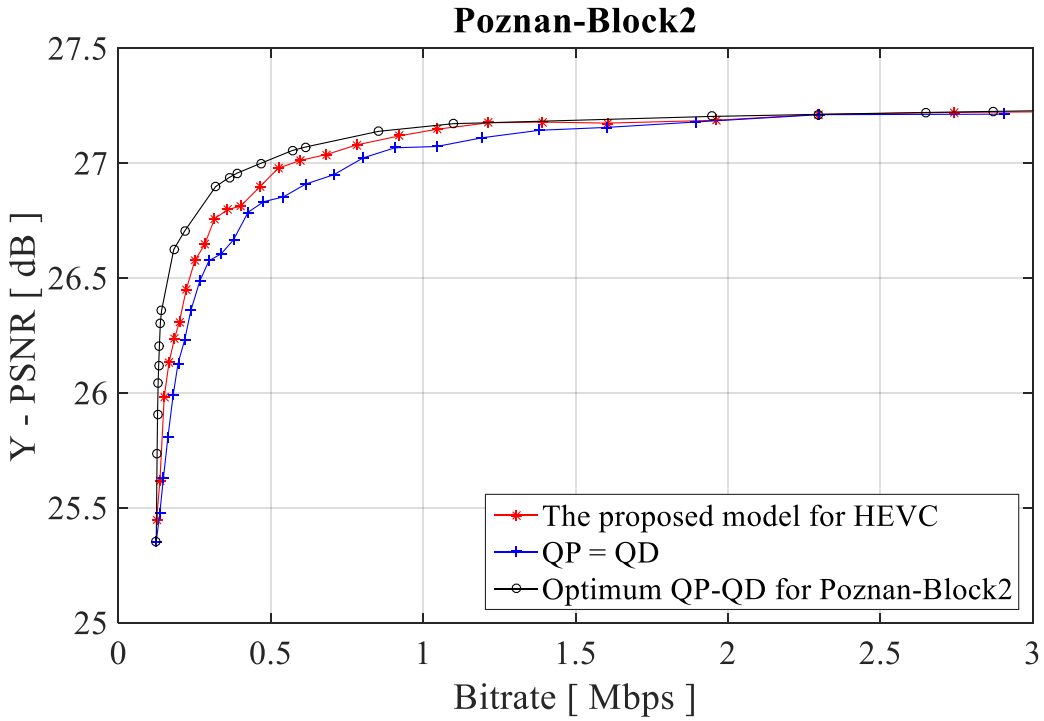


Fig. 5.3. R-D curves for the specific model proposed for HEVC (Eq. 5.2 with average parameter values shown in Table 5.2), the reference approach ($QP = QD$), and the optimum (*QP*, *QD*) pairs for HEVC codec for the *Poznan_Block2* sequence.

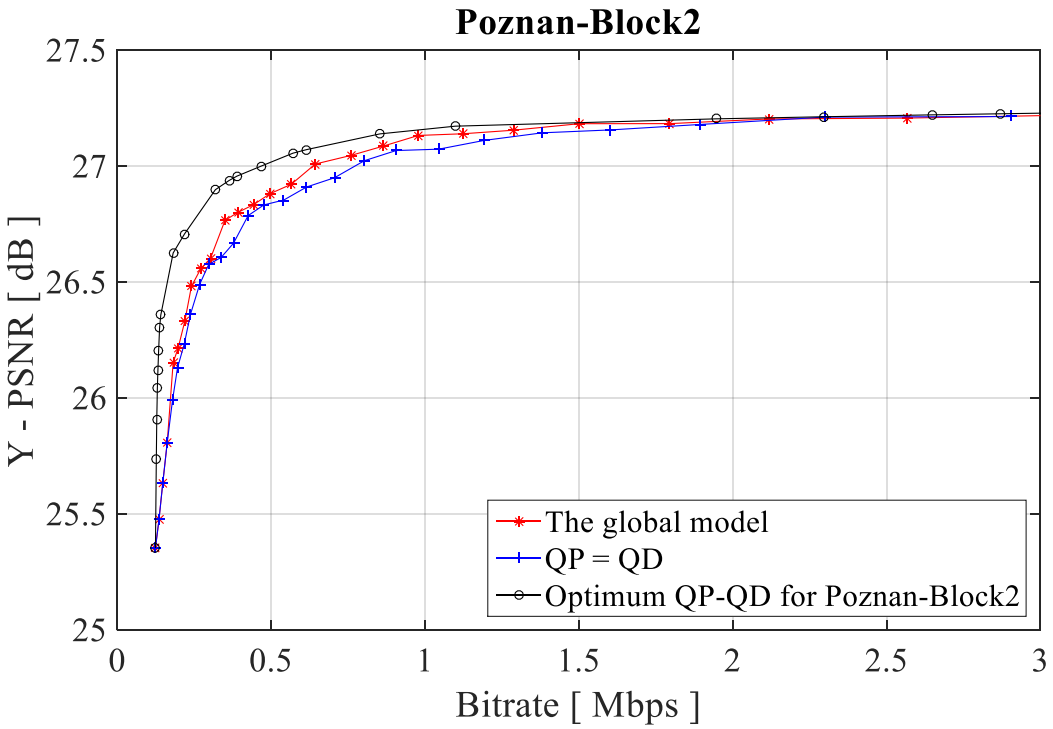


Fig. 5.4. R-D curves for the global proposed model (Eq. 5.2 with parameter values shown in Table 5.3), the reference approach ($QP = QD$), and the optimum (*QP*, *QD*) pairs for HEVC codec for the *Poznan_Block2* sequence.

The global model (Table 5.3) is of particular practical importance. The experimental results demonstrate that it gives a simple rule for bitrate allocation between views and depth that yields higher rate-distortion performance of the video codecs for stereoscopic video plus depth.

Additionally, the specific model (Eq. 5.2 with average parameter values shown in Table 5.2) and the global model (Eq. 5.2 with parameter values shown in Table 5.3) have been compared to the reference approach and the optimum (*QP*, *QD*) pairs for verification sequences by calculating *ΔIV-PSNR* and *ΔBitrate* (Bjøntegaard rates), as shown in Tables 5.6 and 5.7. Figs. 5.5 and 5.6 depict R-D (rate-distortion) curves for the *Poznan_Block2* sequence for HEVC codec to the specific and global model. For more figures, see Appendix H (Figs. H.5-H.8 and Figs. H.15- H.18 in Appendix H).

Table 5.6: Bitrate reductions calculated by Bjøntegaard rates between the specific model for different codecs (Eq. 5.2 with average parameter values shown in Table 5.2) against coding with reference model and the optimum (*QP*, *QD*) pairs for the verification sequences.

Codec	Specific model for different codecs vs Reference approach		Specific model for different codecs vs Optimum (*QP*, *QD*) pairs	
	ΔIV-PSNR [dB]	ΔBitrate [%]	ΔIV-PSNR [dB]	ΔBitrate [%]
Poznan_Block2				
HEVC	0.09	-10.70	-0.10	12.59
VVC	0.06	-8.32	-0.10	16.71
MV-HEVC	0.07	-5.73	-0.26	24.22
3D-HEVC	0.30	-20.79	-0.20	18.50
Poznan_Fencing				
HEVC	0.01	-2.45	-0.08	15.11
VVC	0.01	-1.97	-0.05	11.66
MV-HEVC	0.01	-2.10	-0.07	14.09
3D-HEVC	0.02	-6.07	-0.01	2.04

Table 5.7: Bitrate reductions calculated by Bjøntegaard rates between the global model for codecs (Eq. 5.2 with parameter values shown in Table 5.3) against coding with the reference and the optimum (*QP*, *QD*) pairs for verification sequences.

Codec	Global model vs Reference approach		Global model vs Optimum (*QP*, *QD*) pairs	
	ΔIV-PSNR [dB]	ΔBitrate [%]	ΔIV-PSNR [dB]	ΔBitrate [%]
Poznan_Block2				
HEVC	0.04	-4.78	-0.15	18.64
VVC	0.04	-5.86	-0.12	19.69
MV-HEVC	0.08	-6.77	-0.24	22.85
3D-HEVC	0.44	-31.25	-0.08	8.66
Poznan_Fencing				
HEVC	0.01	-2.24	-0.09	15.82
VVC	0.01	-1.36	-0.05	11.87
MV-HEVC	0.01	-0.24	-0.07	12.88
3D-HEVC	0.02	-7.36	-0.01	3.67

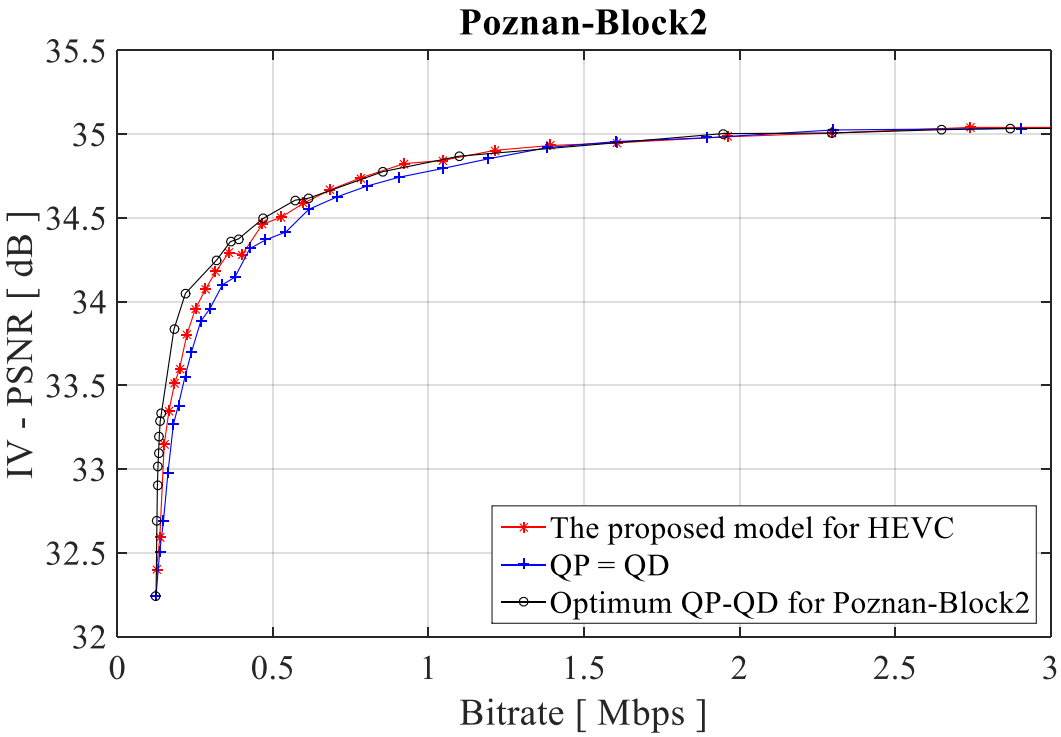


Fig. 5.5. R-D curves for the specific model proposed for HEVC (Eq. 5.2 with average parameter values shown in Table 5.2), the reference approach ($QP = QD$), and the optimum (*QP*, *QD)* pairs for HEVC codec for the *Poznan_Block2* sequence.

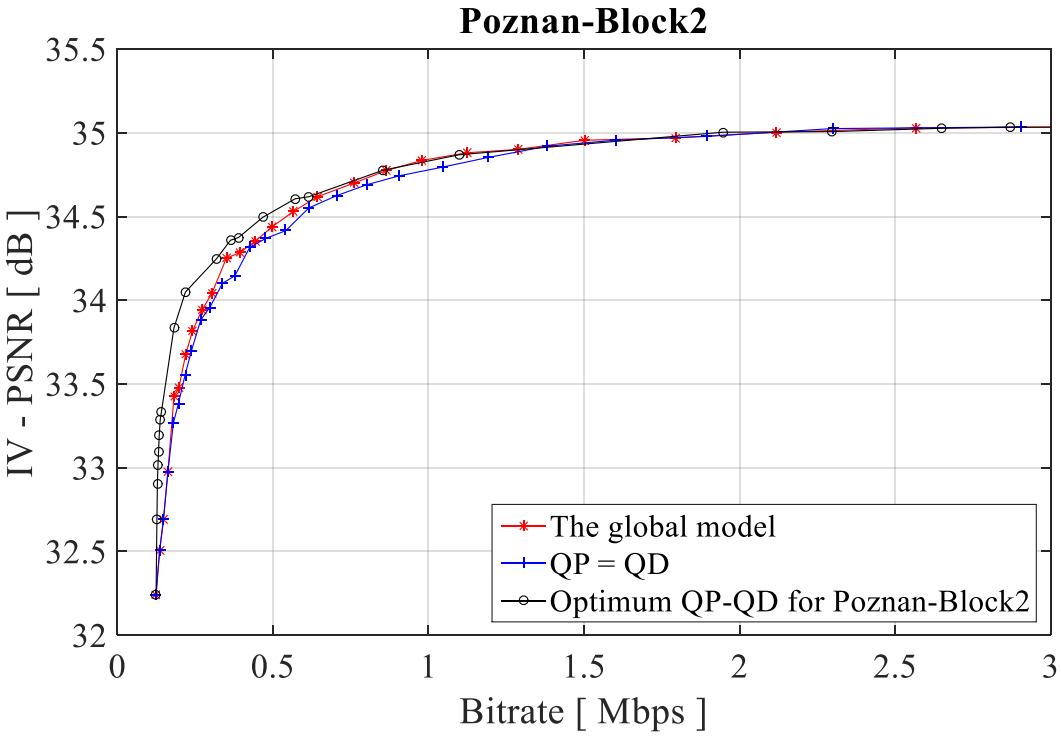


Fig. 5.6. R-D curves for the global proposed model (Eq. 5.2 with parameter values shown in Table 5.3), the reference approach ($QP = QD$), and the optimum (*QP*, *QD*) pairs for HEVC codec for the *Poznan_Block2* sequence.

Regardless of the method used to calculate the quality of the virtual view (PSNR or IV-PSNR), the experiments presented in this section prove that the specific model for individual codec types (Eq. 5.2 with average parameter values shown in Table 5.2) and the global proposed model for codecs (Eq. 5.2 with parameter values shown in Table 5.3) led to a decrease in total bitrate and an increase in virtual view quality for sequences when compared to the reference approach. This means that the specific model and the global model outperform the reference approach in the bitrate allocation for stereoscopic sequences. But comparing these presented models (Eq. 5.2 with parameter values shown in Tables 5.2 and 5.3) with optimum (*QP*, *QD*) pairs for the verification sequences led to an increase in total bitrate and a decrease in virtual view quality, as might be expected. Consequently, the highest quality of the virtual view can only be obtained by appropriate bitrate allocation between views and depth maps.

5.6 Comparison of Models Proposed in This Chapter with Models Presented in Previous Studies of Bit Allocation for MVD

As mentioned in Section 2.9, several models have been proposed to find the bit allocation for multiview video plus depth maps. In [Klim_14a], a second-order equation to describe the relationship between the quantization parameter for views (*QP*) and the quantization parameter for depth maps (*QD*) was proposed, reading as follows:

$$QD = -0.0155 \cdot QP^2 + 2.073 \cdot QP - 14.385. \tag{5.5}$$

Also, the authors in [Klim_14b] presented the following mathematical model of the $QD(QP)$ function:

$$QD = 1.11 \cdot QP + 3.42. \tag{5.6}$$

In [Stan_13a], a first-order model to describe the relationship between the quantization parameters of the video data and depth data was proposed:

$$QD = 1.126 \cdot QP + 2.441. \tag{5.7}$$

Tables 5.8 and 5.9 show $\Delta PSNR$ and $\Delta Bitrate$ calculated by Bjøntegaard rates between the proposed model in Eq. 5.2 (Tables 5.2 and 5.3) and the presented models in the previous studies (Eqs. 5.5, 5.6, and 5.7). In Tables 5.8 and 5.9, a positive number denotes that the proposed model results in an increase in the bitrate or increase in the virtual view quality compared to the bitrate or quality obtained from the models presented in the previous studies (Eqs. 5.5, 5.6, and 5.7), while a negative number indicates a reduction in the bitrate or decrease in the virtual view quality.

Table 5.8: Bitrate reductions calculated by Bjøntegaard rates between the specific model for different codecs (Eq. 5.2 with average parameter values shown in Table 5.2) against coding with the presented models in the previous studies (Eqs. 5.5, 5.6, and 5.7) for the verification sequences (shown in Table 3.2) for different codecs.

Codec	Specific model for different codecs vs the model presented in [Klim_14a] (Eq. 5.5)		Specific model for different codecs vs the model presented in [Klim_14b] (Eq. 5.6)		Specific model for different codecs vs the model presented in [Stan_13a] (Eq. 5.7)	
	ΔPSNR [dB]	ΔBitrate [%]	ΔPSNR [dB]	ΔBitrate [%]	ΔPSNR [dB]	ΔBitrate [%]
Poznan_Block2						
HEVC	0.28	-37.87	0.51	-58.27	0.58	-62.75
VVC	0.18	-36.51	0.33	-58.30	0.37	-62.47
MV-HEVC	0.36	-35.95	0.72	-57.55	0.80	-61.40
3D-HEVC	0.13	-18.11	0.31	-39.06	0.33	-40.40
Poznan_Fencing						
HEVC	0.10	-38.88	0.17	-57.97	0.19	-61.87
VVC	0.10	-33.65	0.19	-52.74	0.22	-56.97
MV-HEVC	0.10	-36.99	0.21	-59.87	0.24	-61.27
3D-HEVC	0.02	-17.68	0.08	-30.06	0.08	-27.61

Table 5.9: Bitrate reductions calculated by Bjøntegaard rates between the global model for codecs (Eq. 5.2 with parameter values shown in Table 5.3) against coding with the presented models in the previous studies (Eqs. 5.5, 5.6, and 5.7) for the verification sequences for different codecs.

Codec	Global model vs the model presented in [Klim_14a] (Eq. 5.5)		Global model vs the model presented in [Klim_14b] (Eq. 5.6)		Global model vs the model presented in [Stan_13a] (Eq. 5.7)	
	ΔPSNR [dB]	ΔBitrate [%]	ΔPSNR [dB]	ΔBitrate [%]	ΔPSNR [dB]	ΔBitrate [%]
Poznan_Block2						
HEVC	0.22	-30.72	0.46	-52.81	0.52	-57.71
VVC	0.16	-33.11	0.31	-56.13	0.35	-60.52
MV-HEVC	0.37	-37.06	0.73	-58.31	0.81	-62.11
3D-HEVC	0.25	-32.00	0.43	-49.97	0.44	-51.16
Poznan_Fencing						
HEVC	0.09	-34.50	0.15	-52.80	0.18	-57.75
VVC	0.09	-30.04	0.18	-50.22	0.20	-54.68
MV-HEVC	0.10	-36.45	0.21	-59.49	0.24	-60.91
3D-HEVC	0.05	-30.04	0.11	-38.64	0.11	-36.51

Figs. 5.7 and 5.8 show R-D (rate-distortion) curves for the *Poznan_Block2* sequence for codecs for the proposed model in this chapter (Eq. 5.2 with parameter values shown in Tables 5.2 and 5.3) and the presented models in the previous studies (Eqs. 5.5, 5.6, and 5.7).

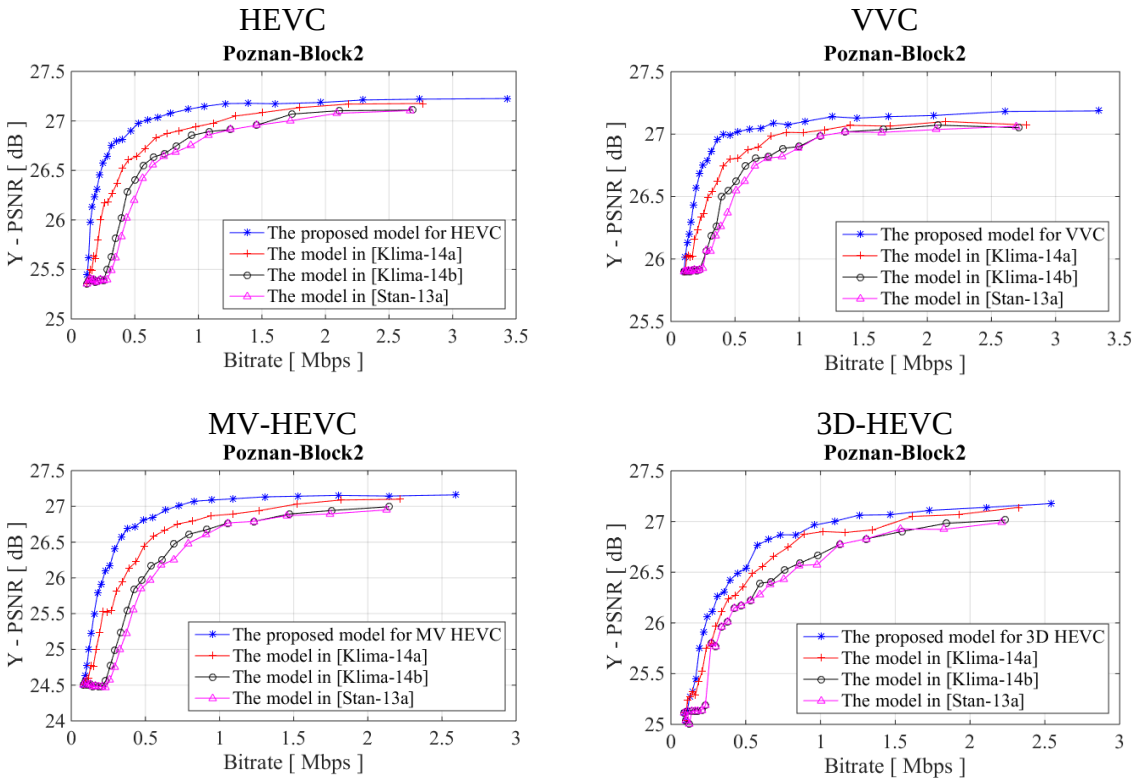


Fig. 5.7. R-D curves comparison between the specific model proposed for different codecs (Eq. 5.2 with average parameter values shown in Table 5.2) and the presented models in the previous studies (Eqs. 5.5, 5.6, and 5.7) for the *Poznan_Block2* sequence.

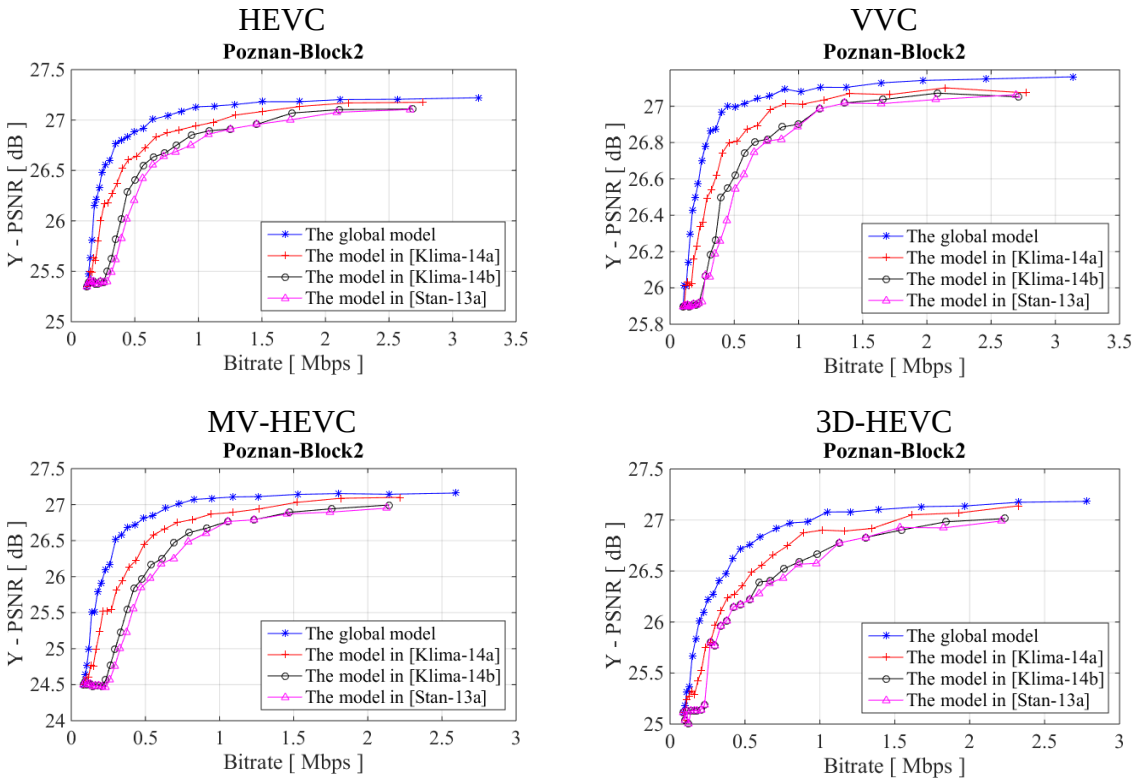


Fig. 5.8. R-D curves comparison between the global model for codecs (Eq. 5.2 with parameter values shown in Table 5.3) and the models in the previous studies (Eqs. 5.5, 5.6, and 5.7) for the *Poznan_Block2* sequence.

Based on the experiments (Tables 5.8 and 5.9, and Figs. 5.7 and 5.8), it has been noticed that the specific model (Eq. 5.2 with average parameter values shown in Table 5.2) and global proposed models (Eq. 5.2 with parameter values shown in Table 5.3) led to a decrease in the total bitrate and an increase in the virtual view quality of the sequences compared to the models presented in previous studies (Eqs. 5.5, 5.6, and 5.7). Thus, these results prove that the specific and global proposed models outperform the models presented in the previous studies of bitrate allocation for MVD.

This section provides the rules for the calculation of the *QD* value as a function of *QP*. Therefore, the bitrate control for stereoscopic video plus depth may be again treated as the bitrate control with one control parameter, i.e. the single quantization parameter.

5.7 A General Model for the Bitrate Ratio for Videos in Total Bitrate of Stereoscopic Video as a Function of *QP*

This section introduces a model for bitrate analysis between videos and depth maps based on the quantization parameter (*QP*) applied to the videos. Optimum (*QP*, *QD*) pairs have been determined in Section 5.2. Currently, it is required to check what bitrate part represents the views and depth maps. According to the data collected for the optimum (*QP*, *QD*) pairs, the relationship between the bitrate ratio for videos from total bitrate (the bitrate of two views plus the bitrate of two depth maps) and *QP* is derived.

The proposed model is derived as a formula for $\frac{View_{Bitrate}}{Total_{Bitrate}}$ value derivation based on *QP* settings:

$$\frac{View_{Bitrate}}{Total_{Bitrate}} = f(QP), \tag{5.8}$$

where:

$View_{Bitrate}$ - bitrate for two views,

$Total_{Bitrate}$ - bitrate for two views and two depth maps,

QP - quantization parameter for video.

Several models can be suggested to find the relationships between $\frac{View_{Bitrate}}{Total_{Bitrate}}$ and *QP* for HEVC coding based on polynomial regression (shown in Table 5.10 and Fig. 5.9):

- A first-order polynomial regression relationship is:

$$\frac{View_{Bitrate}}{Total_{Bitrate}} = a_1 \cdot QP + a_2. \tag{5.9}$$

- A second-order polynomial regression relationship is:

$$\frac{View_{Bitrate}}{Total_{Bitrate}} = \mathrm{b}_1 \cdot QP^2 + \mathrm{b}_2 \cdot QP + \mathrm{b}_3. \tag{5.10}$$

- A third-order polynomial regression relationship is:

$$\frac{View_{Bitrate}}{Total_{Bitrate}} = c_1 \cdot QP^3 + c_2 \cdot QP^2 + c_3 \cdot QP + c_4 \, . \qquad (5.11)$$

To determine the accuracy of the proposed models, a Sum of Squared Error (SSE) [Jain_89] can be used. SSE is a measure to calculate the accuracy by adding the squared error and is as follows: $SSE = \sum_{I=1}^{n}(Y_P - Y_A)$, where n represents the number of samples, Y_A denotes the actual value, and Y_P indicates the predicted value. The lower the SSE the more accurate the prediction.

Table 5.10: Parameters of polynomial regression models (Eqs. 5.9, 5.10, and 5.11) for optimum (*QP*, *QD*) pairs for HEVC coding.

Sequences	First-order regression			Second-order regression				Third-order regression				
	a_1	a_2	SSE	b_1	b_2	b_3	SSE	c_1	c_2	c_3	c_4	SSE
Ballet	0.001	0.28	0.55	0.001	-0.08	1.46	0.16	-0.0001	0.008	-0.29	3.52	0.03
BBB.Flowers	-0.004	0.66	0.31	0.001	-0.06	1.50	0.04	-0.0001	0.002	-0.10	1.87	0.03
Poznan_CarPark	0.012	0.18	0.77	-0.001	0.01	0.16	0.76	-0.0001	0.014	-0.44	4.65	0.16

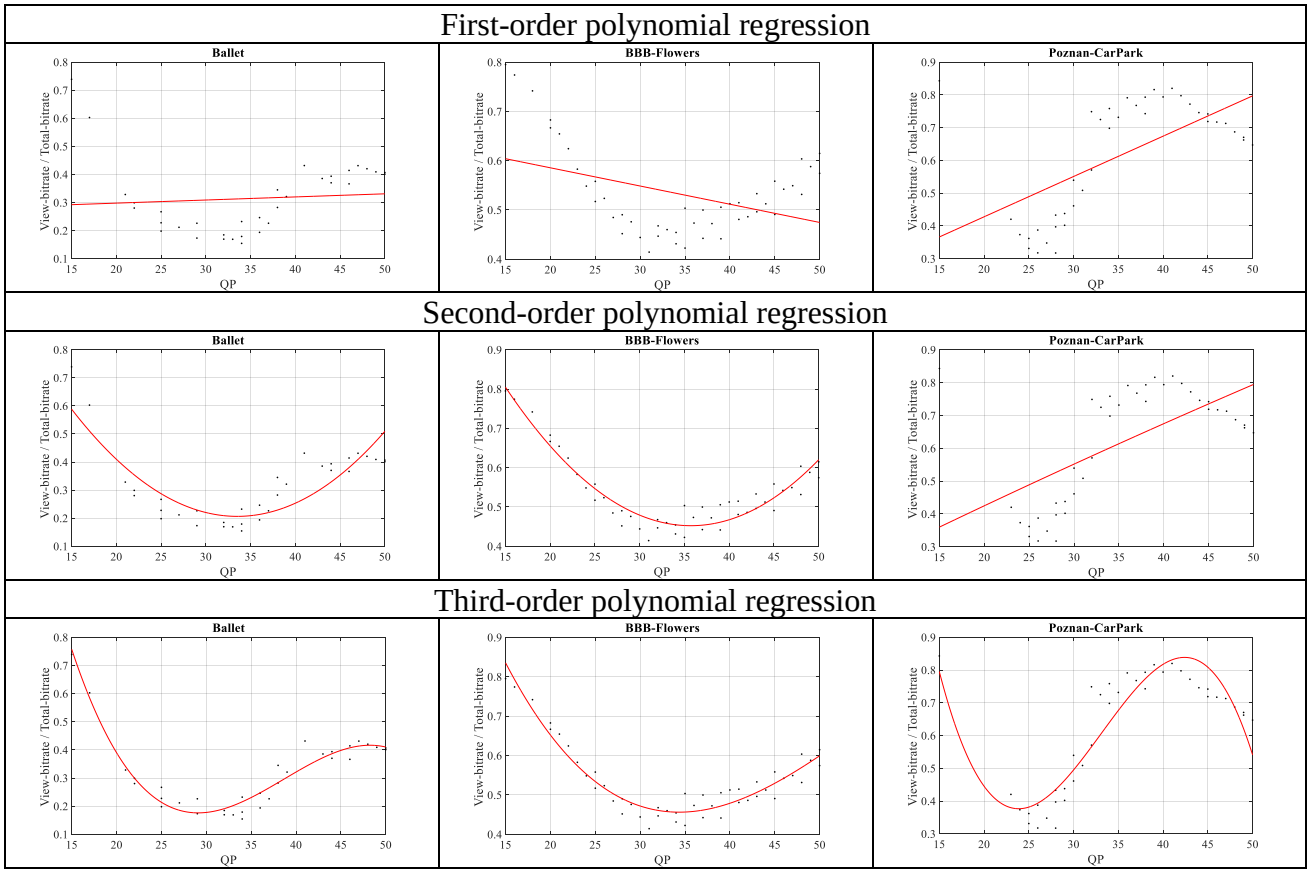


Fig. 5.9. The approximate relationship between $\frac{View_{Bitrate}}{Total_{Bitrate}}$ and *QP* for the optimum pairs for *Ballet*, *BBB.Flowers*, and *Poznan_CarPark* sequences with the use of polynomial regression.

In Table 5.10, it has been observed that the parameters of Eq. 5.11 for the stereoscopic sequences have approximately similar values, which means the general model ($\frac{View_{Bitrate}}{Total_{Bitrate}} = f(QP)$) for stereoscopic sequences can be obtained, while the parameters of Eqs. 5.9 and 5.10 have different values (i.e., a general model cannot be obtained). In Fig. 5.9, it has been noted that the curves of Eq 5.11 have the same shape for the stereoscopic sequences, while the curves of Eqs. 5.9 and 5.10 have different shapes for each stereoscopic sequence. Based on Table 5.10, the error values (SSE) of Eq. 5.11 are the lowest compared to the error values of Eqs. 5.8 and 5.9. Due to the above reasons, it is suitable to use third-order polynomial regression to derive the $\frac{View_{Bitrate}}{Total_{Bitrate}}$ model based on *QP* for HEVC coding.

Parameters c_1, c_2, c_3, and c_4 are calculated with the utilization of least-squares fitting to the optimum $\frac{View_{Bitrate}}{Total_{Bitrate}}$ - *QP* pairs generated by the proposed algorithm, as shown in Figs. I.1 - I.6 in Appendix I. In Table 5.11, the obtained results are gathered. The average model is derived from the optimum $\frac{View_{Bitrate}}{Total_{Bitrate}}$ - *QP* pairs for all training sequences using the least-squares fitting (shown in Fig. I.7 in Appendix I). Thus, the average values of the parameters of the HEVC codec model are used.

Table 5.11: Parameters c_1, c_2, c_3, and c_4 for polynomial regression model approximation (Eq. 5.11) of the optimum view_Bitrate-*QP* curve algorithm.

Sequence	c_1	c_2	c_3	c_4
Ballet	-0.0001	0.0079	-0.2878	3.5213
Breakdancers	-0.0001	0.0101	-0.3194	3.6657
BBB.Butterfly	-0.0001	0.0052	-0.1837	2.6206
BBB.Flowers	-0.0001	0.0020	-0.0967	1.8678
Kermit	-0.0001	0.0039	-0.1348	2.2485
Poznan_CarPark	-0.0001	0.0144	-0.4410	4.6515
Average	-0.0001	0.0057	-0.2055	2.8094

Through the experiments with stereoscopic sequences, as shown in Appendix I, it is noticed that the average bitrate ratio for views is about 60%: for *Breakdancers*, *BBB.Butterfly*, *BBB.Flowers*, and *Poznan_CarPark* it is about 55%, 67%, 53%, and 61%, respectively. But for some sequences, like *Ballet*, the average bitrate ratio for views is different due to many details required in depth maps to produce excellent quality of the synthesized view, where the average bitrate ratio for views in *Ballet* is about 35%. Based on the above results, it has been observed that the bitrate ratio for views and depth maps varies depending on the content in the

views and depth maps, which means that the bitrate ratio for videos and depth maps is not constant.

5.8 Conclusions

In this chapter, the optimum (*QP*, *QD*) pairs are found that form the best rate-distortion curve. It is noticed that an increase in bitrate does not always lead to an increase in the quality of the synthesized virtual view.

Based on the optimum (*QP*, *QD*) pairs, the specific models for each codec (Eq. 5.2 with average parameter values shown in Table 5.2) and the global proposed model for codecs (Eq. 5.2 with parameter values shown in Table 5.3) have been derived to estimate the quantization parameter (*QD*) for depth maps based on the quantization parameter (*QP*) for video components in data compression of stereoscopic video plus depth. The presented models significantly improve the control of the compression of stereoscopic video plus depth. The introduced models ensure almost-perfect bitrate distribution between video and depth at any required bitrate. Based on the test sequences, the bitrate reduction is about 8-35%, drawing from a comparison between the proposed models and the reference approach. The results remain the same regardless of the compression technology used (HEVC, VVC, MV-HEVC, 3D-HEVC). Also, the specific and global proposed models have been compared to the models presented in previous studies. The results prove that the specific and global proposed models outperform the models presented in previous studies of bitrate allocation. In other words, the models proposed in this chapter allow better bitrate division between video and depth data from the models presented in previous studies. In comparison between the proposed models in this chapter and the models presented in previous studies, the proposed models led to 17-62 % bitrate reduction across all test sequences.

Moreover, a model has been presented to find the relationship between view bitrate/total bitrate and the quantization parameter for video (*QP*) to determine the suitable bitrate ratio for views and depth maps in the stereoscopic sequences. Thus, it is possible to know the bitrate ratio for views or depth maps in the total bitrate of the stereoscopic sequence depending on *QP*. Also, the results prove that the bitrate ratio for views and depth maps in the total bitrate of stereoscopic video varies depending on the content in the views and depth maps. This means that the bitrate ratio for videos and depth maps is variable.

Chapter Six

Influence of Depth Map Fidelity on Virtual View Quality

6.1 Motivation and Aim of the Work

Bitrate allocation between view and depth data is one of the essential problems considered in multiview systems. During the work to derive the bitrate allocation models in the previous chapter, it has been noted that not all MVD sequences behave in the same way, as shown in Appendix F. Some of the test sequences present unexpected and surprising behavior. Specifically, decreasing the bit allocation for the depth component leads to an increase in the quality of the virtual views in some bitrate range under the video bitrate constancy condition. This unexpected behavior has been noted only for some sequences, but not all test sequences.

The previous studies on the influence of depth map quality on virtual view quality have not mentioned this phenomenon, which means that these studies did not achieve a comprehensive study. Therefore, the goal of this chapter is to present a comprehensive study on this subject.

6.2 Study of the Influence of Depth Map Fidelity on Virtual View Quality

To study the influence of depth map fidelity on virtual view quality, noise was added to depth maps. Then, the quality of the virtual view produced from the unmodified views and depth maps (depth maps without noise) is compared to the quality of the virtual view obtained from the unmodified views and modified depth maps (depth maps with noise).

In the experiments, adding noise N is implemented as the addition of random values with a probability of uniform distribution between ($-N$ and $+N$) to each of the values stored in depth maps. N is an integer number (such as 1, 2, 3, etc.).

6.3 Methodology of Experiments

Fig. 6.1 presents the flow chart used to study and assess the effect of noise added to depth maps on the quality of virtual views.

Initially, two views with two related depth maps are independently encoded and decoded using HEVC coding [HEVC]. The next step is to create a virtual view depending on decoded views with associated depth maps. Then, this synthesized view is compared with the view obtained by the real camera in the same position in 3D space as the virtual one.

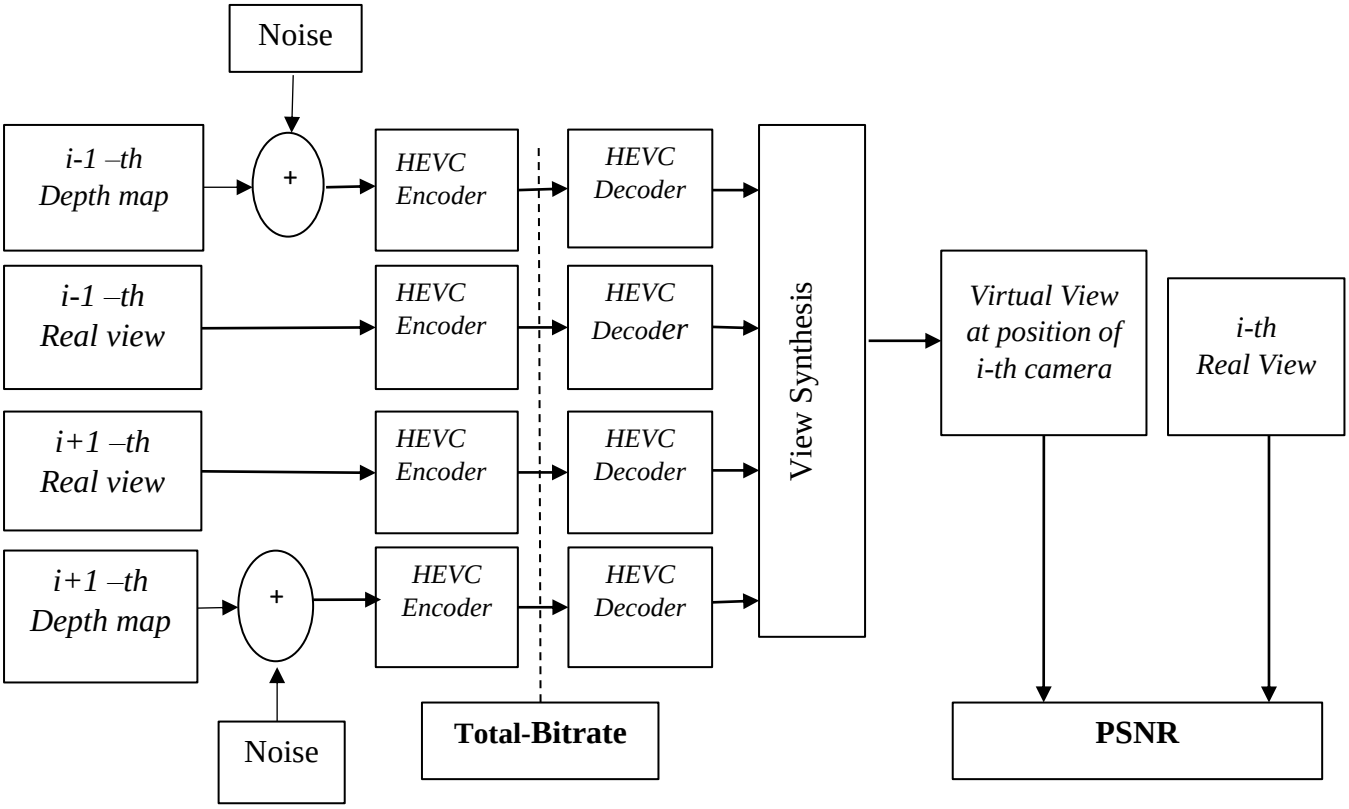


Fig. 6.1. The scenario of adding noise to depth maps in the flow chart of the performed experiments [Alob_18a].

The experiments were performed on many test multiview video sequences with depth maps (i.e., *Ballet, Breakdancers*, *BBB.Butterfly*, and *BBB.Flowers*), as shown in Table 3.2. To compress views and depth maps, the reference software of HEVC, namely HM v.16.18 [HEVC], has been used (mentioned in Section 3.3). For view synthesis, the reference model software VSRS v.3.5 [Stan_13b] has been used, as mentioned in Section 3.4. For the sake of simplicity, it has been assumed that the quantization parameter *QP* is constant for all views, and the quantization parameter *QD* is constant for all depth maps.

6.4 Results of the Experiments

To find the optimum *QP*-*QD* settings, all (*QP, QD*) pairs were tested (*QP* and *QD* values both from 15 to 50). It resulted in many encodings and virtual views generated, for which the data have been collected. Then, the optimum (*QP*, *QD*) pairs were calculated based on the proposed method in [Alob_18b, Alob_19, Alob_20]. Figs. 6.2 and 6.3 show the optimum (*QP*, *QD*) pairs for *Breakdancers* and *BBB.Butterfly* sequences. Optimum (*QP, QD*) pairs belong to the peak envelope over a cloud of PSNR-bitrate points that form the best rate-distortion curve. For the *Breakdancers* sequence, when applying stronger compression (selecting higher quantization parameter values, i.e., decreasing bitrate), the quality of the synthesized view increases. After reaching a maximum point (the highest point in the PSNR), it starts decreasing. This increasing part of the R-D curve is unexpected and surprising because the result of the

decreased number of bits required to represent the MVD sequence leads to a better quality of the virtual view. Also, it is noticed in Appendix F (Fig. F.1) that the *Ballet* sequence behavior is similar to the *Breakdancers* sequence behavior. However, this observation does not apply to the *BBB.Butterfly* and *BBB.Flowers* sequences.

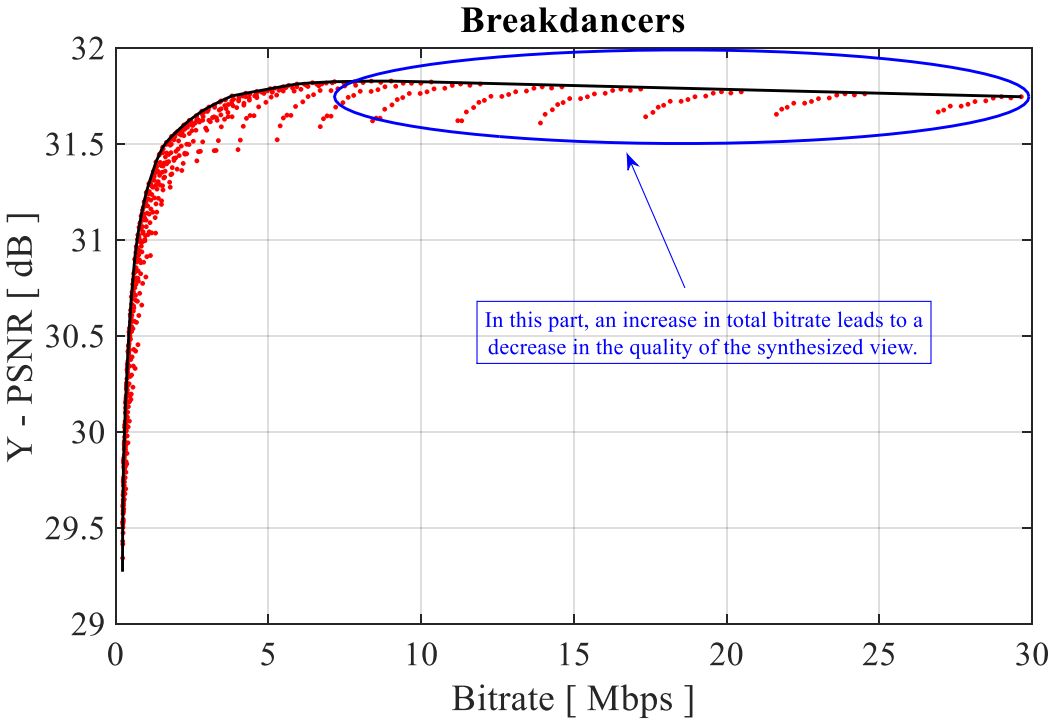


Fig. 6.2. The best R-D curves for the unmodified depth maps for the optimum (*QP*, *QD*) pairs for the *Breakdancers* sequence.

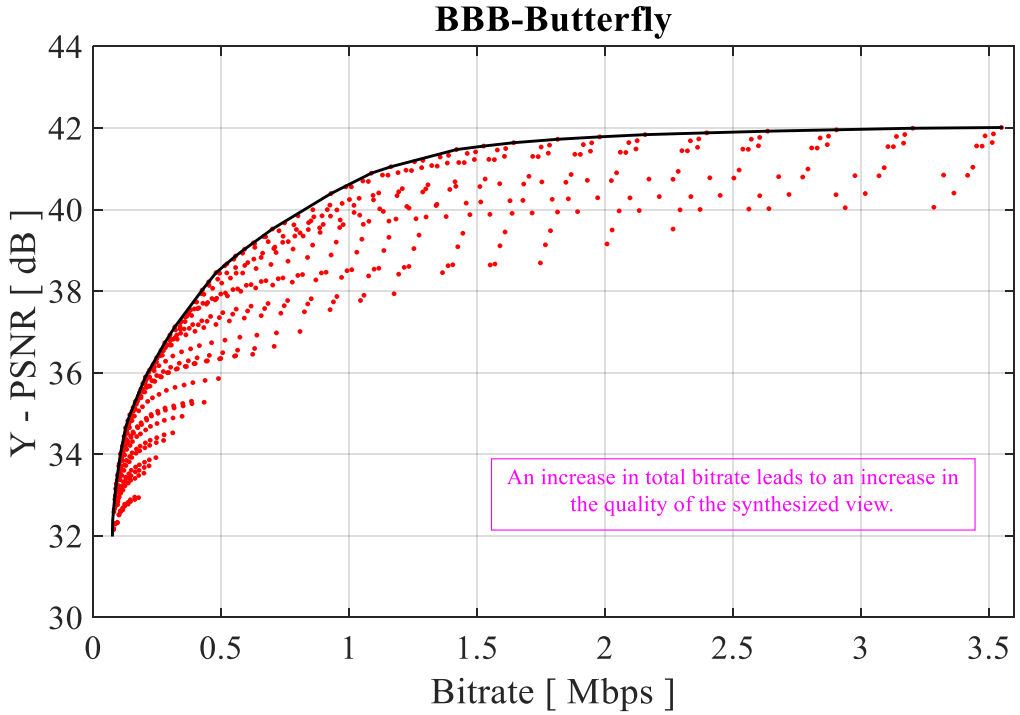


Fig. 6.3. The best R-D curve for the unmodified depth maps for the optimum (*QP*, *QD*) pairs for the *BBB.Butterfly* sequence.

The experiments will be repeated to study the effect of depth map quality on the quality of the virtual synthesized view, but this time by adding noise to depth maps (this additive noise simulates some effects of inaccurate depth estimation). Many noise values added to depth maps are tried. Then, the optimum (*QP*, *QD*) pairs are obtained from data gathered through experiments for every *N* value. The noise value *N* is the value randomly added to or subtracted from the depth samples.

The relationship between the quality of the virtual views and the total bitrate for the considered test sequences with various quality of depth maps is shown in Figs. 6.4, 6.5, 6.6, and 6.7. The lines (blue, red, green, and magenta) denote the optimum (*QP*, *QD*) pairs (optimum R-D curves). The blue line represents the quality of a virtual view synthesized from video and depth maps without noise (unmodified depth maps), whereas red, green, and magenta lines express the quality of virtual views produced from video and depth maps with different noise (*N*) added. Additionally, the black line represents the quality of the virtual view synthesized from the uncompressed and unmodified video and depth maps. In contrast, the (*QP*, *QD*) pair with the highest quality of the virtual view for a given depth map quality is represented in black squares. Despite the amount of noise added, the highest quality of the virtual view is achieved for the same *QP* value. Nevertheless, the more noise is added, the *QD* value should be higher.

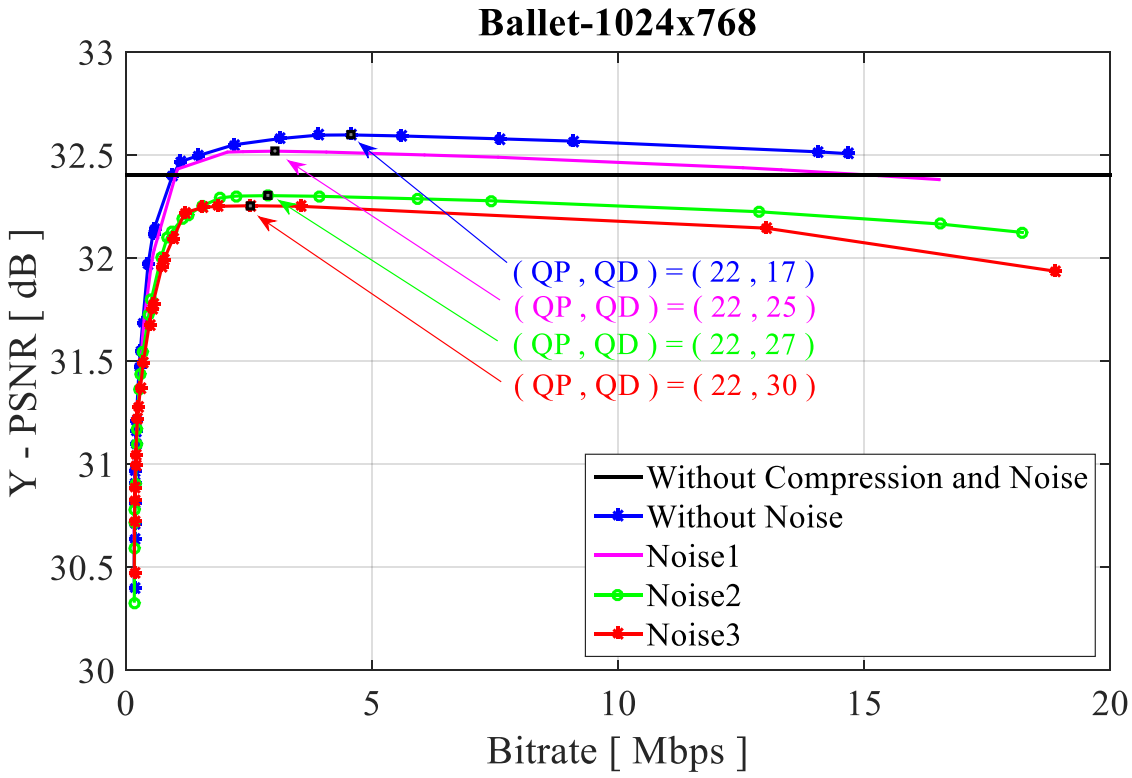


Fig. 6.4. Optimum R-D curves for different quality of depth maps for the *Ballet* sequence.

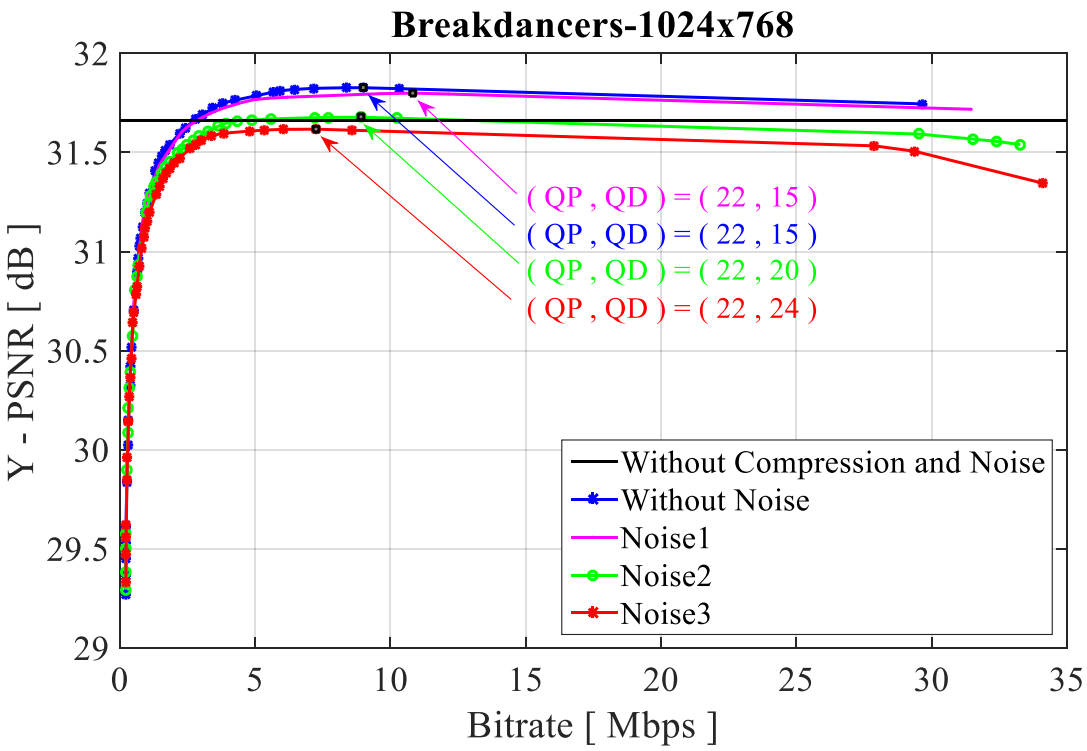


Fig. 6.5. Optimum R-D curves for different quality of depth maps for the *Breakdancers* sequence.

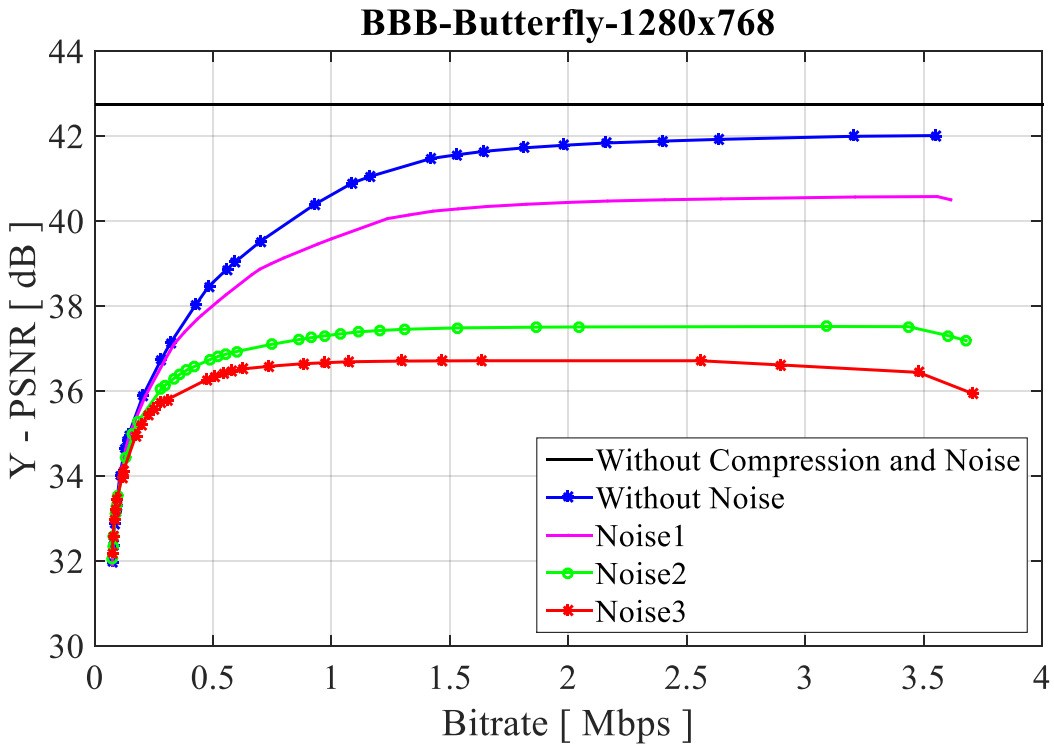


Fig. 6.6. Optimum R-D curves for different quality of depth maps for the *BBB.Butterfly* sequence.

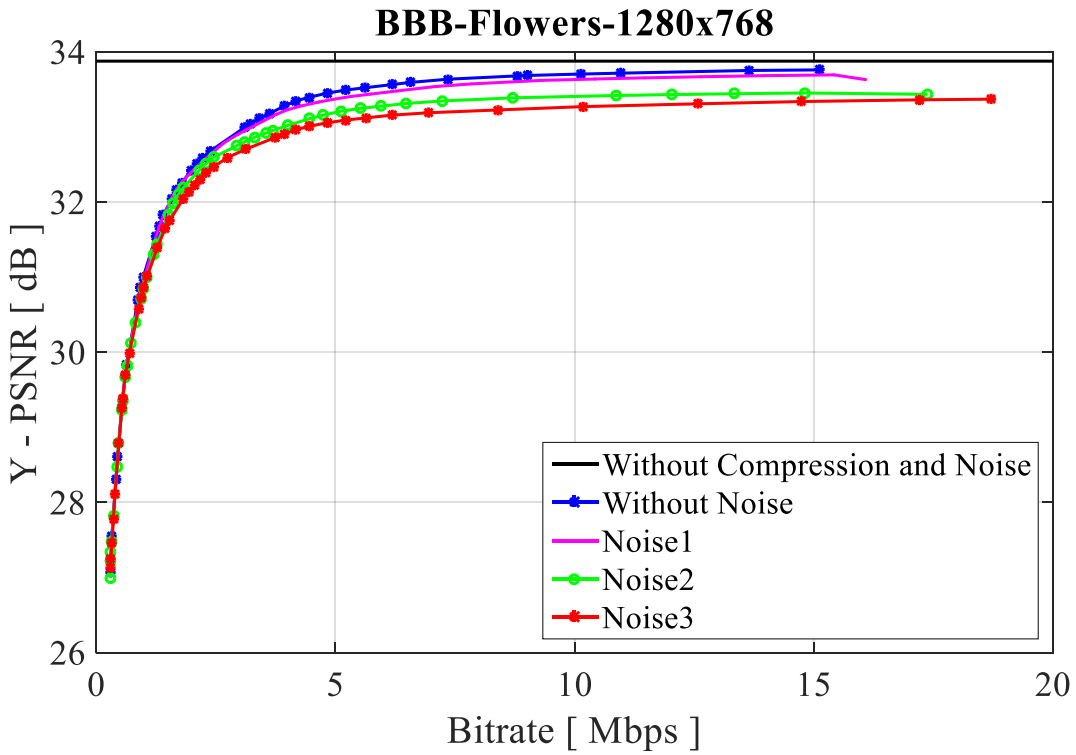
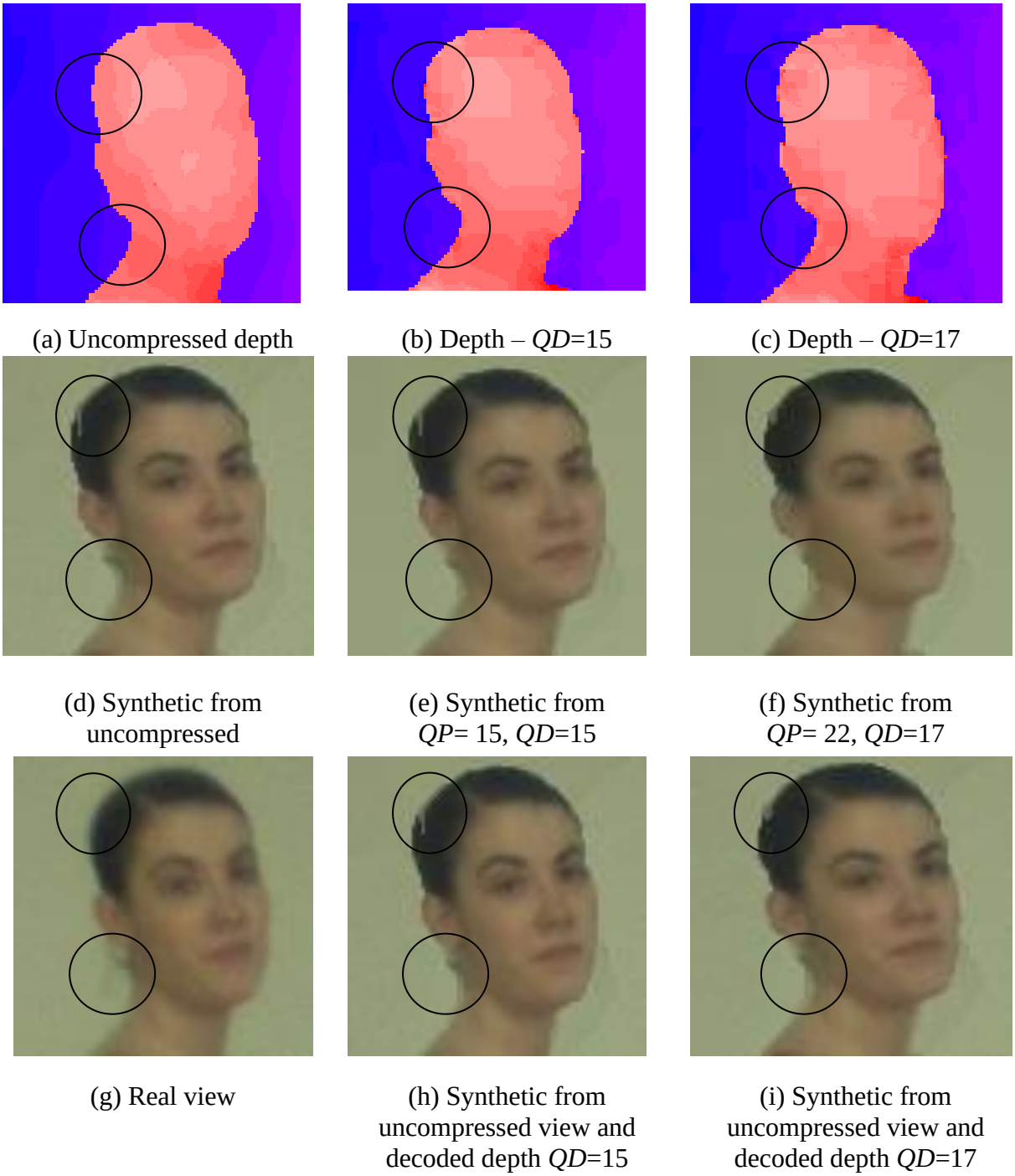


Fig. 6.7. Optimum R-D curves for different quality of depth maps for the *BBB.Flowers* sequence.

As it is known, depth maps are usually produced by depth estimation algorithms (such as DERS (depth estimation reference software) [Stan_13c]), which means that depth maps are not ideal. Therefore, it has been noticed in some cases that the quality of the virtual view produced from decoded video and depth maps is better than the quality of the virtual view produced from the uncompressed video and depth maps because decoded depth maps are smoother than uncompressed depth maps, as shown in Figs. 6.4 and 6.5. This can be explained by a comparison between decoded depth maps and uncompressed depth maps, as shown in Figs 6.8 and 6.9.

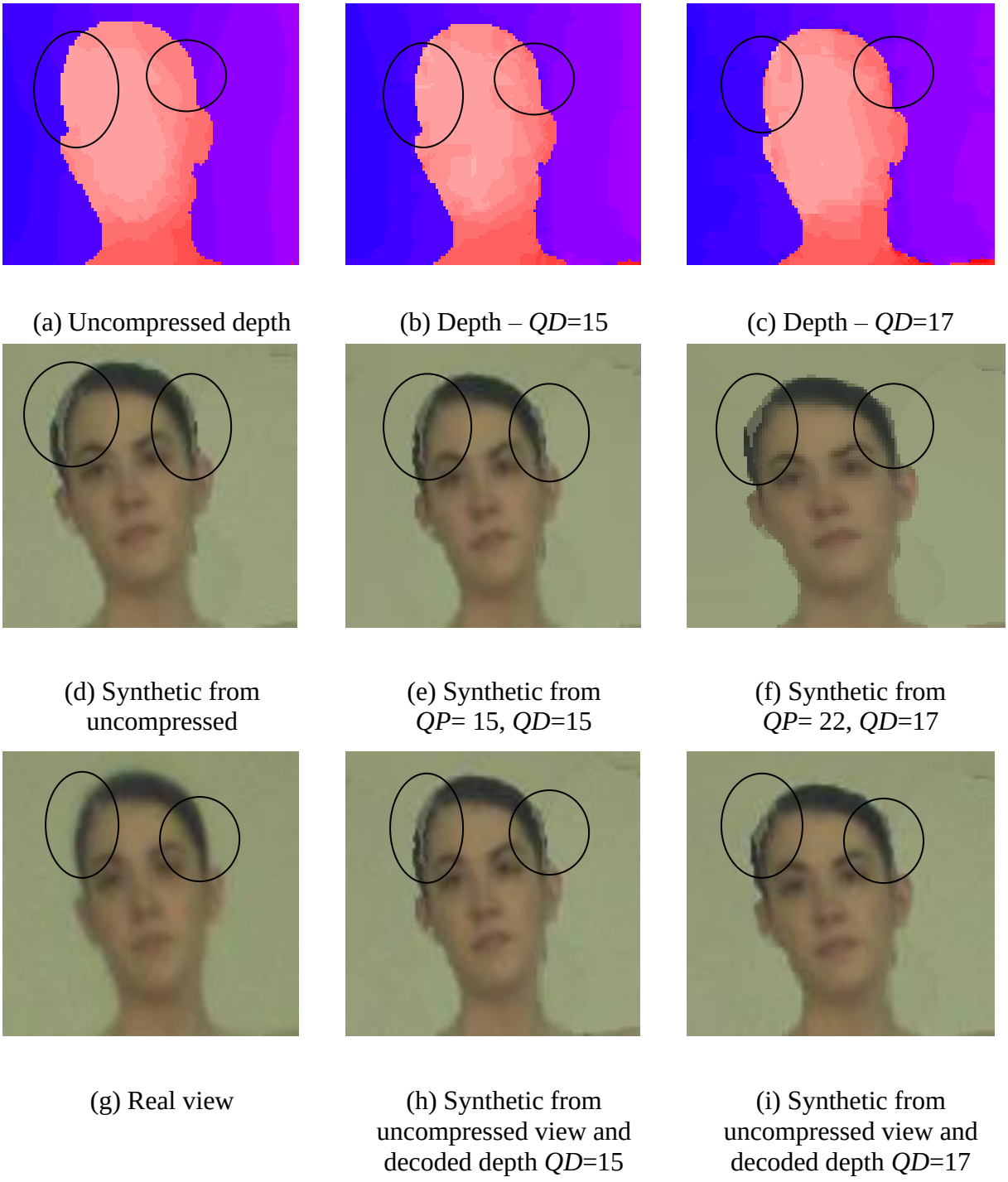


(a) Uncompressed depth (b) Depth – *QD*=15 (c) Depth – *QD*=17

(d) Synthetic from uncompressed (e) Synthetic from *QP*= 15, *QD*=15 (f) Synthetic from *QP*= 22, *QD*=17

(g) Real view (h) Synthetic from uncompressed view and decoded depth *QD*=15 (i) Synthetic from uncompressed view and decoded depth *QD*=17

Fig. 6.8. Comparison of the uncompressed depth map and the decoded depth map, and their effect on virtual views for the HEVC codec for the Ballet sequence. The circles mention some artifacts that deteriorate the quality of the synthetic (virtual) pictures.



(a) Uncompressed depth | (b) Depth – *QD*=15 | (c) Depth – *QD*=17

(d) Synthetic from uncompressed | (e) Synthetic from *QP*= 15, *QD*=15 | (f) Synthetic from *QP*= 22, *QD*=17

(g) Real view | (h) Synthetic from uncompressed view and decoded depth *QD*=15 | (i) Synthetic from uncompressed view and decoded depth *QD*=17

Fig. 6.9. Comparison of the uncompressed depth map and the decoded depth map, and their effect on virtual views for the HEVC codec for the Ballet sequence. The circles indicate some artifacts that deteriorate the quality of the synthetic (virtual) pictures.

In Figs.6.8 and 6.9, it can be noticed in the case of weaker compression (such as $QD < 25$) that the compressed depth map is smoother than the uncompressed depth map because the compression process will remove some errors in the depth maps while maintaining the content in the depth maps. Additionally, it can be observed in some cases that the depth map compressed with a higher quantization parameter value (*QD*=17) is smoother than the depth map compressed with a lower quantization parameter value (*QD*=15), and this explains the reason behind the phenomenon (a decrease in bitrate leads to an increase in the quality of the virtual views) shown in Fig.6.2.

In order to measure the influence of the noise added to depth maps on the virtual view quality, the bitrate reduction has been calculated by the Bjøntegaard rates (shown in Section 3.6) between the best rate-distortion curve obtained from unmodified views and unmodified depth maps and the best rate-distortion curve obtained from unmodified views and modified depth maps, as shown in Table 6.1.

Table 6.1: The bitrate reduction calculated by the Bjøntegaard rates between the best rate-distortion curve obtained from views and depth maps without noise and the best rate-distortion curve obtained from unmodified views and modified depth maps.

Sequence	Unmodified depth maps vs depth maps with noise 1		Unmodified depth maps vs depth maps with noise 2		Unmodified depth maps vs depth maps with noise 3	
	ΔPSNR [dB]	ΔBitrate [%]	ΔPSNR [dB]	ΔBitrate [%]	ΔPSNR [dB]	ΔBitrate [%]
Ballet	0.06	-7.09	0.23	-19.47	0.25	-20.71
Breakdancers	0.01	-0.63	0.09	-9.22	0.13	-14.23
BBB.Butterfly	0.64	-12.60	2.09	-22.44	2.55	-31.39
BBB.Flowers	0.04	-2.39	0.17	-6.83	0.25	-10.16

Based on Table 6.1, it has been observed that the quality of the virtual views obtained from depth maps with noise is lower than the quality of the virtual views obtained from depth maps without noise.

Also, the relationship between depth map quality and the quality of virtual views (Y-PSNR) for several quantities of noise added is illustrated in Figs. 6.10 and 6.11. The quality of depth maps (Depth PSNR) is computed as the average depth map quality related to (*i+1–th*) and (*i-1–th*) views.

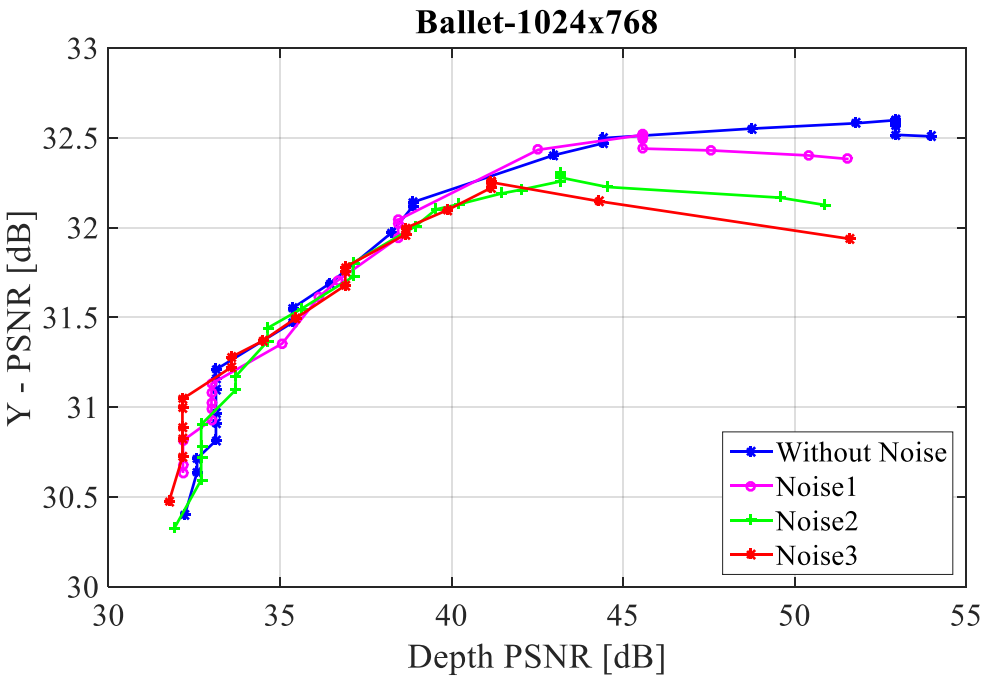


Fig. 6.10. Effect of the fidelity of depth maps on virtual view quality for the *Ballet* sequence.

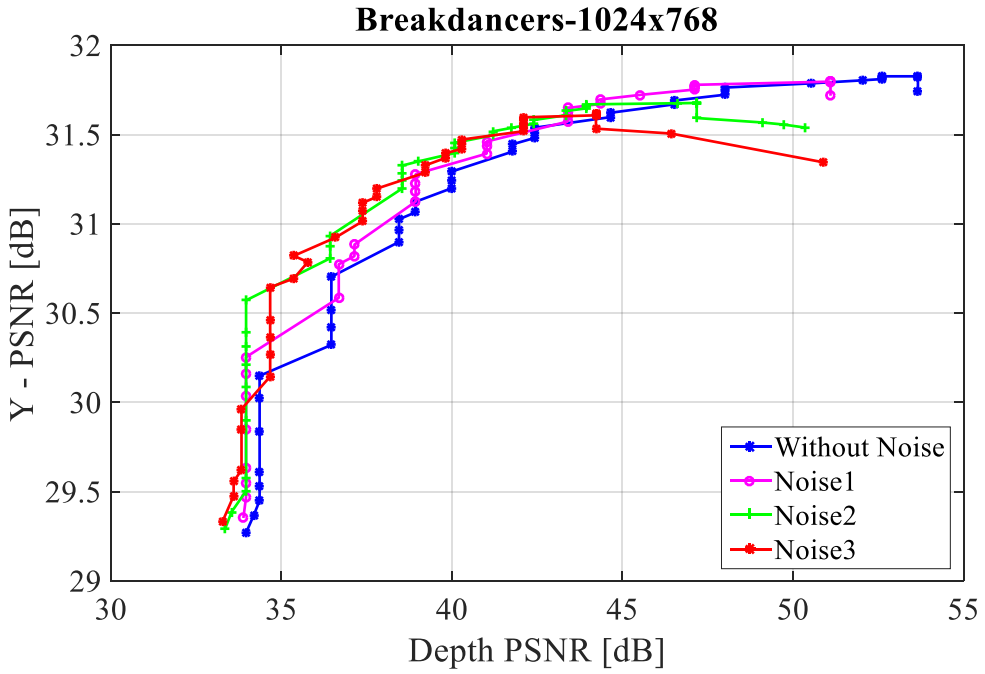


Fig. 6.11. Effect of the fidelity of depth maps on virtual view quality for the *Breakdancers* sequence.

The effect of added noise to depth maps on the virtual view quality is observed in Figs. (6.4, 6.5, 6.10, and 6.11), however, only for high bitrates, when compression is weak. The effect of added noise on virtual view quality becomes negligible if stronger compression is applied. The explanation is that quantization noise produced by compression becomes more robust than the noise added to depth maps before compression at some stage. Generally, the quantization process deletes high-frequency components from video; thus, it will remove noise because noise is a high-frequency signal. In the case of adding stronger noise to the depth maps,

higher quantization is required to delete this added noise during the compression of MVD sequences.

Figs. 6.12, 6.13, 6.14, and 6.15 show the optimum (*QP*, *QD*) pairs for the considered test sequences for different quality of the depth maps. Also, for the values of the quantization parameters (*QP*) under 30, the higher the noise is added, the higher *QD* should be used to obtain the highest possible quality of virtual views.

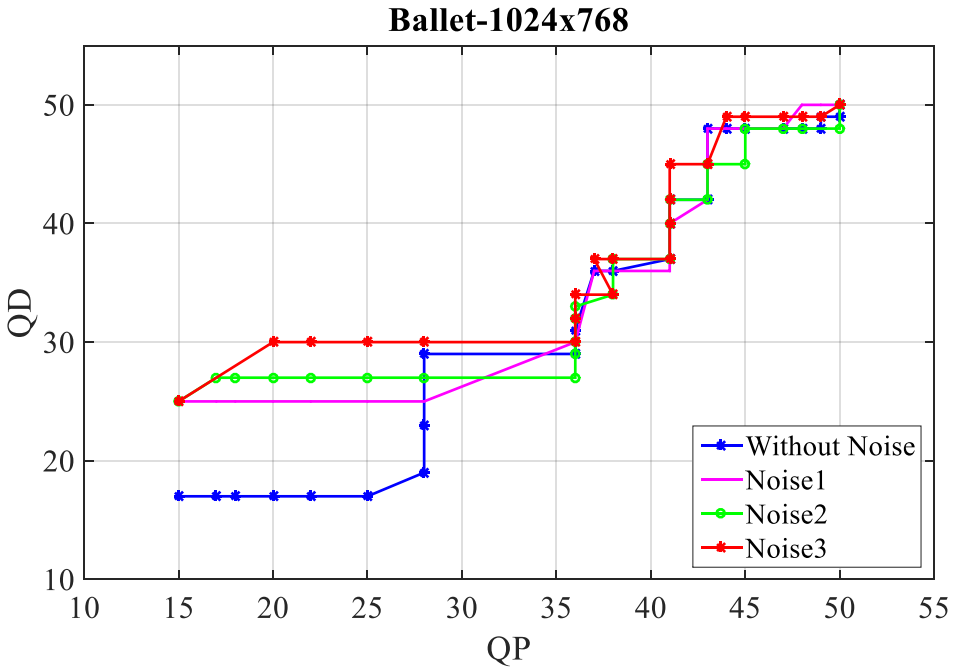


Fig. 6.12. The optimum (*QP*, *QD*) pairs for the *Ballet* sequence for different fidelity of the depth maps.

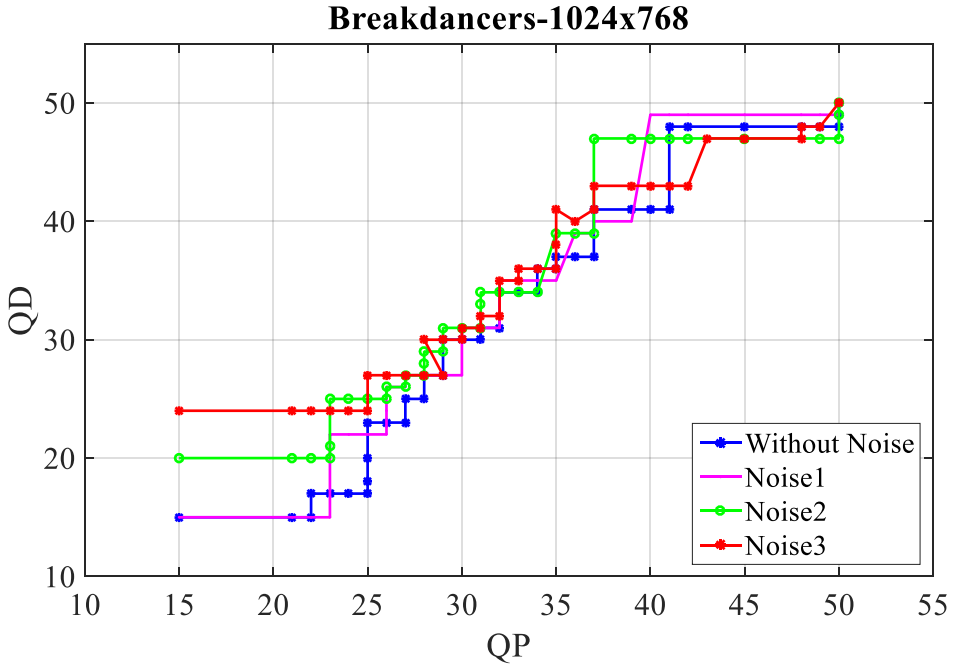


Fig. 6.13. The optimum (*QP*, *QD*) pairs for the *Breakdancers* sequence for different fidelity of the depth maps.

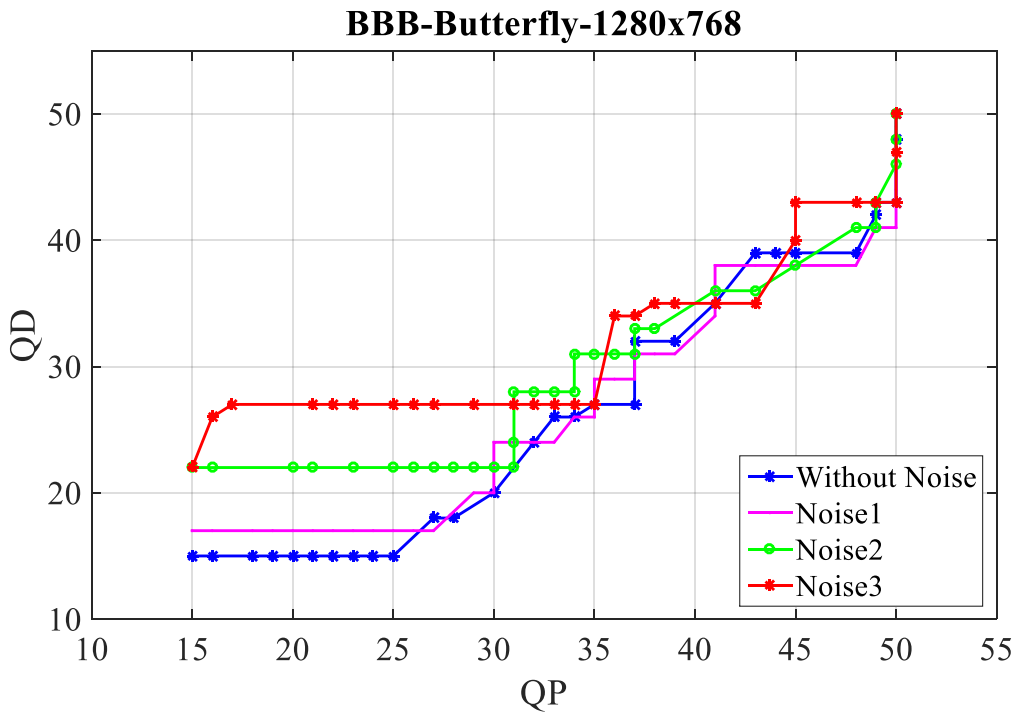


Fig. 6.14. The optimum (*QP*, *QD*) pairs for the *BBB.Butterfly* sequence for different fidelity of the depth maps.

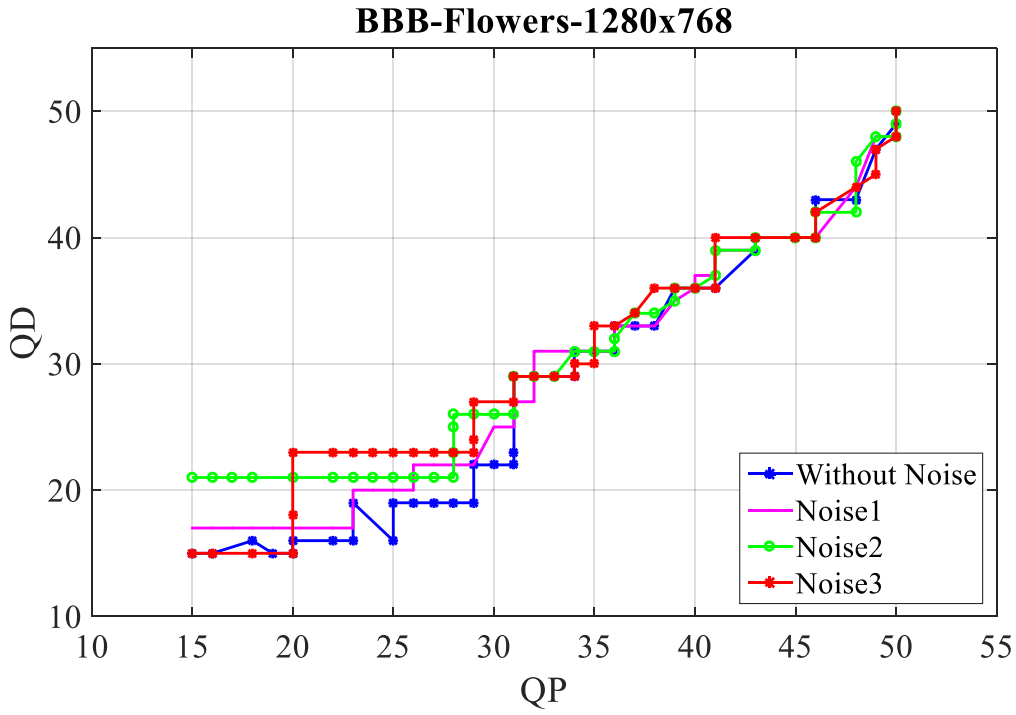


Fig. 6.15. The optimum (*QP*, *QD*) pairs for the *BBB.Flowers* sequence for different fidelity of the depth maps.

6.5 Conclusions

The impact of the quality of the depth maps on virtual view quality is investigated in the chapter. The influence of depth map quality on the quality of the synthesized views was tested by adding noise to depth maps (shown in Section 6.2) and then comparing the best R-D curves obtained before and after adding noise to depth maps. Experiments in this chapter have proved that depth maps' errors affect the virtual view quality. Additionally, experimental results demonstrate that the best R-D curves obtained from views and depth maps without noise can result in 1-31% bitrate reduction compared with the best R-D curves obtained from the views and the depth maps with noise *N* (noise1, noise2, and noise3), which means that a decrease in depth map quality by increasing the amount of noise added to depth maps leads to a decrease of virtual view quality.

The effect of noise added to depth maps become more prominent on the virtual view quality in some cases of weaker compression (higher bitrate, i.e., lower quantization parameters values), where an increase in the depth map quality results in a decrease of virtual view quality because the quantization noise (error) introduced by compression is lower than the added noise. For higher compression cases, the quantization noise is higher than the added noise, which means the difference between the qualities of virtual views achieved for different quantities of added noise is negligible.

For somewhat high bitrates, the more noise is added to the depth maps, the higher *QD* should be used to obtain the highest possible quality of virtual views.

Additionally, depth maps are usually obtained by depth estimation algorithms. It means that depth maps are not perfect (real cameras do not record depth maps), thus, smoother depth maps are required to obtain a higher quality of synthesized views. For computer-generated sequences, it has not been observed that an increase in the bitrate of depth maps leads to a decrease in the quality of synthesized views, but this phenomenon only occurred after adding noise to the depth maps because the modified depth maps became less accurate (less smooth) than the unmodified depth maps. Thus, this phenomenon occurs because of the inaccuracy of the depth data acquired or estimated.

Chapter Seven

Encoder Model for Stereoscopic Video plus Depth

7.1 Objective of the Work

The R-Q model is one of the efficient models to control the bitrate for stereoscopic video plus depth (two videos and two depth maps) for MVD coding. Many R-Q models were proposed to describe the relationship between the quantization step size and the bitrate or frame size (the number of bits per frame), as described in Subsection 2.3.1. The model presented in [Graj_10] (Eq. 2.4) is one of the proposed R-Q models for the two-dimensional video. As mentioned in Subsection 2.3.1, the authors in [Graj_10] claimed that the model proposed in [Graj_10] outperforms the quadratic R-Q model (Eq. 2.2). Consequently, the goal of the chapter is to use the model presented in [Graj_10] to control the bitrate and frame size for MVD sequences for many compression techniques, depending on the bit allocation models obtained in Subsection 5.4.2.

7.2 Derivation of the R-Q Model Proposed for Bitrate Control

Generally, the R-Q model is used to control the bitrate and frame size for MVD sequences depending on the quantization step size (Q). The quantization step size used in the R-Q model is calculated based on the quantization parameter for video (QP) (shown in Eq. 2.1). Also, the bitrate of the MVD sequence (two videos plus two depth maps) is calculated by MVD coding with the bitrate allocation model ($QD = f(QP)$) obtained in Subsection 5.4.2. In this section, the used R-Q model and its simplified models are presented.

According to Fig. 3.2, the experiments have been performed to control the bitrate for MVD sequences. Reference software for HEVC, VVC, MV-HEVC, and 3D-HEVC (mentioned in Section 3.3) was used to encode and decode two views and two depth maps. Multiview sequences recommended by the Moving Picture Experts Group (MPEG) were used, as shown in Table 3.2. The data were gathered for values of QP from 25 to 50, which corresponds to practically used bitrates. This QP interval corresponds to a Q interval from about 11 to 203. The QP values are the used values for GOP. The QP offsets were set according to the MPEG common test conditions (CTC) for individual frame types, shown in Table 3.3. The GOP structure used in the experiments for all codecs is I B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3 P B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3 I.

7.3 R-Q Model Used for GOP-Level Bitrate Control

The R-Q model is used to describe the relationship between the quantization step size and the bitrate measured over the whole GOP. For matching the experimental data, the R-Q model is used as follows:

$$R(Q, \emptyset) = \frac{a}{Q^b + c}, \tag{7.1}$$

where:

R - estimated bitrate for two videos and two depth maps by the proposed model with three parameters,

$\emptyset = [a\ b\ c]$ - parameters of the model that depend on sequence content,

Q - quantization step size for video.

Using the trust-region optimization method with the number of iterations equal to 2×10^6 and with the tolerance value of 1×10^{-6} (shown in Section 3.7), the model parameters are estimated by minimizing the error between the experimental and approximated curves, and the best estimate of the model parameters means getting the smallest error between these curves. The accuracy of the approximation of the experimental data is measured as follows [Graj_10, Choi_13, Hu_13, Guo_15]:

$$Relative\ error\ (Q, \emptyset) = \frac{|R_X(Q) - R\ (Q, \emptyset)|}{R_X(Q)} \times 100\%, \tag{7.2}$$

where:

$Relative\ error\ (Q, \emptyset)$ - relative approximation error,

$R_X\ (Q)$ - bitrate measured for two videos and two depth maps,

$R\ (Q, \emptyset)$ - bitrate estimated using the proposed model.

In the experiments to estimate model parameters and errors, the number of GOP used for all codecs is three. Table 7.1 shows the values of the mean relative approximation error for the considered MVD sequences. The values of parameters *a*, *b*, and *c* shown in Table 7.1 are calculated by finding mean values (average values) of the values of the parameters *a*, *b*, and *c*, respectively, for the considered MVD sequences; more details in Appendix J (Table J.1- Table J.4). The experimental and approximate curves of the bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for different codecs are presented in Figs.7.1, 7.2, 7.3, and 7.4. In Figs.7.1, 7.2, 7.3, and 7.4, the approximate curves are calculated using the model used with parameters' values shown in Appendix J (Table J.1-Table J.4).

Table 7.1: Mean relative approximation error for the bitrate of considered MVD sequences (shown in Table 3.2).

Codec	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev.
HEVC	55503	1.01	-3.84	5.88	3.99
VVC	73175	1.10	-3.33	3.10	2.54
MV-HEVC	71950	1.16	-3.54	3.24	2.39
3D-HEVC	62006	1.12	-2.70	2.71	2.13

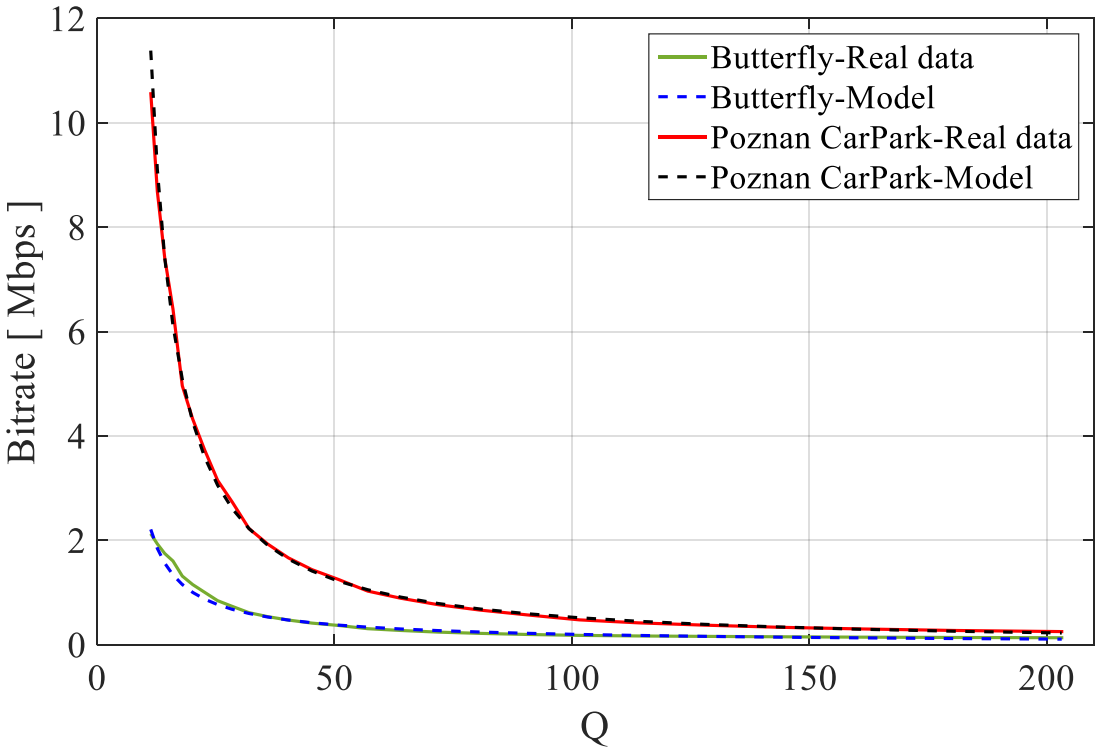


Fig. 7.1. The experimental and the approximated (curve estimated with the model (Eq.7.1)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the HEVC coding.

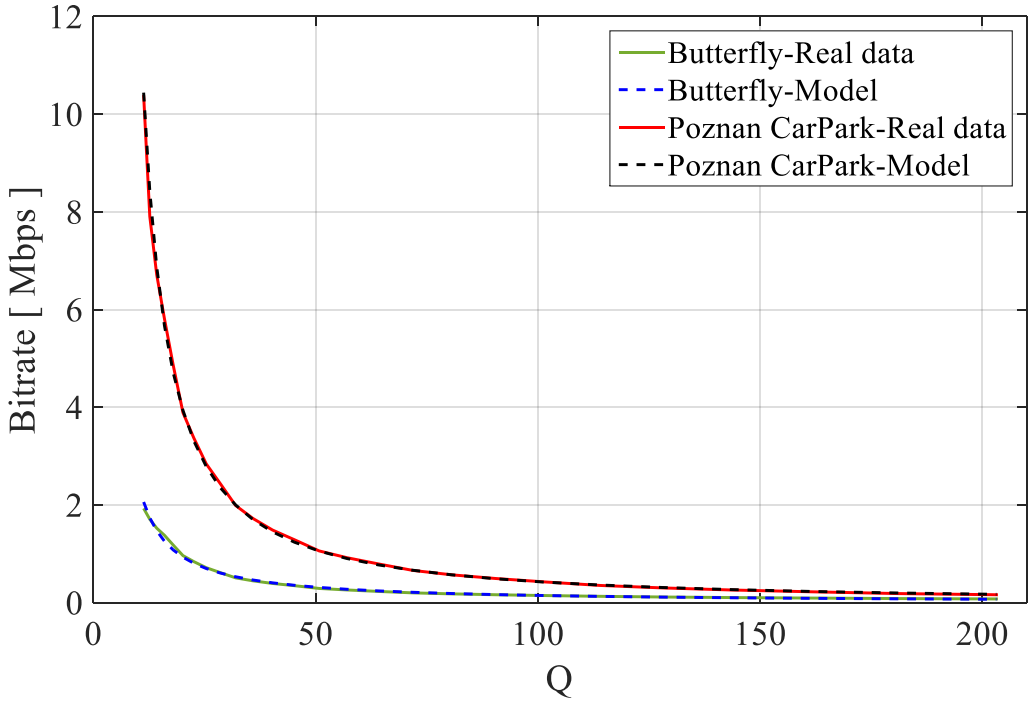


Fig. 7.2. The experimental and approximated (curve estimated with the model (Eq.7.1)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the VVC coding.

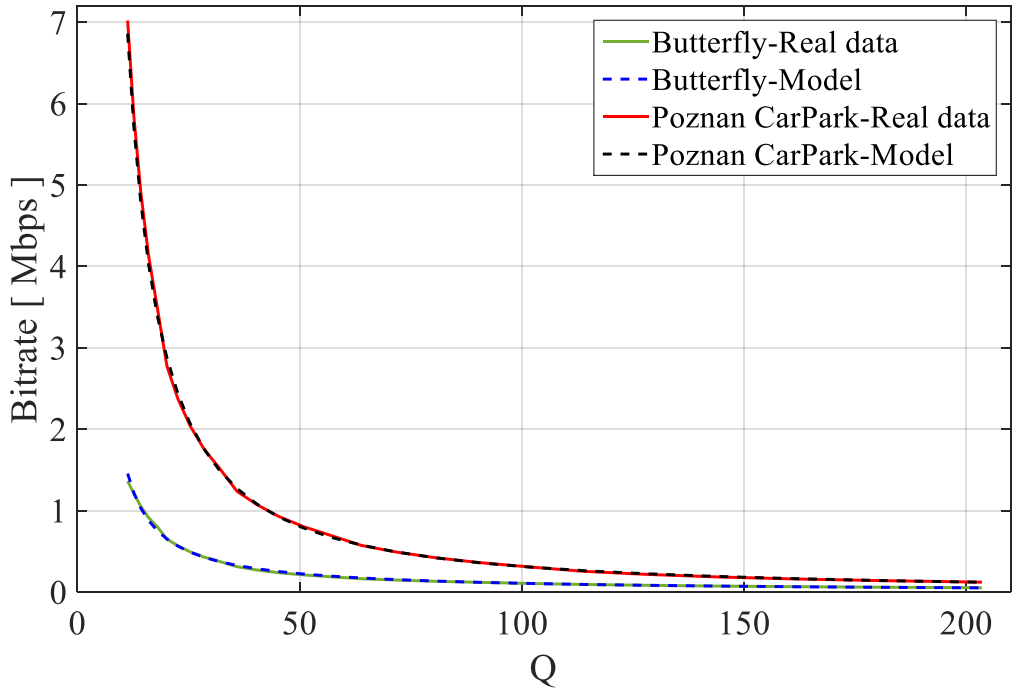


Fig. 7.3. The experimental and approximated (curve estimated with the model (Eq.7.1)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the MV-HEVC coding.

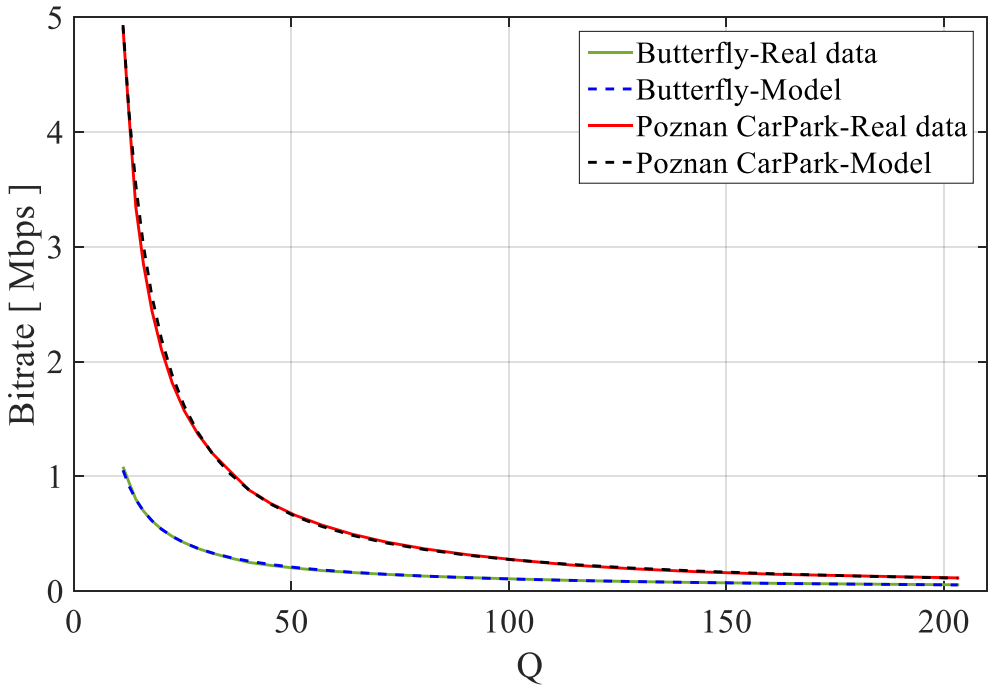


Fig. 7.4. The experimental and approximated (curve estimated with the model (Eq.7.1)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the 3D-HEVC coding.

Based on the experiments (Table 7.1, and Figs. 7.1, 7.2, 7.3, and 7.4), it has been observed that the error of the used R-Q model for the considered MVD sequences is low. Consequently, the R-Q model used can be employed to bitrate control for stereoscopic video plus depth.

7.4 Simplified Models for GOP-Level Bitrate Control

Based on the results of experiments in the previous subsection (Table 7.1), it has been noticed that the values of some of the model parameters are roughly similar for the considered MVD sequences. Therefore, the model parameters that depend on sequence content can be reduced by using the following two approaches. In the experiments to estimate model parameters and errors, the number of GOP used for all codecs is three.

7.4.1 A Model with Two Parameters

According to Table 7.1, the values of parameter *c* are about -3.5 for the considered MVD sequences for many compression techniques. Therefore, it is assumed that parameter *c* is equal to -3.5 for all MVD sequences for compression techniques. As a result, this approach uses two parameters, instead of three, that depend on the video content. The model used in this approach is the following:

$$R(Q, \emptyset) = \frac{a}{Q^{b-3.5}}, \tag{7.3}$$

where:

R - estimated bitrate for two videos and two depth maps by the proposed model with two parameters,

$\emptyset = [a\ b]$ - parameters of the model that depend on sequence content,

Q - quantization step size for video.

Based on Eq. 7.2, the model parameters are estimated by minimizing the error between the experimental and approximated curves, as mentioned in the previous subsection.

The values of selected statistics of the mean relative approximation error for the considered MVD sequences are shown in Table 7.2, with more details in Appendix K (Tables K.1-K.4). Additionally, Figs. 7.5, 7.6, 7.7, and 7.8 illustrate the experimental and approximated curves (curve estimated using the model of Eq. 7.3) of the bitrate for the *Poznan_CarPark* and *BBB.Butterfly* sequences for different codecs.

Table 7.2: Mean relative approximation error for the bitrate of considered sequences.

Codec	Relative error [%]	
	mean	std. dev.
HEVC	7.38	5.06
VVC	5.42	4.05
MV-HEVC	4.69	3.08
3D-HEVC	3.21	2.20

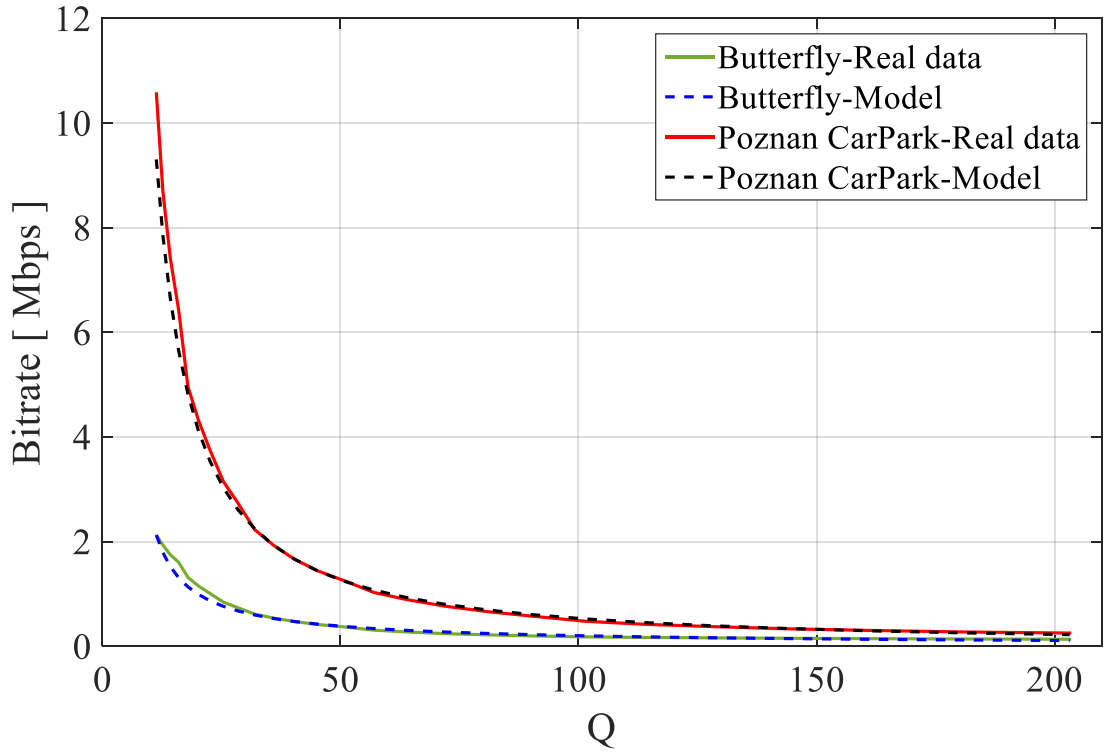


Fig. 7.5. The experimental and approximated (curve estimated by the model (Eq. 7.3)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the HEVC coding.

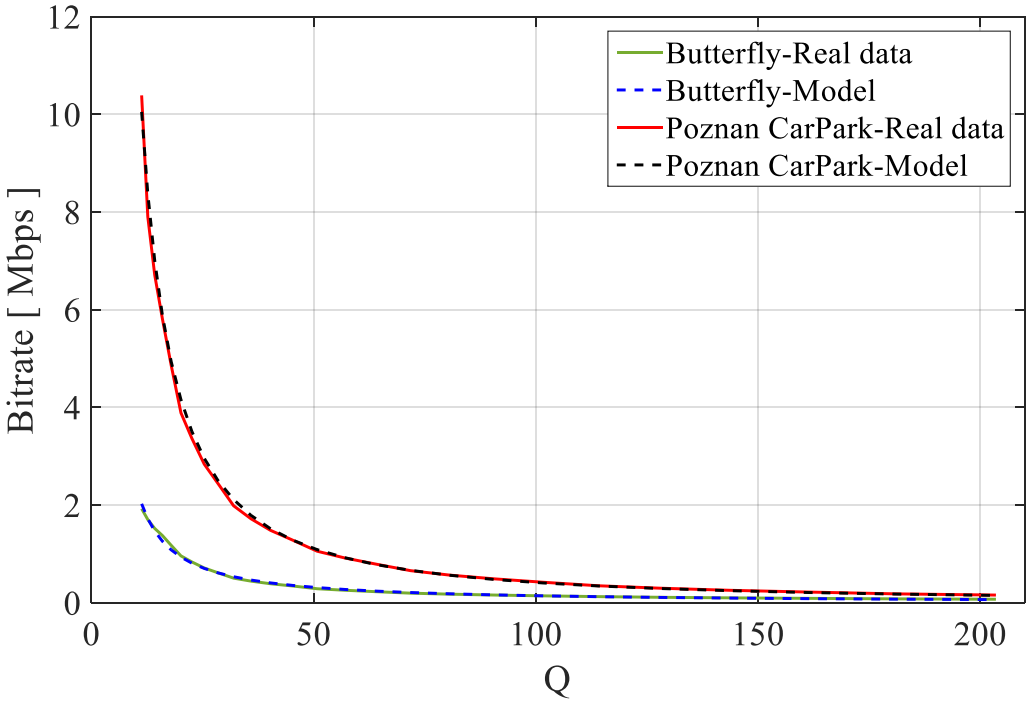


Fig. 7.6. The experimental and approximated (curve estimated by the model (Eq. 7.3)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the VVC coding.

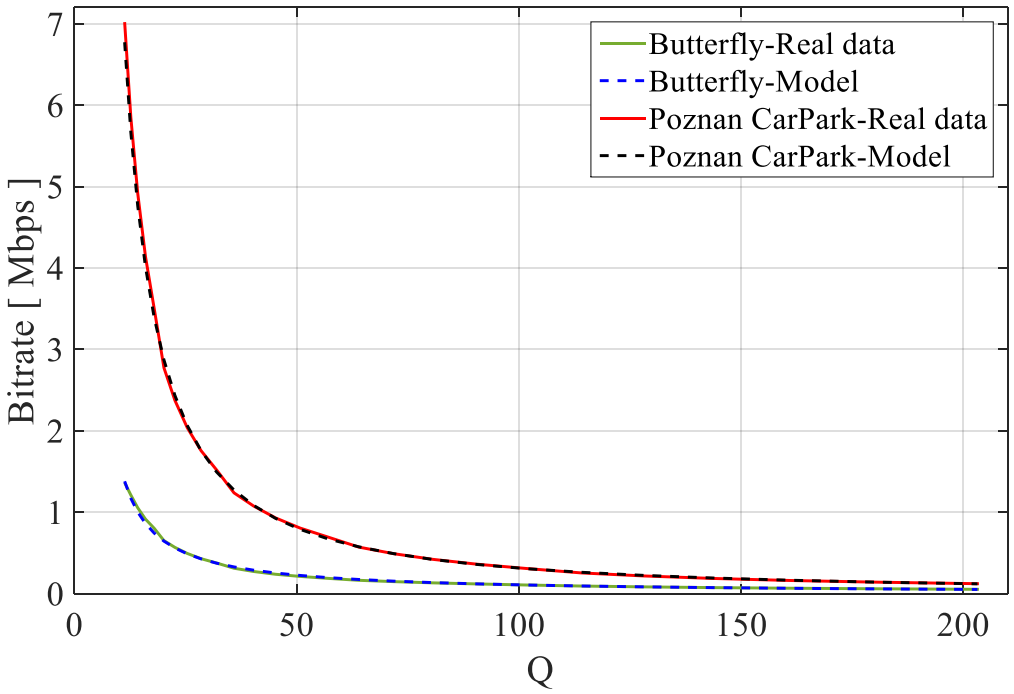


Fig. 7.7. The experimental and approximated (curve estimated by the model (Eq. 7.3)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the MV-HEVC coding.

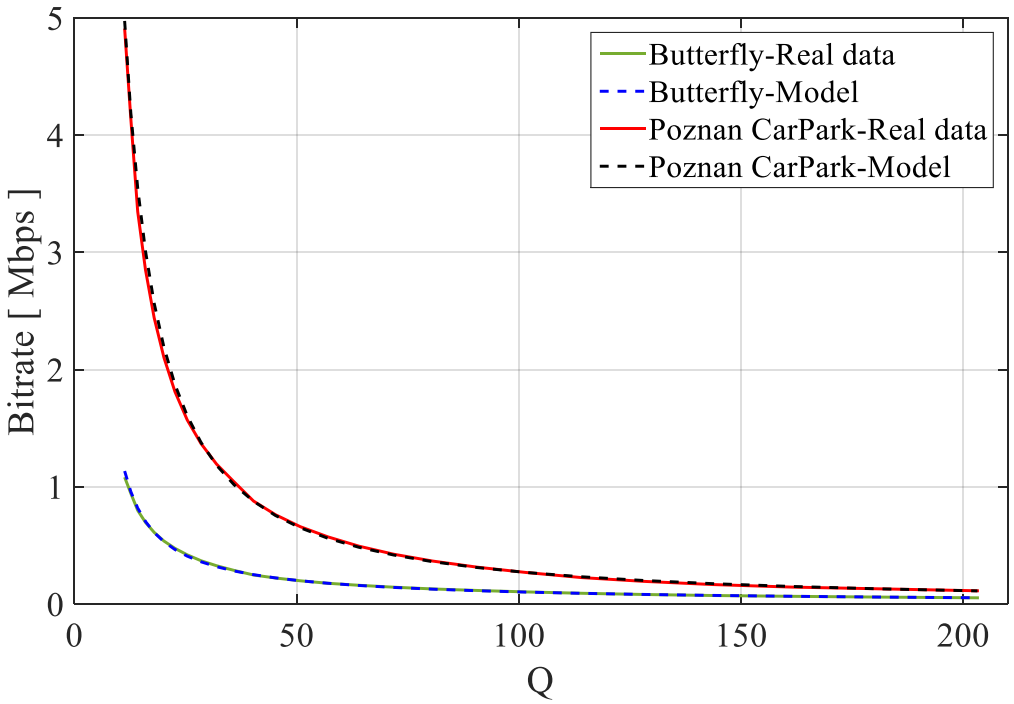


Fig. 7.8. The experimental and approximated (curve estimated by the model (Eq. 7.3)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the 3D-HEVC coding.

The performed experiments (Table 7.2, and Figs. 7.5, 7.6, 7.7, and 7.8) showed that the error between the experimental and approximated (curve estimated using the model of Eq. 7.3) curves of the considered MVD sequences is low, not exceeding 7% on average. Based on the results of experiments (Tables 7.1 and 7.2), it has also been noticed that the accuracy of the model of Eq. 7.3 is about 2 percentage points lower than the accuracy of the model of Eq. 7.1, but the accuracy is acceptable, and only two parameters are estimated instead of three parameters.

7.4.2 A Model with One Parameter

Based on the results of experiments in Subsection 7.3 (Table 7.1), it has been observed that the values of parameters *b* and *c* are about 1.11 and -3.5, respectively, for the considered MVD sequences for several codecs. Therefore, it is assumed in this approach that parameters *b* and *c* are equal to 1.11 and -3.5, respectively, for all MVD sequences for compression techniques. As a result, this approach uses one parameter, instead of three, that depends on the video content. The model used in this approach is the following:

$$R(Q, \emptyset) = \frac{a}{Q^{1.11-3.5}}, \tag{7.4}$$

where:

R - estimated bitrate for two videos and two depth maps by the proposed model with one parameter,

$\emptyset = [a]$ - parameter of the model that depends on sequence content,

Q - quantization step size for video.

Based on Eq. 7.2, the model parameter is estimated by minimizing the error between the experimental and approximated curves, as mentioned in Subsection 7.3.

The experimental and approximated curves (curve estimated using the model of Eq. 7.4) of bitrate for the *Poznan_CarPark* and *BBB.Butterfly* sequences for various codecs are shown in Figs. 7.9, 7.10, 7.11, and 7.12. Furthermore, Table 7.3 presents the values of the mean relative approximation error for the used MVD sequences, with more details in Appendix L (Tables L.1-L.4).

Table 7.3: Mean relative approximation error for the bitrate of considered sequences.

Codec	Relative error [%]	
	mean	std. dev.
HEVC	8.72	8.06
VVC	7.42	7.19
MV-HEVC	8.40	5.76
3D-HEVC	5.84	4.66

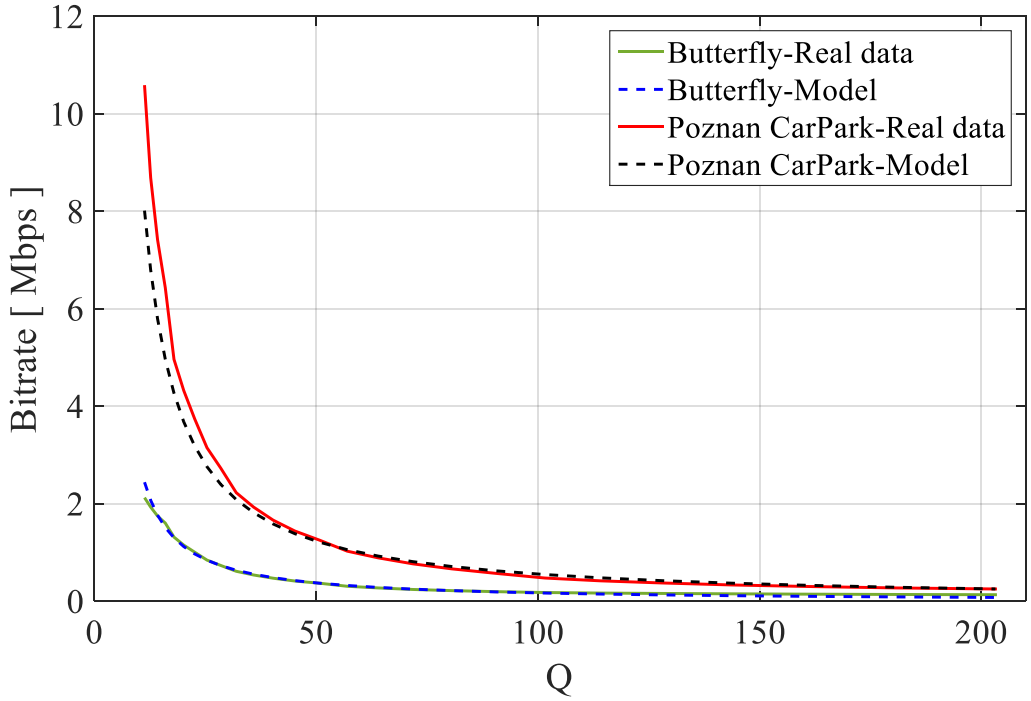


Fig. 7.9. The experimental and approximated (curve estimated by the model (Eq. 7.4)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the HEVC coding.

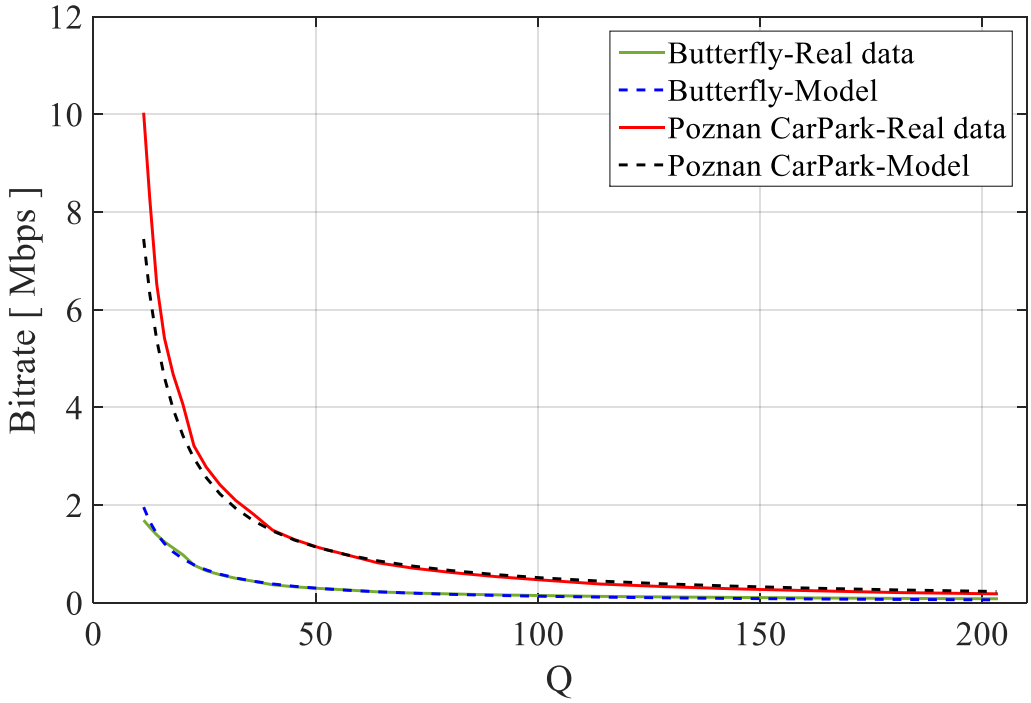


Fig. 7.10. The experimental and approximated (curve estimated by the model (Eq. 7.4)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the VVC coding.

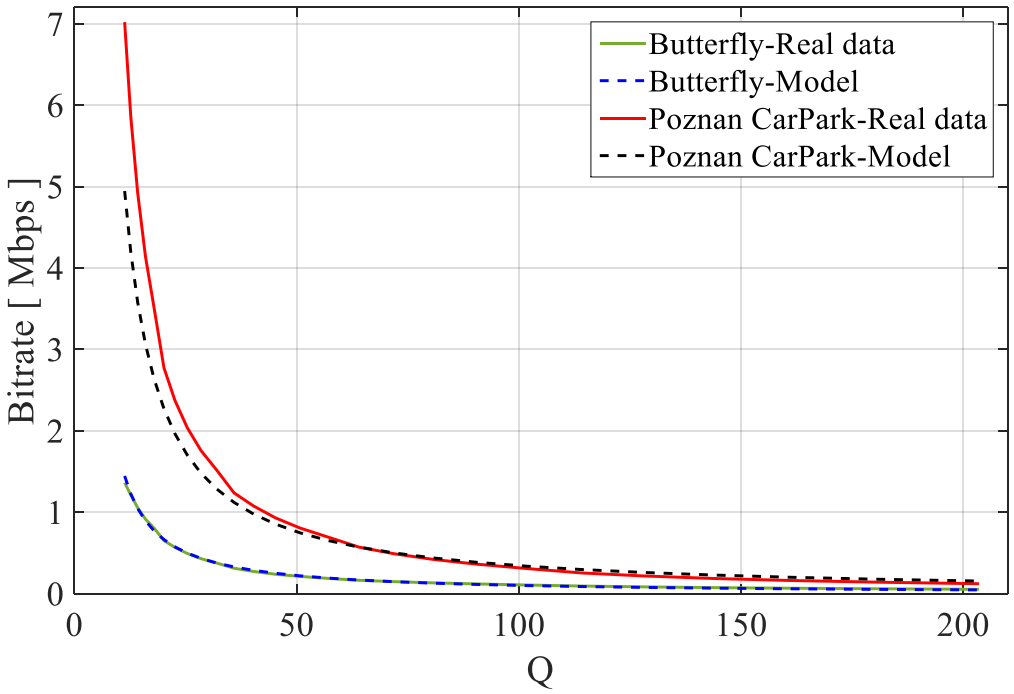


Fig. 7.11. The experimental and approximated (curve estimated by the model (Eq. 7.4)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the MV-HEVC coding.

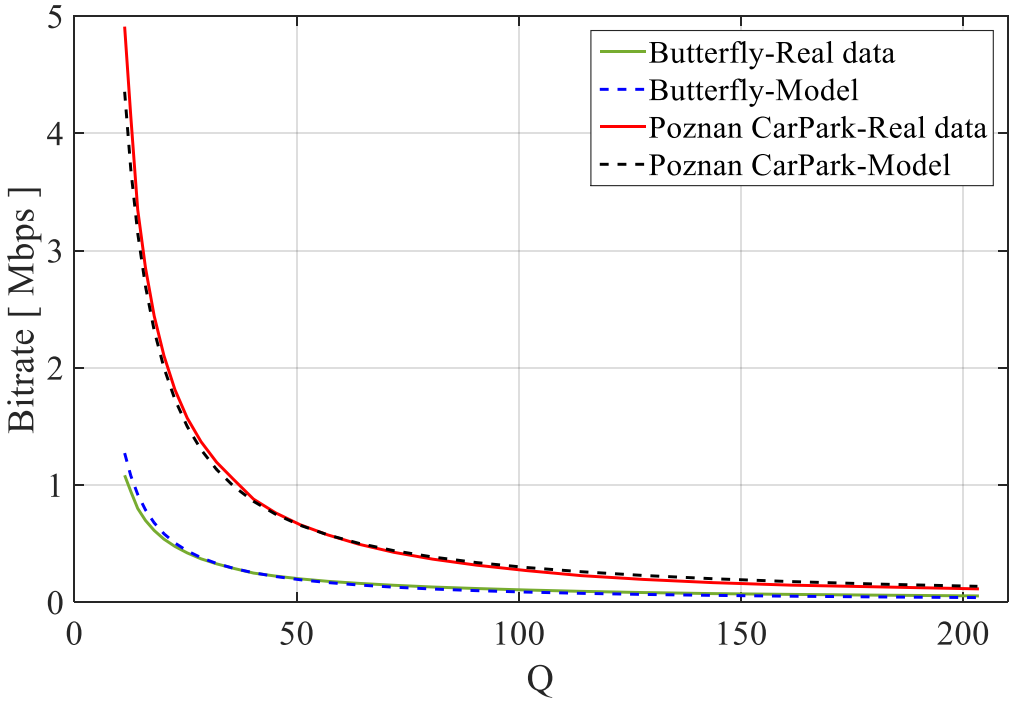


Fig. 7.12. The experimental and approximated (curve estimated by the model (Eq. 7.4)) curves for bitrate for *Poznan_CarPark* and *BBB.Butterfly* sequences for the 3D-HEVC coding.

Based on the results of experiments (Tables 7.1, 7.2, and 7.3), it has been observed that the accuracy of the model of Eq. 7.4 is about 4 and 2 percentage points lower than the accuracy of the model of Eq. 7.1 and the accuracy of the model of Eq. 7.3, respectively, but the accuracy is acceptable, and only one parameter is estimated.

7.5 R-Q Model Used for Frame-Level Bitrate Control

The R-Q model controls the frame size for MVD sequences based on the quantization step size. Thus, the following model is applied to match the experimental data:

$$B(Q,\emptyset) = \frac{a}{Q^b + c} \ , \tag{7.5}$$

where:

B - frame size estimated for two videos and two depth maps by the proposed model with three parameters,

$\emptyset = [a\ b\ c]$ - parameters of the model that depend on sequence content,

Q - quantization step size for a given frame type.

Based on the trust-region optimization method with the number of iterations equal to 2×10^6 and with the tolerance value of 1×10^{-6} (shown in Section 3.7), the model parameters are estimated by minimizing the error between the experimental and approximated curves, and the best estimate of the model parameters means getting the smallest error between these curves. The accuracy of the approximation of the experimental data is measured as follows [Graj_10, Choi_13, Guo_15]:

$$Relative\ error\ \ (Q,\emptyset) = \frac{|B_{MVD}(Q) - B\ (Q,\emptyset)|}{B_{MVD}(Q)}, \tag{7.6}$$

where:

Relative error (Q, Ø) - relative approximation error,

$B_{MVD}\ (Q)$ - frame size of the two videos and two depth maps,

$B\ (Q,\emptyset)$ - estimated frame size using the proposed model.

The number of GOP used in the experiments to estimate model parameters and errors is three, and each GOP structure used is I B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3 P B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3 I.

Table 7.4 shows the average relative approximation error calculated individually for each frame type for the considered sequences. The values of parameters *a*, *b*, and *c* shown in Table 7.4 are computed by finding mean values (average values) of the values of the parameters *a*, *b*, and *c*, respectively, for the considered MVD sequences; see more details in Appendix M (Table M.1- Table M.12).

Table 7.4: Mean relative approximation error for the frame size of different frame types of considered sequences.

Codec	Frame type	*a*	*b*	*c*	Relative error [%]	
					mean	std. dev.
HEVC	I	5648430	1.08	2.59	2.91	2.17
	P	2532740	1.18	2.48	6.02	4.17
	B0	2812274	1.27	2.43	7.03	5.94
	B1	3673775	1.25	4.76	9.74	6.30
	B2	3549056	1.36	3.57	11.18	6.82
	B3	4404238	1.51	2.66	11.49	6.78
VVC	I	26395.75	1.06	3.16	2.28	1.94
	P	27009	1.22	2.22	5.97	4.37
	B0	9883	1.21	4.26	5.82	5.18
	B1	9848	1.23	2.19	6.46	5.78
	B2	10153	1.23	3.00	7.51	7.28
	B3	12625	1.24	3.00	10.08	8.34

Figs. 7.13 and 7.14 explain the experimental and approximated curves of the frame size of different frame types for *Breakdancers* and *Poznan_Fencing* sequences. In Figs. 7.13 and 7.14, the approximate curves are calculated using the model used with parameters' values shown in Appendix M (Table M.1- Table M.12).

The performed experiments (Table 7.4, and Figs. 7.13 and 7.14) showed that the accuracy of the model is higher for large frames (I and P frames). However, the efficiency decreases for frames with smaller numbers of bits (B0, B1, B2, and B3 frames) because even for a small difference between the experimental and approximated data, the error will be clearly observed.

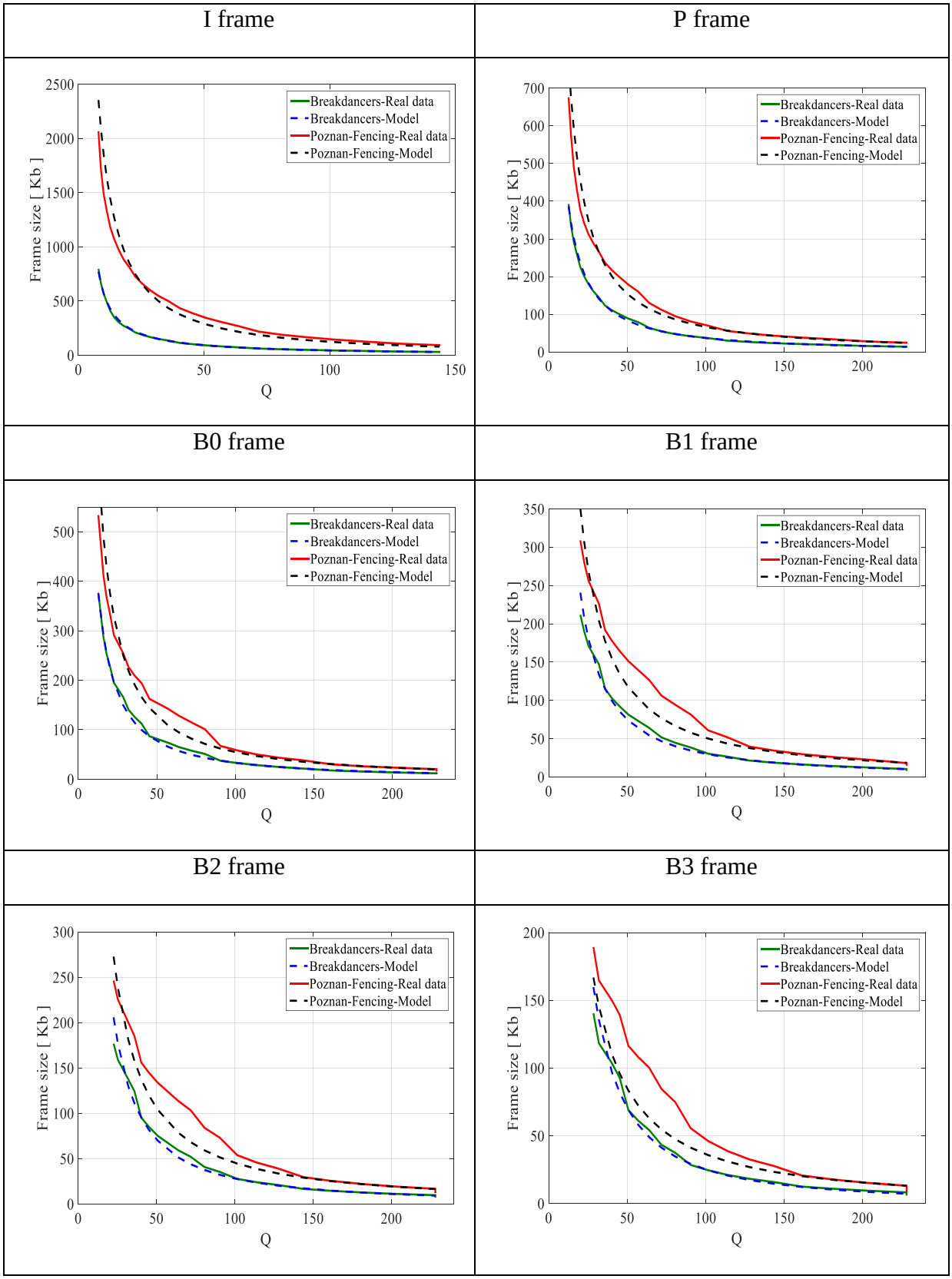


Fig. 7.13. The experimental and approximated (curve estimated by the model (Eq. 7.5)) curves for the frame size of different frame types for *Breakdancers* and *Poznan_Fencing* sequences for HEVC.

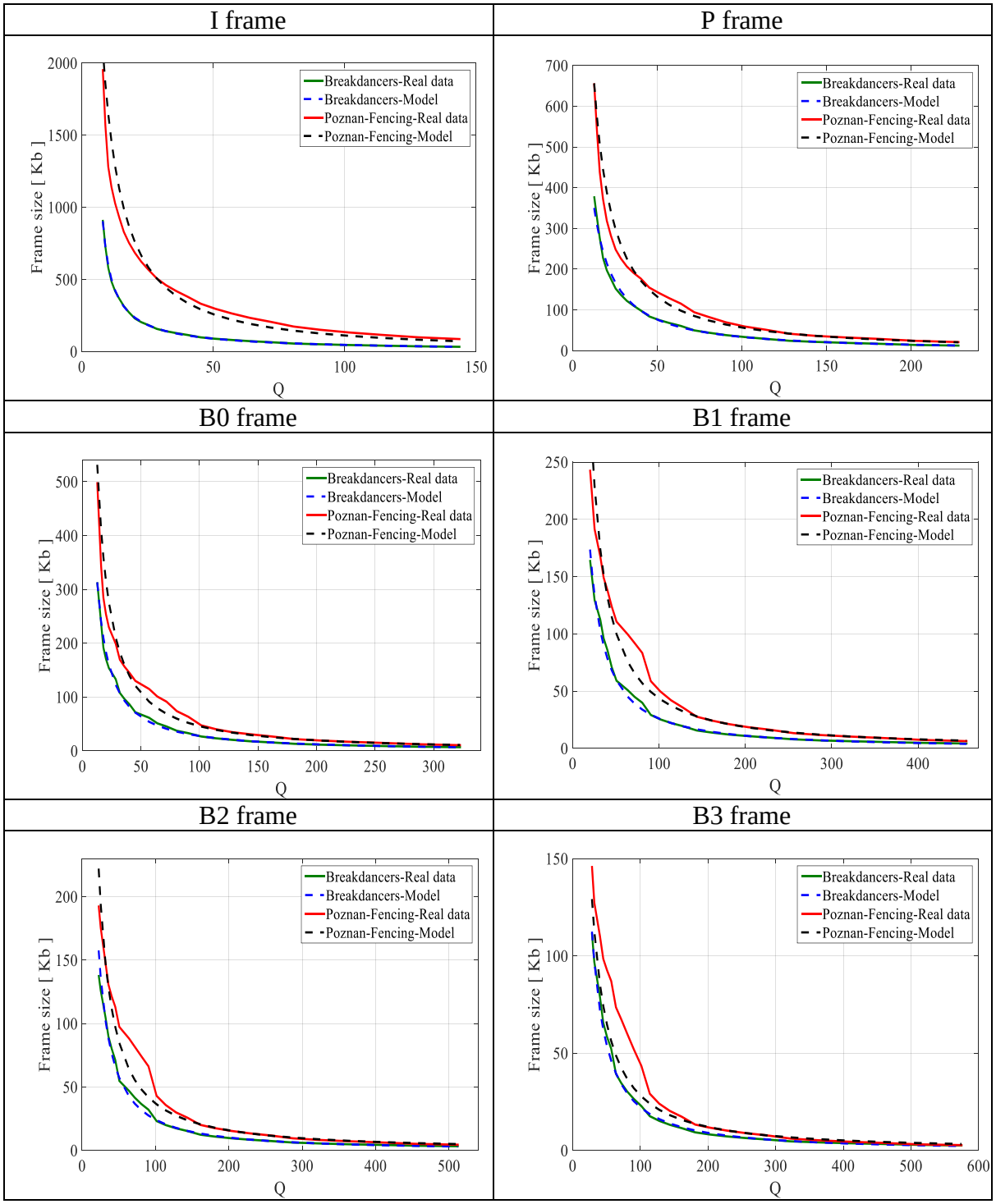


Fig. 7.14. The experimental and approximated (curve estimated by the model (Eq. 7.5)) curves for the frame size of different frame types for *Breakdancers* and *Poznan_Fencing* sequences for VVC.

7.6 Simplified Models for Frame-Level Bitrate Control

According to the results of experiments shown in Table 7.4, it has been observed that values of some of the model parameters are approximately similar for the considered MVD sequences. Therefore, model parameters can be reduced by using the following two approaches. In the experiments to estimate model parameters and errors, the number of GOP used for each codec is

three, and each GOP structure used is I B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3 P B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3 I.

7.6.1 A Model with Two Content-Dependent Parameters

Based on Table 7.4, the values of parameter *c* are about 3 for the considered MVD sequences for the used codecs (HEVC and VVC). Therefore, it has been assumed in this approach that parameter *c* is equal to 3 for all MVD sequences for the used codecs. As a result, this approach uses two parameters, instead of three, that depend on the video content. The model used in this approach is the following:

$$B(Q, \emptyset) = \frac{a}{Q^b + 3}, \tag{7.7}$$

where:

B - frame size estimated for two videos and two depth maps by the proposed model with two parameters,

$\emptyset = [a\ b]$ - parameters of the model that depend on sequence content,

Q - quantization step size for a given frame type.

Based on Eq. 7.6, the model parameters are estimated by minimizing the error between the experimental and approximated curves, as mentioned in the previous subsection.

The values of the mean relative approximation error for the considered MVD sequences are presented in Table 7.5; see more details in Appendix N (Table N.1- Table N.12).

Table 7.5: Mean relative approximation error for the frame size of different frame types of considered sequences.

Codec	Frame type	Relative error [%]	
		mean	std. dev.
HEVC	I	4.48	2.69
	P	6.69	4.60
	B0	8.14	6.49
	B1	10.26	6.61
	B2	11.35	6.94
	B3	11.58	6.83
VVC	I	4.62	3.09
	P	7.26	5.36
	B0	8.75	6.15
	B1	8.50	6.97
	B2	9.53	7.87
	B3	12.31	9.08

The experimental and approximated curves of the frame size of different frame types for the *Breakdancers* and *Poznan_Fencing* sequences for codecs are demonstrated in Figs. 7.15 and 7.16.

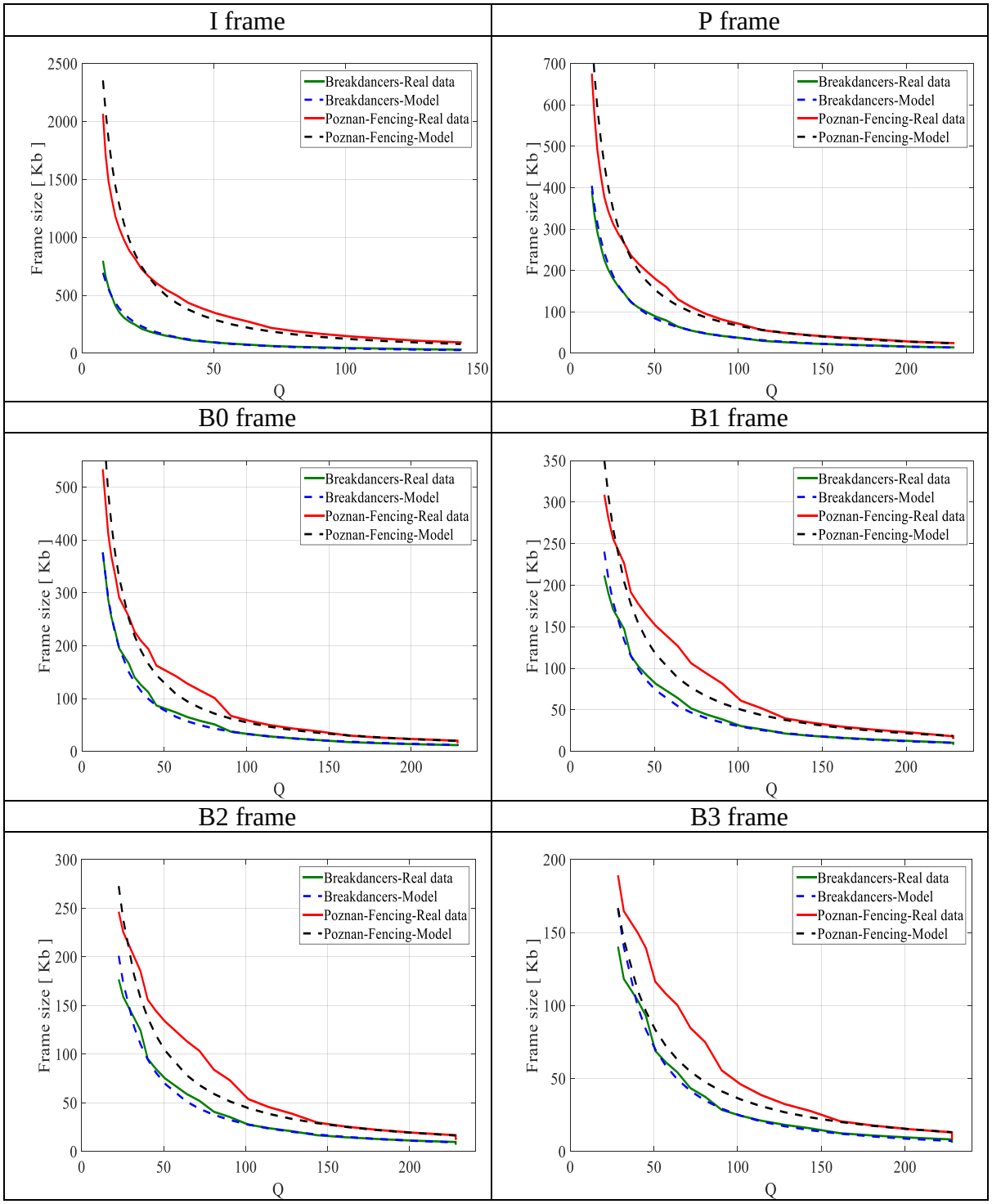


Fig. 7.15. The experimental and approximated (curve estimated by the model (Eq. 7.7)) curves for the frame size of different frame types for *Breakdancers* and *Poznan_Fencing* sequences for HEVC.

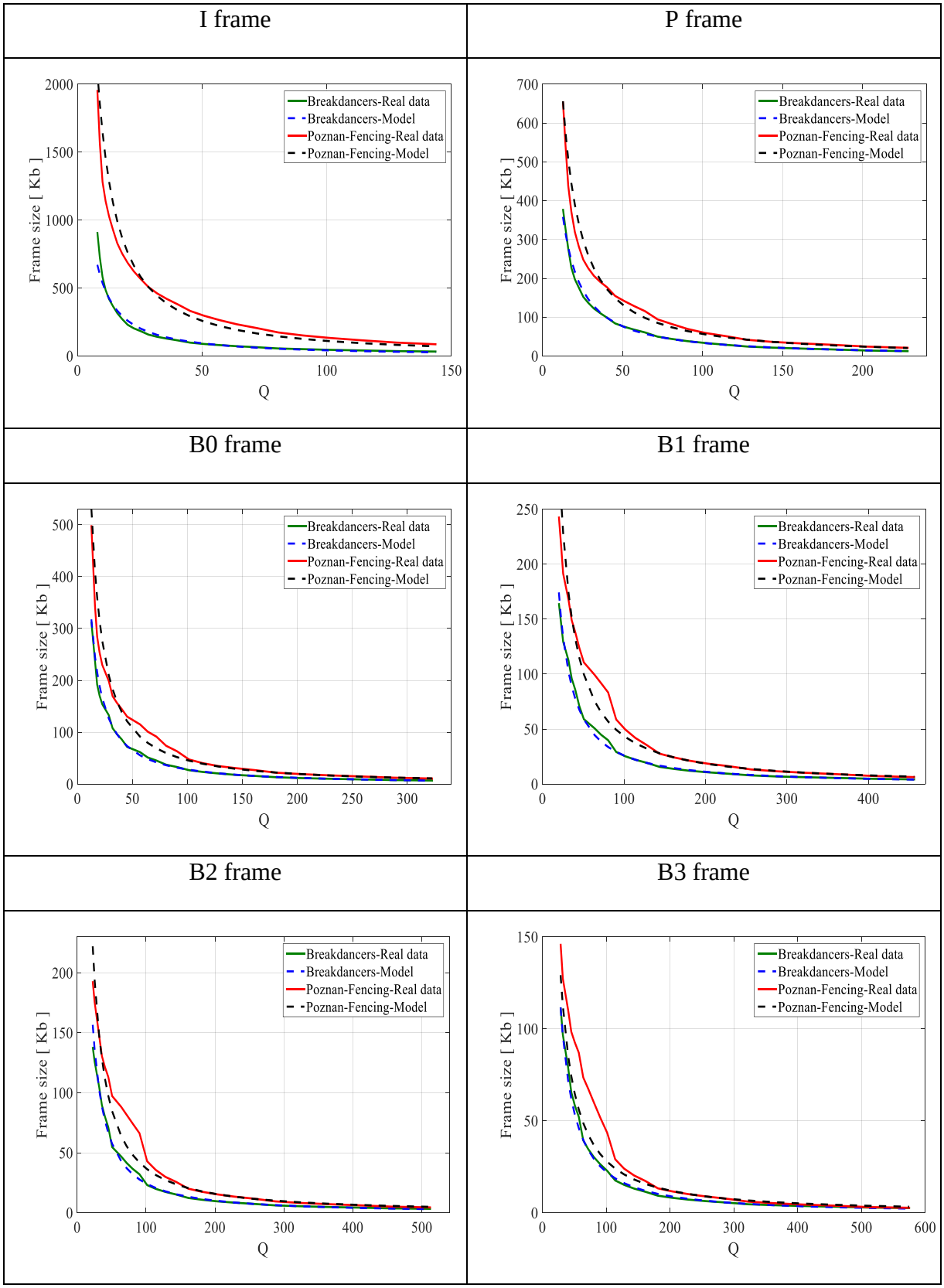


Fig. 7.16. The experimental and approximated (curve estimated by the model (Eq. 7.7)) curves for the frame size of different frame types for *Breakdancers* and *Poznan_Fencing* sequences for VVC.

According to the performed experiments (Table 7.5, and Figs. 7.15 and 7.16), it is noted that the efficiency of the model (Eq.7.7) is higher for large frames (I and P frames), but the accuracy decreases for small frames (B frames), which mean that the results of the model of Eq.7.7 are roughly similar to the results of the model of Eq.7.5. Based on the experimental results (Tables 7.4 and 7.5), it has been observed that the accuracy of the model of Eq. 7.7 is about 1 and 2 percentage points lower than the accuracy of the model of Eq. 7.5 for HEVC and VVC coding, respectively, but the accuracy is acceptable, and only two parameters are estimated instead of three parameters.

7.6.2 A Model with One Content-Dependent Parameter

In the experiments (Table 7.4), it has been observed that the values of parameters *b* and *c* are about 1.24 and 3, respectively, for the considered MVD sequences for the used codecs (HEVC and VVC). Therefore, it is assumed in this approach that parameters *b* and *c* are equal to 1.24 and 3, respectively, for all MVD sequences for the used codecs. As a result, this approach uses one parameter, instead of three, that depends on the video content. The model used in this approach is the following:

$$B(Q, \emptyset) = \frac{a}{Q^{1.24}+3}, \tag{7.8}$$

where:

B - frame size estimated for two videos and two depth maps by the proposed model with one parameter,

Q - quantization step size for a given frame type,

$\emptyset = [a]$ - parameter that depends on sequence content.

Based on Eq. 7.6, the model parameter is estimated by minimizing the error between the experimental and approximated curves, as mentioned in Subsection 7.5.

Figs.7.17 and 7.18 illustrate experimental and approximated curves of the frame size of different frame types for the *Breakdancers* and *Poznan_Fencing* sequences for codecs. The values of the mean relative approximation error for the considered sequences are explained in Table 7.6; see more details in Appendix O (Table O.1 – Table O.12).

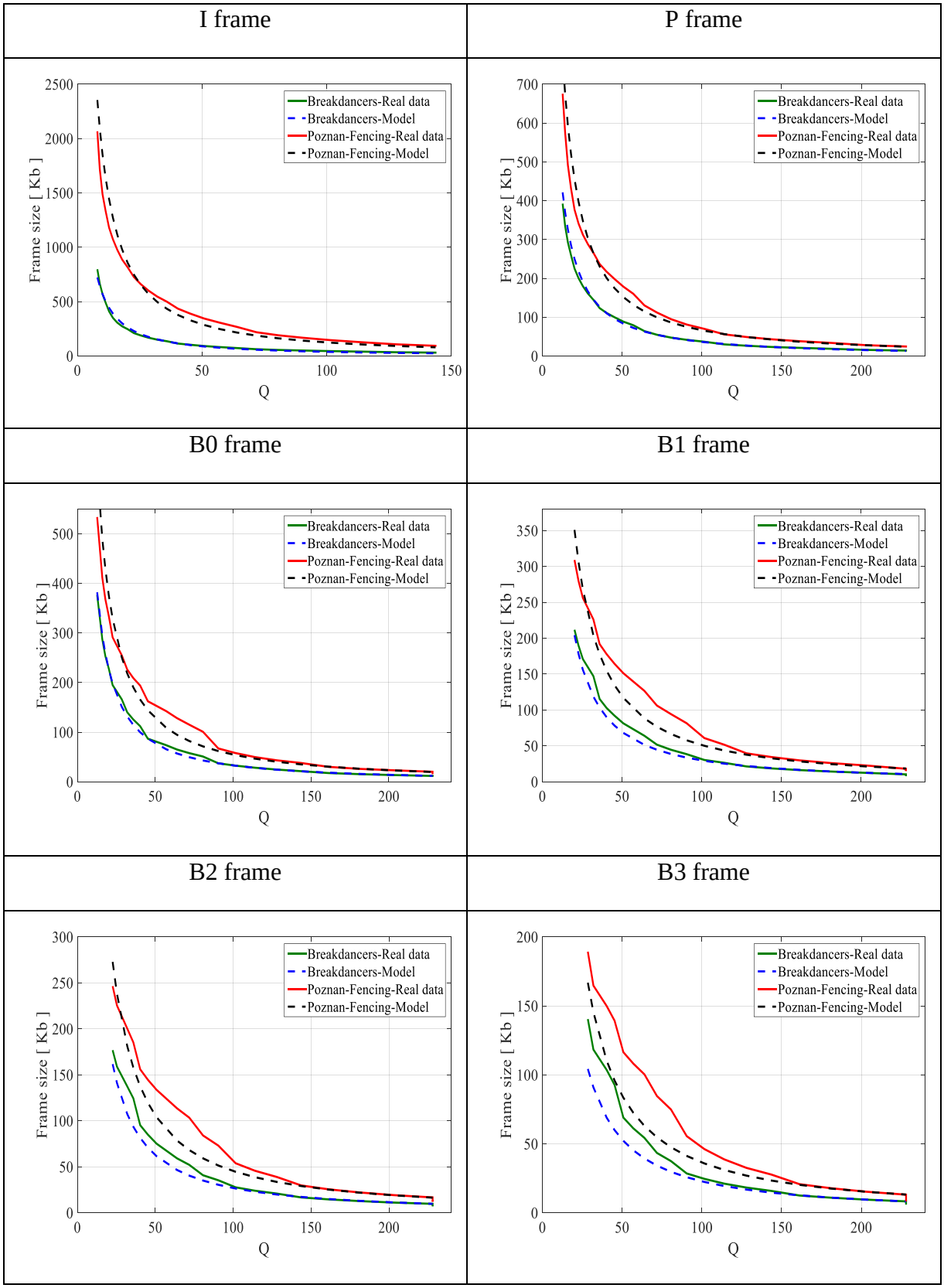


Fig. 7.17. The experimental and approximated (curve estimated by the model (Eq. 7.8)) curves for the frame size of different frame types for *Breakdancers* and *Poznan_Fencing* sequences for HEVC.

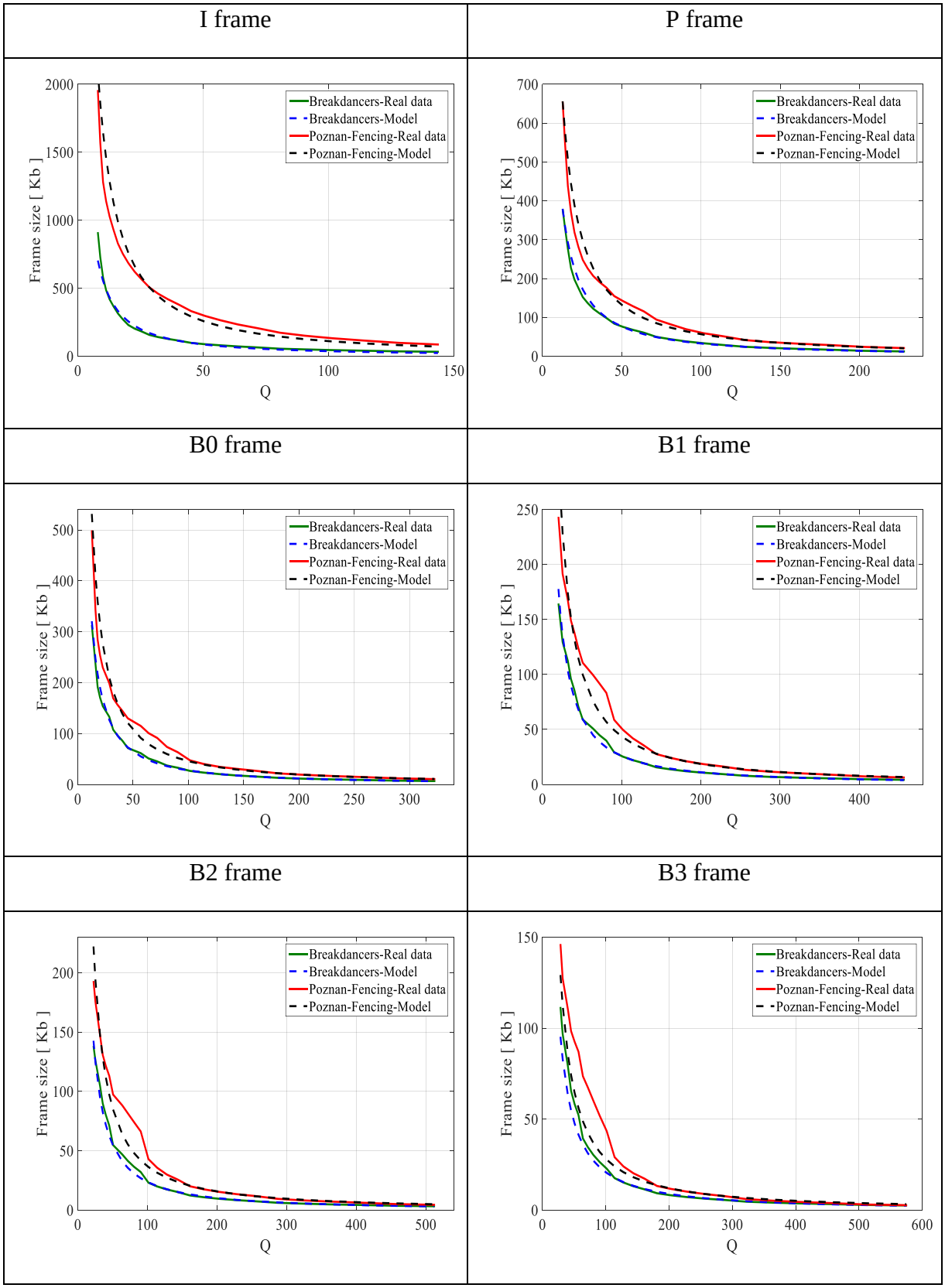


Fig. 7.18. The experimental and approximated (curve estimated by the model (Eq. 7.8)) curves for the frame size of different frame types for *Breakdancers* and *Poznan_Fencing* sequences for VVC.

Table 7.6: Mean relative approximation error for the frame size of different frame types of considered sequences.

Codec	Frame type	Relative error [%]	
		mean	std. dev.
HEVC	I	12.71	7.36
	P	9.24	7.66
	B0	11.53	10.26
	B1	12.81	8.74
	B2	15.77	10.77
	B3	19.38	14.31
VVC	I	13.19	7.66
	P	10.76	7.66
	B0	11.69	11.65
	B1	11.08	9.74
	B2	13.02	11.39
	B3	17.34	13.89

From the results of experiments (Tables 7.4, 7.5, and 7.6), it has been noticed that the inaccuracy of the model of Eq. 7.8 is about 6 and 5 percentage points higher than the inaccuracy of the model of Eq. 7.5 and the inaccuracy of the model of Eq. 7.7, respectively. However, the accuracy is still acceptable, and only one parameter is estimated instead of three parameters. B frames use smaller numbers of bits compared to the numbers of bits for I and P frames, thus the error of the model can be clearly observed for any small difference between the experimental and approximated data, as shown in Figs. 7.17 and 7.18.

7.7 Conclusions

The model proposed in [Graj_10] has been used to control the bitrate and frame size for stereoscopic video plus depth for many compression techniques, depending on the bit allocation models obtained in Subsection 5.4.2. In the experiments that have been performed on different sequences for different codecs, the results demonstrate the effectiveness of the proposed method.

For GOP-level bitrate control, the average relative approximation error for the model proposed in Eq. 7.1 for MVD sequences is about 6% for HEVC and 3% for VVC, MV-HEVC, and 3D-HEVC. While fixing one parameter value, a model with two content-dependent parameters will be applied (Eq. 7.3). In such a case the average approximation error for MVD sequences is about 7%, 5%, 5%, and 3% for HEVC, VVC, MV-HEVC, and 3D-HEVC, respectively. When fixing the value of two parameters, a model with one content-dependent parameter will be applied (Eq. 7.4). For this model the mean approximation error for sequences is about 9%, 7%, 8%, and 6% for HEVC, VVC, MV-HEVC, and 3D-HEVC, respectively.

For Frame-level bitrate control, the average relative approximation errors for the model proposed in Eq. 7.5 for MVD sequences are about 2% for I-frames, 6% for P-frames, 6% for B0-frames, 8% for B1-frames, 10% for B2-frames, and 11% for B3-frames. Whereas fixing a single parameter value, so a model with two content-dependent parameters will be applied (Eq. 7.7), the average approximation errors are about 4%, 7%, 8%, 9%, 10%, and 12% for I, P, B0, B1, B2, and B3 frames, respectively. When fixing the value of two parameters, a model with one content-dependent parameter will be applied (Eq. 7.8), the mean approximation errors are about 13%, 10%, 12%, 12%, 14%, and 18% for I, P, B0, B1, B2, and B3 frames, respectively.

Increasing the number of model parameters that depend on the video content gives the model the ability to represent any bitrate curve with high accuracy. Therefore, in the experiments, it can be observed that the model with more parameters dependent on the sequence content is more accurate than the model with fewer parameters. The proposed model that uses more parameters (e.g., the model of Eq. 7.1 for GOP-level bitrate control and the model of Eq. and 7.5 for frame-level bitrate control) is a more complicated one to estimate bitrate and frame size for stereoscopic video plus depth compared to other proposed models that use fewer parameters (e.g., models of Eqs. 7.3 and 7.4 for GOP-level bitrate control and models of Eqs.7.7 and 7.8 for frame-level bitrate control). However, by comparing the accuracy of the results for the proposed models, it is noticed that the difference is small.

Chapter Eight

Bitrate Control for Stereoscopic Video plus Depth

8.1 Purpose of the Work

Rate control is the process responsible for adjusting the quantization step to change the bitrate of the compressed video such that the limitation of communication channel throughput or storage capacity can be met. Constant bit rate (CBR) or variable bit rate (VBR) algorithm can be used for bitrate control, as shown in Section 2.2. In applications with constant bitrate, bitrate control algorithms are primarily focused on improving the accuracy of matching between the required bitrate and actual bitrate and meeting the limitations of low latency and buffer size. A buffer is used to reduce any alteration in bitrate that can happen [Liu_14, Li_20c]. Many practical applications like digital television, video calls and satellite-based video communication work at a constant bit rate (CBR), as shown in Section 2.2. Thus, this chapter is limited to the consideration of bitrate control according to the CBR scenario.

Bitrate control for stereoscopic video plus depth is more complicated than for the classic 2D video because it deals with two components (views and depth maps) instead of one. As mentioned in Section 1.1, the quality and bitrate of videos are controlled by the quantization step (Q) [Lim_07, Bai_17, Cai_20]. Consequently, two Q values are employed to control the quality and bitrate for stereoscopic video plus depth, one Q value for views and the other for depth maps. In Chapters 5 and 7, the bitrate allocation model (*QP-QD* model) and the encoder model (R-Q model) for stereoscopic video plus depth have been proposed, and the results showed the efficiency of the models. Consequently, this chapter aims to present an approach to bitrate control for stereoscopic video plus depth for several compression techniques (HEVC, VVC, MV-HEVC, 3D-HEVC) based on the bitrate allocation models ($QD = f(QP)$) obtained in Subsection 5.4.2 and the encoder model (R-Q model) for stereoscopic video plus depth shown in the previous chapter. Also, this chapter demonstrates the practical usefulness of bitrate control models presented in the previous chapters.

8.2 Bitrate Control Using the Encoder Model

Bitrate control for stereoscopic video plus depth depends on the encoder model and bitrate allocation model. The encoder model (R-Q model) aims to calculate the relationship between the bitrate and the quantization step for video. Whereas the bitrate allocation model (*QP-QD* model) estimates the relationship between the quantization parameters for video (QP) and quantization parameters for depth (QD) to obtain the maximum quality of a synthesized view at a given bitrate.

The initial quantization parameter for video ($QP_{initial}$) that has to be encoded at the desired bitrate is found using any method or just using an arbitrary value from the reasonable interval of the quantization parameter. The initial quantization parameter for depth ($QD_{initial}$) is

calculated from the $QP_{initial}$ according to the bitrate allocation model presented in Subsection 5.4.2. The initial bitrate is calculated by MVD coding using $QP_{initial}$ and $QD_{initial}$. Based on the initial bitrate and $QP_{initial}$, the encoder model is determined. The final quantization parameter for the video and depth map is computed depending on the required bitrate and the parameter of the encoder model estimated in the previous step.

Based on experiments shown in Chapter 7, the accuracy of the results of the encoder model with one content-dependent parameter is acceptable. Therefore, this encoder model will be applied and verified to GOP-level bitrate control and frame-level bitrate control in this chapter. For GOP-level bitrate control, the encoder model describes the relationship between bitrate and the quantization step for video. In contrast, the encoder model represents the relationship between frame size and the quantization step for a given frame type in frame-level bitrate control.

The procedure of the bitrate control may be described by the following steps:

Step 1: After encoding a frame or GOP with some guessed initial ($QP_{initial}$), measure $Bitrate_{initial}$.

Step2: Calculate the initial quantization step ($Q_{initial}$) from $QP_{initial}$ according to Eq. 2.1.

Step3: Calculate parameter *a* depending on the following:

$$Bitrate_{initial} = \frac{a}{Q_{initial}^{b} - c}, \tag{8.1}$$

Step4: Compute the quantization step ($Q_{calculated}$) depending on the parameter a and the target bitrate ($Bitrate_{goal}$) produced by MVD coding using QP_{goal} and QD_{goal} (the target quantization parameter for the depth map), as follows:

$$Bitrate_{goal} = \frac{a}{Q_{calculated}^{b} - c}. \tag{8.2}$$

Step5: Determine the calculated quantization parameter for the video ($QP_{calculated}$) from $Q_{calculated}$ based on Eq. 2.1.

Step6: Compute the calculated quantization parameter for depth ($QD_{calculated}$) from $QP_{calculated}$ according to the bitrate allocation model ($QD = f(QP)$) obtained in Subsection 5.4.2.

The above-mentioned algorithm may be applied both for frames as well as for GOPs.

8.3 Simulation of Bitrate Control Using Encoder Model

The goal of the experiments is to assess the accuracy of bitrate control using the bitrate allocation models presented in Section 5.4.2 and the encoder model shown in Sections 7.3 and 7.5. The simulation corresponds to the following control scenario for stereoscopic video plus depth. The test is to estimate QP_{goal} (the target quantization parameter for video) and QD_{goal}

(the target quantization parameter for depth) for a group of pictures (GOP) such that the bitrate possibly well approximates the requested bitrate ($Bitrate_{goal}$). It has been assumed that an initial value $QP_{initial}$ (the initial quantization parameter for video) may be roughly estimated such there exists a certain error Δ such that

$$QP_{initial} = QP_{goal} \pm \Delta \tag{8.3}.$$

In the experiments, the simulation of the relevant estimation procedure for estimation of QP_{goal} and QD_{goal} is provided both for GOP-level bitrate control as well as frame-level bitrate control. The experiments are performed according to the following steps:

Step1: Calculate the initial quantization parameter for the video ($QP_{initial}$) from the target quantization parameter for video (QP_{goal}) according to Eq. 8.3, where Δ is an integer number and $\Delta \in [2-5] \in \mathbb{N}$.

Step2: Calculate the initial quantization parameter for depth ($QD_{initial}$) from $QP_{initial}$ according to the bitrate allocation model ($QD = f(QP)$) obtained in Subsection 5.4.2.

Step3: Determine the initial bitrate ($Bitrate_{initial}$) for stereoscopic video plus depth by MVD coding using $QP_{initial}$ and $QD_{initial}$.

Step4: Calculate the initial quantization step ($Q_{initial}$) from $QP_{initial}$ according to Eq. 2.1 ($Q = (2^{1/6})^{QP-4}$).

Step5: Calculate parameter *a* depending on the following:

$$Bitrate_{initial} = \frac{a}{Q_{initial}^{b} - c}, \tag{8.4}$$

where, the values of parameters *b* and *c* are constants, as listed in Tables 8.1 for the GOP-level bitrate control.

This approach will be applied to simulate GOP-level bitrate control and frame-level bitrate control using the encoder model.

Multiview sequences recommended by the Moving Picture Experts Group (MPEG) were used in the experiments. In this chapter, Breakdancers, Kermit, and Poznan_Block2 sequences (shown in Table 3.2) were used in order to assess the accuracy of the encoder model with sequences of diverse resolutions. The GOP structure used in the experiments for all codecs is I B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3 P B3 B2 B3 B1 B3 B2 B3 B0 B3 B2 B3 B1 B3 B2 B3 I. The interval of *QP* used in the experiments ranges from 25 to 50, which corresponds to practically used bitrates.

8.4 Experimental Results for GOP-Level Bitrate Control Based on Encoder Modeling

Experiments have been performed to evaluate the performance of the GOP-level bitrate control using the encoder model for stereoscopic video plus depth according to the block diagram presented in Fig 3.2. To encode stereoscopic video plus depth (two videos plus two depth maps), reference software for HEVC, VVC, MV-HEVC, and 3D-HEVC (shown in Table 3.3) was used.

Based on Table 7.1, the used values for parameters *b* and *c* of the encoder model (Eqs. 8.1, 8.2, and 8.4) to GOP-level bitrate control are presented in Table 8.1. The values of parameters *b* and *c* shown in Table 8.1 are calculated by finding mean values (average values) of the values of the parameters *b* and *c*, respectively, for the considered MVD sequences.

Table 8.1: Encoder model parameters for GOP-level rate control for different codecs (estimated in Subsection 7.3).

Codec	*b*	*c*
HEVC	1.01	-3.84
VVC	1.10	-3.33
MV-HEVC	1.16	-3.54
3D-HEVC	1.12	-2.70

The experiments were performed for all QP_{goal} in the range from 25 to 45. Tables 8.2, 8.3, 8.4, and 8.5 show the matching accuracy between the target quantization parameter for video (QP_{goal}) and the quantization parameter calculated ($QP_{calculated}$) according to the proposed method of bitrate control for stereoscopic video plus depth. The error σ is calculated by computing the difference between QP_{goal} and $QP_{calculated}$, as follows:

$$\sigma = QP_{goal} - QP_{calculated}. \tag{8.4}$$

The percentage values (i.e., accuracy) shown in Tables 8.2, 8.3, 8.4, and 8.5 represent the percentage of occurrences of each σ value to the total number of tests for QP_{goal} values from 25 to 45 for each sequence. The total number of tests is 42 for each Δ.

For the simulcast coding, the depth was compressed as a monochromatic component.

Table 8.2: Accuracy of the proposed method for GOP-level bitrate control for the HEVC coding (HEVC simulcast).

Sequences	Δ	σ=0	σ=±1	σ=±2
Breakdancers	2	71.88%	28.12%	0%
	3	59.38%	40.62%	0%
	4	50%	43.75%	6.25%
	5	34.38%	59.37%	6.25%
Kermit	2	93.75%	6.25%	0%
	3	75%	25%	0%
	4	59.38%	40.62%	0%
	5	43.75%	53.13%	3.12%
Poznan_Block2	2	100%	0%	0%
	3	93.75%	6.25%	0%
	4	90.63%	9.37%	0%
	5	81.25%	18.75%	0%

Table 8.3: Accuracy of the proposed method for GOP-level bitrate control for the VVC coding (VVC simulcast).

Sequences	Δ	σ=0	σ=±1	σ=±2
Breakdancers	2	65.63%	34.37%	0%
	3	65.63%	34.37%	0%
	4	40.63%	59.37%	0%
	5	34.38%	65.62%	0%
Kermit	2	93.75%	6.25%	0%
	3	78.13%	21.87%	0%
	4	59.38%	40.62%	0%
	5	53.13%	43.75%	3.12%
Poznan_Block2	2	100%	0%	0%
	3	100%	0%	0%
	4	90.63%	9.37%	0%
	5	75%	25%	0%

Table 8.4: Accuracy of the proposed method for GOP-level bitrate control for the MV-HEVC coding.

Sequences	Δ	σ=0	σ=±1	σ=±2
Breakdancers	2	78.13%	21.87%	0%
	3	62.50%	37.50%	0%
	4	56.25%	43.75%	0%
	5	53.13%	46.87%	0%
Kermit	2	93.75%	6.25%	0%
	3	75%	25%	0%
	4	68.75%	31.25%	0%
	5	65.63%	31.25%	3.12%
Poznan_Block2	2	100%	0%	0%
	3	100%	0%	0%
	4	93.75%	6.25%	0%
	5	90.63%	9.37%	0%

Table 8.5: Accuracy of the proposed method for GOP-level bitrate control for the 3D-HEVC coding.

Sequences	Δ	σ=0	σ=±1	σ=±2
Breakdancers	2	100%	0%	0%
	3	100%	0%	0%
	4	93.75%	6.25%	0%
	5	71.88%	28.12%	0%
Kermit	2	100%	0%	0%
	3	96.88%	3.12%	0%
	4	78.13%	21.87%	0%
	5	71.88%	25%	3.12%
Poznan_Block2	2	100%	0%	0%
	3	100%	0%	0%
	4	100%	0%	0%
	5	87.50%	12.50%	0%

In Tables 8.2, 8.3, 8.4, and 8.5, it has been noticed that the accuracy of the proposed method for GOP-level rate control is about 72%, 72%, 79%, and 92% for HEVC, VVC, MV-HEVC, and 3D-HEVC, respectively. During the transformation between Q and QP, the approximation process will reduce the efficiency of the proposed method. Changing the value of Δ affects the value of the quantization parameter calculated according to the method described in Section 8.3. If the value of Δ is high, the effect of the approximation process during the proposed method can be clearly observed in the value of the quantization parameter calculated. From Tables 8.2, 8.3, 8.4, and 8.5, it has been observed that increasing the value of Δ leads to a decrease in the accuracy of the proposed method.

8.5 Experimental Results for Frame-Level Bitrate Control Based on Encoder Modeling

According to the block diagram presented in Fig 3.2, experiments have been executed to assess the efficiency of the frame-level bitrate control using the encoder model for stereoscopic video plus depth. Reference software for HEVC and VVC (shown in Table 3.3) was used to encode stereoscopic video plus depth (two videos plus two depth maps). In order to assess the accuracy of the encoder model with sequences of various resolutions, Breakdancers and Poznan_Block2 sequences were used. The GOP structure used in the experiments for all codecs is described in Section 8.3. The *QP* offsets were set according to the MPEG common test conditions for individual frame types, shown in Table 3.3.

Based on Table 7.4, the used values for parameters *b* and *c* of the encoder model (Eqs. 8.1, 8.2, and 8.4) to frame-level bitrate control are shown in Table 8.6. The values of parameters *b* and *c* shown in Table 8.6 are calculated by finding mean values (average values) of the parameters *b* and *c*, respectively, for different frame types of the considered MVD sequences.

Table 8.6: Encoder model parameters for frame-level rate control for different codecs.

Codec	*a*	*b*
HEVC	1.28	3.08
VVC	1.20	2.97

The experiments were performed for all QP_{goal} in the range from 25 to 45. Tables 8.7 and 8.8 show the matching efficiency between the target quantization parameter for a given frame type and the quantization parameter calculated according to the proposed method of bitrate control for stereoscopic video plus depth. The percentage values shown in Tables 8.7 and 8.8 represent the percentage of the number of occurrences of each σ value to the total number of tests for all QP_{goal} for each sequence.

Table 8.7: Efficiency of the proposed method for frame-level bitrate control for the HEVC coding.

Frame type	Δ	Breakdancers			Poznan_Block2		
		δ=0	δ=±1	δ=±2	δ=0	δ=±1	δ=±2
I	2	90.63%	9.37%	0%	100%	0%	0%
	3	62.50%	37.50%	0%	59.38%	40.62%	0%
	4	43.75%	56.25%	0%	37.50%	62.50%	0%
	5	37.50%	62.50%	0%	31.25%	68.75%	0%
P	2	87.50%	12.50%	0%	100%	0%	0%
	3	65.63%	34.37%	0%	96.88%	3.12%	0%
	4	59.37%	40.63%	0%	93.75%	3.12%	3.13%
	5	56.25%	40.63%	3.12%	90.63%	3.12%	6.25%
B0	2	75%	25%	0%	79.17%	20.83%	0%
	3	66.67%	33.33%	0%	66.67%	33.33%	0%
	4	41.67%	58.33%	0%	66.67%	33.33%	0%
	5	37.50%	62.50%	0%	59.09%	40.91%	0%
B1	2	90.91%	9.09%	0%	95.45%	4.55%	0%
	3	68.18%	31.82%	0%	86.36%	13.64%	0%
	4	54.55%	45.45%	0%	77.27%	22.73%	0%
	5	45.45%	54.55%	0%	68.18%	31.82%	0%
B2	2	63.64%	36.36%	0%	95.45%	4.55%	0%
	3	59.09%	40.91%	0%	81.82%	18.18%	0%
	4	54.55%	45.45%	0%	68.18%	31.82%	0%
	5	50%	50%	0%	63.64%	36.36%	0%
B3	2	59.09%	40.91%	0%	90.91%	9.09%	0%
	3	59.09%	40.91%	0%	65%	35%	0%
	4	54.55%	45.45%	0%	63.64%	36.36%	0%
	5	36.36%	63.64%	0%	59.09%	40.91%	0%

Table 8.8: Efficiency of the proposed method for frame-level bitrate control for the VVC coding.

Frame type	Δ	Breakdancers			Poznan_Block2		
		σ=0	σ=±1	σ=±2	σ=0	σ=±1	σ=±2
I	2	90.63%	9.37%	0%	100%	0%	0%
	3	65.63%	34.37%	0%	93.75%	6.25%	0%
	4	40.63%	56.25%	3.12%	68.75%	28.13%	3.12%
	5	25%	68.75%	6.25%	43.75%	50%	6.25%
P	2	100%	0%	0%	96.88%	3.12%	0%
	3	87.50%	12.50%	0%	93.75%	3.12%	3.13%
	4	78.12%	21.88%	0%	90.63%	3.12%	6.25%
	5	59.38%	37.50%	3.12%	84.38%	6.25%	9.37%
B0	2	95.83%	4.17%	0%	95.83%	4.17%	0%
	3	79.17%	20.83%	0%	87.50%	12.50%	0%
	4	54.55%	45.45%	0%	81.82%	18.18%	0%
	5	50%	50%	0%	81.82%	18.18%	0%
B1	2	68.18%	31.82%	0%	81.82%	18.18%	0%
	3	63.64%	27.27%	9.09%	77.27%	22.73%	0%
	4	50%	50%	0%	72.73%	27.27%	0%
	5	45.45%	54.55%	0%	72.73%	27.27%	0%
B2	2	72.73%	27.27%	0%	68.18%	31.82%	0%
	3	63.64%	36.36%	0%	63.64%	36.36%	0%
	4	45.46%	45.45%	9.09%	63.64%	36.36%	0%
	5	36.36%	63.64%	0%	63.64%	36.36%	0%
B3	2	63.64%	36.36%	0%	81.82%	18.18%	0%
	3	45.45%	54.55%	0%	77.27%	22.73%	0%
	4	22.73%	77.27%	0%	72.73%	27.27%	0%
	5	18.18%	72.73%	9.09%	54.55%	45.45%	0%

In Tables 8.7 and 8.8, it has been noticed that the efficiency of the proposed method for frame-level rate control is about 68% for frame types for the considered codecs. Changing the value of Δ affects the value of the quantization parameter calculated according to the steps described in Section 8.3. Consequently, the effect of the approximation process during these steps can be clearly noticed in the value of the quantization parameter calculated, especially when the value of Δ is high. Based on Tables 8.7 and 8.8, it has been observed that increasing the value of Δ leads to a decrease in the accuracy of the proposed method.

8.6 Conclusions

The chapter presents a method of bitrate control for stereoscopic video plus depth for several compression techniques. The presented method depends on the R-Q model discussed in the previous chapters, the relationship between *Q* and *QP* (shown in Eq.2.1), and the bitrate allocation model shown in Subsection 5.4.2. This chapter's results proved the efficiency of the proposed method to bitrate control for stereoscopic video plus depth.

For GOP-level bitrate control, the efficiency of the proposed method for bitrate control based on the bitrate allocation model and the used R-Q model for stereoscopic video plus depth is about 79%.

For frame-level bitrate control, the efficiency of the proposed method for bitrate control based on the bitrate allocation model and the used R-Q model is about 62%, 84%, 70%, 70%, 64%, and 58% for I, P, B0, B1, B2, and B3 frames, respectively.

The encoder model with one content-dependent parameter is the simplest one to control the bitrate for stereoscopic video plus depth compared to other models (the encoder model with three or two content-dependent parameters) because this model needs to estimate one parameter, instead of three or two parameters, to bitrate control. Thus, this encoder model requires less computation effort to bitrate control.

Based on experiments, the proposed method for bitrate control based on the bitrate allocation model and the used R-Q model can be applied to GOP- and frame-level bitrate control with good efficiency for stereoscopic video plus depth for many codecs. Consequently, the proposed method for bitrate control can be used in practical applications like networks, communication systems, dynamic adaptive streaming over HTTP (DASH), applications with real-time and/or low latency requirements, and jitter-sensitive applications (for example, video call and satellite-based video communication).

In comparison to other bitrate control studies, the R-λ model was proposed to control the bitrate in [Cord_16], but this study depends on the CTC approach to distribute the bitrate between videos and depth maps. In Chapter Five, the bitrate allocation model proposed in this dissertation has been proven to outperform the CTC approach. In [Liu_11], a simple R-Q model was used to control the bitrate. As shown in Subsection 2.3.1, the encoder model used in the dissertation outperforms the simple and quadratic R-Q model. Also, this research depends on the straightforward approach ($QP = QD$) to allocate bitrate between videos and depth maps. It has been proven in Chapter Five that the straightforward approach is less efficient than the bitrate allocation model proposed in this dissertation. The presented method for bitrate control for multiview video plus depth was shown in [Lie_14]. The study presented in [Lie_14] relies on the CTC approach to allocate bitrate between videos and depth maps, and this approach is less efficient than the bitrate allocation proposed in the dissertation, as mentioned earlier. The error of the presented method in [Lie_14] is about 30% for MVD sequences. Consequently, the proposed method to bitrate control for stereoscopic video plus depth for several compression techniques based on the bitrate allocation models obtained in Subsection 5.4.2 and the encoder model (R-Q model) is expected to outperform the other methods presented in previous studies.

Chapter Nine

Summary of the Dissertation

The dissertation focuses on issues related to modeling of the codecs for stereoscopic video plus depth. The main goal of the research presented in this dissertation is to study original encoder models and methods suitable for bitrate allocation and rate control for stereoscopic video plus depth for the major available video compression technologies.

The main achievements of the dissertation are enumerated in Section 9.1. Section 9.2 presents a general discussion of the results obtained in the dissertation. The future directions of research are offered in Section 9.3.

9.1 Main Achievements of the Dissertation

The primary achievements of the dissertation concern the development of a new approach to encoder modeling, rate control and bitrate allocation for stereoscopic video plus depth for different codecs:

- **Video and depth bitrate allocation**
  The optimum ratio between the bitrate of video and depth data for stereoscopic video plus depth maps has been derived with the goal to obtain the maximum quality of a synthesized view at a given total bitrate by establishing the relationship between the quantization parameters for video (*QP*) and quantization parameters for depth (*QD*). In the dissertation, the optimum (*QP*, *QD*) pairs for MVD sequences that form the best rate-distortion curve are found with the goal to derive the bitrate allocation models. By depending on optimum (*QP*, *QD*) pairs for MVD sequences, a new approach has been proposed to select the quantization parameter for depth based on the quantization parameter for video. Thus, this approach allows automatic determination of the quantization parameter for depth for many codecs for MVD sequences. This approach guarantees a near-optimum bitrate allocation between video and depth at any required bitrate. According to the results of the experiments, it has been observed that the proposed models of bitrate allocation for many codecs outperform the models presented in previous studies, e.g. the straightforward approach ($QP = QD$).

- **Rate control for stereoscopic video plus depth**
  The proposed method has been presented to control the bitrate of stereoscopic video plus depth for many codecs to meet various communication channel bandwidths based on two models: the encoder model (the model describes the relationship between the bitrate of stereoscopic video plus depth and the quantization step size for video) and the bitrate allocation model (the model describes the relationship between the quantization parameters for video and depth). The results of the experiments prove the effectiveness of the proposed method of rate control for stereoscopic video plus depth for several codecs by comparing target data and data calculated using the proposed approach for MVD sequences.

Moreover, in addition to the main achievements shown above, two secondary achievements of the dissertation that increase the usefulness of performed research are offered in this dissertation:

- **Study of the influence of depth map quality on the quality of synthesized views**
  A comprehensive study has been presented on the influence of depth map fidelity on the quality of the virtual view by comparing optimum R-D curves (optimum (*QP*, *QD*) pairs) for different qualities of depth maps for MVD sequences. The results show that the virtual view quality is affected by depth map quality.

- **Rate control for HEVC and VVC depending on AVC data**
  AVC is the least complicated encoder compared to HEVC and VVC, which needs the shortest time to calculate the number of video bits. A new method has been suggested to calculate rate control for HEVC and VVC based on AVC data, that is, to estimate the parameters of rate control models for HEVC and VVC depending on AVC data. This way, the required time is reduced to estimate the bitrate and frame size for HEVC and VVC. The presented results show the effectiveness of the proposed method of rate control for HEVC and VVC.

9.2 Discussion of the Results

In the previous section, the main and additional achievements of the dissertation are shown. Many models have been proposed in the dissertation to make those achievements possible.

The shapes of R-Q characteristics for different codecs (AVC, HEVC, and VVC) are roughly similar for the given content. Therefore, a novel approach to rate control for HEVC and VVC, depending on AVC data, was presented in Chapter 4. The results proved the effectiveness of this approach to rate control for HEVC and VVC by comparing the bitrate or frame size calculated using the proposed approach with real data for 2D sequences. The

accuracy of the proposed approach is suitable for I- and P-frames, while it is significantly lower for B-frames. Because B-frame sizes are small, the error can be clearly observed for any small difference between the experimental and approximated data.

In Chapter 5, the quantization parameter for depth is automatically selected according to the quantization parameter for a video to obtain the maximum quality of the virtual view at a given bitrate. The proposed model was calculated by relying on the optimum (*QP*, *QD*) pairs for MVD sequences. The presented experiments proved that the performance of the proposed models is better than the other approaches (other approaches are the models presented in previous studies, straightforward model, and (*QP*, *QD*) pairs used in CTC for joint coding), because encoding with the proposed models leads to total bitrate reduction and improvement of the virtual view quality for MVD sequences compared to other approaches. Also, a new approach was introduced into search for the optimum ratio of view bitrate to total bitrate for stereoscopic video plus depth based on the quantization parameter for video. The experiments have shown that the mean bitrate ratio for views is approximately 60% of the total bitrate for stereoscopic video plus depth. In contrast, this ratio for some MVD sequences may change because of the difference in the content of the views and depth maps required in view synthesis to produce good quality of the synthesized view.

In Chapter 6, the effect of depth map quality on the synthesized view quality has been studied by adding noise to depth maps. Then, the optimum (*QP*, *QD*) pairs (the best rate-distortion curve) for depth maps without noise (unmodified depth maps) were compared to the optimum (*QP*, *QD*) pairs for depth maps with noise. It was noticed that the impact of noise added to the depth maps on virtual view quality is negligible in a strong compression, because the quantization noise produced by compression becomes stronger than the noise added to depth maps before compression at some stage. For weak compression, this effect can be observed because the noise added to the depth maps becomes stronger than the quantization noise. Therefore, a higher quantization parameter must be used to delete this noise added to depth maps.

The encoder model for stereoscopic video plus depth map has been presented in Chapter 7. The encoder model is used to calculate the bitrate or frame size of stereoscopic video plus depth based on the quantization step size for the video. The results of the model have been compared to experimental data, and it was noticed that the difference between experimental and approximate data is minimal, i.e., the accuracy of the model is good. Moreover, many scenarios have been presented to find the relationship between the bitrate or frame size and the quantization step size. It was observed that the model that uses more parameters is more accurate than the model with fewer parameters, but the difference is small.

In Chapter 8, a rate control method for stereoscopic video plus depth map for many codecs has been proposed. The proposed method depends on three relationships: the relationship between the bitrate or frame size of stereoscopic video plus depth and quantization step size for video (the encoder model), the relationship between the quantization step size (*Q*) and the quantization parameter (*QP*), and the relationship between the quantization parameters for video and depth (the bitrate allocation model). The target and approximate values (the

quantization parameter) have been compared to verify the effectiveness of the proposed approach.

9.3 Future Work

The suggestions for future work that can expand this project are as follows:

- Rate control for stereoscopic video plus depth map based on blocks,
- $R-\lambda$ model-based rate control for stereoscopic video plus depth map for several compression techniques. Also, a comparison between the results of the R-Q model and the $R-\lambda$ model will be made to find which model is more accurate in the proposed scheme of rate control for stereoscopic video plus depth map,
- Frame-level rate control for 3D-HEVC and MV-HEVC,
- The *QP-QD* model was derived based on two views plus depth maps in the dissertation. The suggestion is to verify whether these models can be used with more views (3-5 views) plus depth maps.

References

[3DHEVC] 3D-HEVC reference codec available online https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-16.3.

[Abol_19] R. Abolfathi, H. Roodaki, S. Shirmohammadi, "A novel rate control method for free-viewpoint video in MV-HEVC," Int. Conf. Computing, Networking, and Communications (ICNC), Honolulu, HI, USA, Feb. 2019.

[Adel_91] E. Adelson, J. Bergen, "The plenoptic function and the elements of early vision," Computational Models of Visual Processing, Cambridge, MA: MIT Press, pp. 3-20, 1991.

[Akin_15] A. Akin, R. Capoccia, J. Narinx, J. Masur, A. Schmid, Y. Leblebici, "Realtime free viewpoint synthesis using three-camera disparity estimation hardware," IEEE Int. Symposium on Circuits and Systems (ISCAS), Lisbon, Portugal, May 2015.

[Akra_14] S. Akramulla, "Digital video concepts, methods, and metrics: quality, compression, performance, and power trade-off analysis," Apress Media, LLC, 1'st edition, 2014.

[Alex_98] N. Alexandrov, J. Dennis, R. Lewis, V. Torczon, "A trust-region framework for managing the use of approximation models in optimization," Structural Optimization, vol. 15, no.1, pp. 16-23, Feb. 1998.

[Alob_18a] Y. Al-Obaidi, T. Grajek, "Influence of depth map fidelity on virtual view quality," Int. Conf. Signals and Electronic Systems (ICSES), Kraków, Poland, Sep. 2018.

[Alob_18b] Y. Al-Obaidi, T. Grajek, O. Stankiewicz, M. Domański, "Bitrate allocation for multiview video plus depth simulcast coding," Int. Conf. Systems, Signals, and Image Proc. (IWSSIP), Maribor, Slovenia, Jun. 2018.

[Alob_19] Y. Al-Obaidi, T. Grajek, M. Domański, "Quantization of depth in simulcast and multiview coding of stereoscopic video plus depth using HEVC, VVC and MV-HEVC," Picture Coding Symposium (PCS), Ningbo, China, Nov. 2019.

[Alob_20] Y. Al-Obaidi, T. Grajek, "Estimation of the optimum depth quantization parameter for a given bitrate of multiview video plus depth in 3D-HEVC coding," Int. Conf. Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG), Pilsen, Czech Republic, May 2020.

[Anse_10] T. Anselmo, D. Alfonso, "Constant quality variable bitrate control for SVC," 11th Int. Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), Desenzano del Garda, Italy, Apr. 2010.

[AVC] 2D AVC reference codec available online http://iphome.hhi.de/suehring/tml/.

[AVC_std] ISO/IEC 14496-10, MPEG-4 Part 10, Generic coding of audio-visual objects, Advanced Video Coding. Edition 8, 2016, also: ITU-T Rec. H.264 Edition 13.0, 2019.

[AVS_15] AVS working group, "information technology—high efficiency coding and media delivery in heterogeneous environments—part 2: high-efficiency video coding," Doc. AVS-N2150, Mar. 2015.

[Bai_17] L. Bai, L. Song, R. Xie, L. Zhang, Z. Luo, "Rate control model for high dynamic range video," IEEE Visual Communications and Image Proce. (VCIP), St. Petersburg, FL, USA, Dec. 2017.

[Bani_13] A. Banitalebi-Dehkordi, M. Pourazad, P. Nasiopoulos, "A study on the relationship between depth map quality and the overall 3D video quality of experience," 3DTV-Conf.: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON), Aberdeen, Scotland, Oct. 2013.

[Beac_08] A. Beach, "Real world video compression," by Peachpit Press, 1'st edition, 2008.

[Bjon_01] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," ITU-T SG16, Doc. VCEG-M33, Austin, USA, Apr. 2001.

[Bosc_11] E. Bosc, V. Jantet, M. Pressigout, L. Morin, C. Guillemot, "Bit-rate allocation for multi-view video plus depth," 3DTV Conf.: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), Antalya, Turkey, May 2011.

[Bosc_13] E. Bosc, F. Racape, V. Jantet, P. Riou, M. Pressigout, L. Morin, "A study of depth/texture bit-rate allocation in multi-view video plus depth compression," Annals of telecommunications, vol. 68, no. 11-12, pp.615-625, 2013.

[Boss_12] F. Bossen, "Common test conditions and software reference configurations," Joint Collaborative Team on Video Coding (JCT-VC) of ITUT SG16 WP3 and ISO. IEC JTC1/SC29/WG11, Doc. JCTVC-J1100, Stockholm, Sweden, Jul. 2012.

[Boss_19] F. Bossen, J. Boyce, W. Li, V. Seregin, K. Sühring, "JVET common test conditions and software reference configurations for SDR video," Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Doc. JVET-N1010, Geneva, Switzerland, Mar. 2019.

[Bros_20] B. Bross, J. Chen, S. Liu, Y. Wang, "Versatile video coding (Draft 10)," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Doc. JVET-S2001-vH, Jul. 2020.

[Bros_21] B. Bross, J. Chen, J. Ohm, G. Sullivan, Y. Wang, "Developments in international video coding standardization after AVC, with an overview of versatile video coding (VVC)," in Proceedings of the IEEE, Early Access, Jan. 2021.

[Bute_17] B. Butenko, P. Pardalos, V. Shylo, "Optimization methods and applications," Springer International Publishing, 2017.

[Byrd_87] R. Byrd, R. Schnabel, G. Shultz, "A trust region algorithm for nonlinearly constrained optimization," SIAM Journal on Numerical Analysis, vol. 24, no. 5, pp. 1152–1170, Oct. 1987.

[Cai_20] Q. Cai, Z. Chen, D. Wu, B. Huang, "Real-time constant objective quality video coding strategy in high efficiency video coding," IEEE Trans. on Circuits and Systems for Video Technology, vol. 30, no. 7, pp. 2215-2228, Jul. 2020.

[Cao_18] G. Cao, X. Pan, Y. Zhou, Y. Li, Z. Chen, "Two-pass rate control for constant quality in high efficiency video coding," IEEE Visual Communications and Image Proc. (VCIP), Taichung, Taiwan, Dec. 2018.

[Cao_20] H. Cao, X. Shang, H. Qi, "Virtual reality image technology assists training optimization of motion micro-time scale," IEEE Access, vol. 8, pp. 123215-123227, 2020.

[Cela_16] G. Celant, M. Broniatowski, "Interpolation and extrapolation optimal designs 1: Polynomial regression and approximation theory," ISTE Ltd and John Wiley & Sons, Inc., 2016.

[Cen_14] F. Cen, Q. Lu, W. Xu, "Efficient rate control for intra-frame coding in high efficiency video coding," Int. Conf. on Signal Proc. and Multimedia Applications (SIGMAP), Vienna, Austria, Aug. 2014.

[Ceul_18] B. Ceulemans, S. Lu, G. Lafruit, A. Munteanu, "Robust multiview synthesis for wide-baseline camera arrays," IEEE Trans. Multimedia, vol. 20, no. 9, pp. 2235-2248, Sep. 2018.

[Chan_07] S. Chan, H. Shum, K. Ng, "Image-based rendering and synthesis," IEEE Signal Proc. Magazine, vol. 24, no. 6, pp. 22-33, Nov. 2007.

[Chen_14] Y. Chen, M. Hannuksela, T. Suzuki, S. Hattori, "Overview of the MVC+D 3D video coding standard," Journal of Visual Communication and Image Representation, vol. 25, no. 4, pp. 679-688, May 2014.

[Chen_15] Y. Chen, G. Tech, K. Wegner, S. Yea, "Test Model 11 of 3D-HEVC and MV-HEVC," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Doc. JTC3V-H1003, Geneva, Switzerland, Feb. 2015.

[Chen_18] L. Chen, J. Hu, W. Peng, "Reinforcement learning for HEVC/H.265 frame-level bit allocation," IEEE 23rd Int. Conf. Digital Signal Proc. (DSP), Shanghai, China, Nov. 2018.

[Chen_19] L. Chen, C. Hsu, Y. Huang, S. Lei, "AHG17: New NAL unit types for VVC," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Doc. JVET-N0072, Geneva, Switzerland, Mar. 2019.

[Chen_20] Y. Chen, S. Kwong, M. Zhou, S. Wang, G. Zhu, Y. Wang, "Intra frame rate control for Versatile Video Coding with quadratic rate-distortion modelling," IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, May 2020.

[Chen_22] Y. Chen, Q. Yu, "Efficient rate control algorithm for 3-D wavelet-based scalable video coding," IEEE Int. Conf. Computer and Communications (ICCC), Chengdu, China, Dec. 2022.

[Cheng_19] B. Cheng, Y. Zhang, "A neural network approach to GOP-level rate control of x265 using Lookahead," Picture Coding Symposium (PCS), Ningbo, China, Nov. 2019.

[Chia_96] T. Chiang, Y. Zhang, "A new rate control scheme using quadratic rate distortion model," IEEE Int. Conf. Image Proc. (ICIP), Lausanne, Switzerland, Sep. 1996.

[Chia_97] T. Chiang, Y. Zhang, "A new rate control scheme using quadratic rate distortion model," IEEE Trans. on Circuits and Systems for Video Technology, vol. 7, no. 1, pp. 246-250, Feb. 1997.

[Chiu_12] Y. Chiu, W. Zhang, "Cross-check of QP determination by lambda value (JCTVC-I0426)," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Doc. JCTVC-I0573, Geneva, Switzerland, May 2012.

[Choi_13] H. Choi, J. Yoo, J. Nam, D. Sim, I. Bajić, "Pixel-wise unified rate-quantization model for multi-level rate control," IEEE Journal of Selected Topics in Signal Proc., vol. 7, no. 6, pp. 1112-1123, Dec. 2013.

[Choi_20] Y. Choi, H. Lee, S. Chae, "High throughput CBAC hardware encoder with bin merging for AVS 2.0 video coding," IEEE Trans. on Circuits and Systems for Video Technology, Early Access, Dec. 2020.

[Chou_22] R. Choudhary, M. Sharma, A. Wadaskar, "A high resolution multi-exposure stereoscopic image & video database of natural scenes," arXiv preprint arXiv:2206.11095, Jun. 2022.

[Cisc_18] Cisco, "Cisco visual networking index: forecast and methodology, 2017–2022," updated December, 2018.

[Conn_00] A. Conn, N. Gould, P. Toint, "Trust-region methods," Society for Industrial and Applied Mathematics and Mathematical Programming Society, 2000.

[Cord_16]. M. Cordina, C. Debono, "A depth map rate control algorithm for HEVC multi-view video plus depth," IEEE Int. Conf. Multimedia & Expo Workshops (ICMEW), Seattle, WA, USA, Jul. 2016.

[Deng_23] X. Deng, Y. Deng, R. Yang, W. Yang, R. Timofte, M. Xu, "MASIC: deep mask stereo image compression," IEEE Trans. Circuits and Systems for Video Technology, Early Access, Mar. 2023.

[Do_11] L. Do, G. Bravo, S. Zinger, P. de With, "Real-time free-viewpoint DIBR on GPUs for large base-line multi-view 3DTV videos," Visual Communications and Image Processing (VCIP), Tainan, Taiwan, Nov. 2011.

[Doma_09] M. Domański, T. Grajek, K. Klimaszewski, M. Kurc, O. Stankiewicz, J. Stankowski, K. Wegner, "Poznań multiview video test sequences and camera parameters," ISO/IEC JTC1/SC29/WG11, Doc. MPEG/M17050, Xi'an, China, Oct. 2009.

[Doma_13] M. Domański, O. Stankiewicz, K. Wegner, M. Kurc, J. Konieczny, J. Siast, J. Stankowski, R. Ratajczak, T. Grajek, "High efficiency 3D video coding using new tools based on view synthesis," IEEE Trans. on Image Proc., vol. 22, no. 9, pp. 3517-3527, Sep. 2013.

[Doma_14] M. Domański, A. Dziembowski, A. Kuehn, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, "Experiments on acquisition and processing of video for free-viewpoint television," 3DTV-Conf.: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), Budapest, Hungary, Jul. 2014.

[Doma_15] M. Domański, A. Dziembowski, D. Mieloch, A. Łuczak, O. Stankiewicz, K. Wegner, "A practical approach to acquisition and processing of free viewpoint video," Picture Coding Symposium (PCS), Cairns, QLD, Australia, Jun. 2015.

[Doma_16a] M. Domański, M. Bartkowiak, A. Dziembowski, T. Grajek, A. Grzelka, A. Łuczak, D. Mieloch, J. Samelak, O. Stankiewicz, J. Stankowski, K. Wegner, "New results in free-viewpoint television systems for horizontal virtual navigation," IEEE Int. Conf. Multimedia and Expo (ICME), Seattle, WA, USA, Jul. 2016.

[Doma_16b] M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, O. Stankiewicz, K. Wegner, "Multiview test video sequences for free navigation exploration obtained using pairs of cameras," ISO/IEC JTC1/SC29/WG11, Doc. MPEG M38247, Geneva, Switzerland, May 2016.

[Doma_16c] M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, "Optimization of camera positions for free-navigation applications," Int. Conf. on Signals and Electronic Systems (ICSES), Kraków, Poland, Sep. 2016.

[Doma_16d] M. Domański, A. Dziembowski, A. Grzelka, Ł. Kowalski, D. Mieloch, J. Samelak, O. Stankiewicz, J. Stankowski, K. Wegner, "[FTV AHG] Extended results of Poznan University of Technology proposal for call for evidence on free-viewpoint television," ISO/IEC JTC1/SC29/WG11, Doc. MPEG2016/ M38246, Geneva, Switzerland, Jun. 2016.

[Doma_17] M. Domański, O. Stankiewicz, K. Wegner, T. Grajek, "Immersive visual media — MPEG-I: 360 video, virtual navigation and beyond," Int. Conf. on Systems, Signals and Image Proc. (IWSSIP), Poznań, Poland, May 2017.

[Doma_18] M. Domański, D. Łosiewicz, T. Grajek, O. Stankiewicz, K. Wegner, A. Dziembowski, D. Mieloch, "Extended VSRS for 360 degree video," ISO/IEC JTC1/SC29/WG11, Doc. m41990, Gwangju, South Korea, Jan. 2018.

[Doma_21] M. Domański, Y. Al-Obaidi, T. Grajek, "Universal modeling of monoscopic and multiview video codecs with applications to encoder control," IEEE Int. Conf. Image Proc. (ICIP), Anchorage, Alaska, USA, Sep. 2021.

[Dong_07a] J. Dong, N. Ling, "On model parameter estimation for H.264/AVC rate control," IEEE Int. Symposium on Circuits and Systems, New Orleans, LA, USA, May 2007.

[Dong_07b] J. Dong, N. Ling, "Enhanced linear R-Q model based rate control for H.264/AVC using context-adaptive parameter estimation," IEEE Workshop on Signal Proc. Systems, Shanghai, China, Oct. 2007.

[Dong_07c] J. Dong, N. Ling, "Enhanced linear R-Q model based rate control for H.264/AVC using context-adaptive parameter estimation," IEEE Workshop on Signal Proc. Systems, Shanghai, China, Oct. 2007.

[Dong_09] J. Dong, N. Ling, "A context-adaptive prediction scheme for parameter estimation in H.264/AVC macroblock layer rate control," IEEE Trans. Circuits and Systems for Video Technology, vol. 19, no. 8, pp. 1108-1117, Aug. 2009.

[Du_19] Q. Du, R. Liu, Y. Pan, S. Sun, S. Sun, Z. Zheng, "Depth estimation with multi-resolution stereo matching," IEEE Visual Communications and Image Proc. (VCIP), Sydney, Australia, Dec. 2019.

[Dzie_16] A. Dziembowski, A. Grzelka, D. Mieloch, O. Stankiewicz, K. Wegner, M. Domański, "Multiview synthesis – improved view synthesis for virtual navigation," 32nd Picture Coding Symposium (PCS), Nuremberg, Germany, Dec. 2016.

[Dzie_18a] A. Dziembowski, J. Samelak, M. Domański, "View selection for virtual view synthesis in free navigation systems," Int. Conf. on Signals and Electronic Systems (ICSES), Kraków, Poland, Sep. 2018.

[Dzie_18b] A. Dziembowski, J. Stankowski, "Real-time CPU-based virtual view synthesis," Int. Conf. on Signals and Electronic Systems (ICSES), Kraków, Poland, Sep. 2018.

[Dzie_19] A. Dziembowski, M. Domański, "[MPEG-I Visual] Objective quality metric for immersive video," ISO/IEC JTC1/SC29/WG11, Doc. MPEG/M48093, Gothenburg, Sweden, Jul. 2019.

[Dzie_20] A. Dziembowski, "Software manual of IV-PSNR for immersive video," ISO/IEC JTC 1/SC 29/WG 11, Doc. MPEG/N18709, Göteborg, Sweden, Jul. 2019.

[Dzie_22] A. Dziembowski, D. Mieloch, J. Stankowski, A. Grzelka, "IV-PSNR – the objective quality metric for immersive video applications," IEEE Trans. on Circuits and Systems for Video Technology, pp. 1-16, June 2022.

[Fach_18] S. Fachada, D. Bonatto, A. Schenkel, G. Lafruit, "Depth image based view synthesis with multiple reference views for virtual reality," 3DTV-Conf.: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), Helsinki, Finland, Jun. 2018.

[Fari_15] M. Farid, M. Lucenteforte, M. Grangetto, "Objective quality metric for 3D virtual views," IEEE Int. Conf. Image Proc. (ICIP), Quebec City, QC, Canada, Sep. 2015.

[Fehn_04] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," Proc. SPIE, vol. 5291, pp. 93-104, May 2004.

[Fili_19] A. Filippov, V. Rufitskiy, "Recent advances in intra prediction for the emerging H.266/VVC video coding standard," Int. Conf. on Engineering, Computer and Information Sciences (SIBIRCON), Novosibirsk, Russia, Oct. 2019.

[Flie_02] M. Flierl, T. Wiegand, B. Girod, "Rate-constrained multihypothesis prediction for motion-compensated video compression," IEEE Trans. on Circuits and Systems for Video Technology, vol. 12, no. 11, pp. 957-969, Nov. 2002.

[Fran_22] A. Franzluebbers, C. Li, A. Paterson, K. Johnsen, "Virtual reality point cloud annotation," IEEE Conf. Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW), Christchurch, New Zealand, Mar. 2022.

[Fuji_94] T. Fujii, "The basic study on the integrated 3-D visual communication," Ph.D dissertation, The University of Tokyo, Japan, 1994.

[Fuxi_23] C. Fuxiong, "A novel virtual reality (VR) based intelligent guiding system," Int. Conf. Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, Jan. 2023.

[Gao_14] W. Gao, S. Ma, "Advanced video coding systems," by Springer, Inc., 2014.

[Gao_19] W. Gao, S. Kwong, Q. Jiang, C. Fong, P. Wong, W. Yuen, "Data-driven rate control for rate-distortion optimization in HEVC based on simplified effective initial QP learning," IEEE Trans. on Broadcasting, vol. 65, no. 1, pp. 94-108, Mar. 2019.

[Ghaz_19] R. Ghaznavi-Youvalari, A. Aminlou, J. Lainema, "Regression-based motion vector field for video coding," IEEE Trans. on Circuits and Systems for Video Technology, Early Access, Sep. 2019.

[Gong_19] Y. Gong, S. Wan, K. Yang, H. Wu, Y. Liu, "Temporal-layer-motivated lambda domain picture level rate control for random-access configuration in H.265/HEVC," IEEE Trans. on Circuits and Systems for Video Technology, vol. 29, no. 1, pp. 156-170, Jan. 2019.

[Graj_10] T. Grajek, M. Domański, "New model of MPEG-4 AVC/H.264 video encoders," IEEE Int. Conf. Image Proc. (ICIP), Hong Kong, China, Sep. 2010.

[Graz_20] D. Graziosi, O. Nakagami, S. Kuma, A. Zaghetto, T. Suzuki, A. Tabatabai, "An overview of ongoing point cloud compression standardization activities: video-based (V-PCC) and geometry-based (G-PCC)," APSIPA Trans. on Signal and Information Processing, vol. 9, no.13, pp 1-17, 2020.

[Groi_11] D. Grois, O. Hadar, "Complexity-aware adaptive bit-rate control with dynamic ROI pre-processing for scalable video coding," IEEE Int. Conf. on Multimedia and Expo, Barcelona, Spain, Jul. 2011.

[Guo_15] Y. Guo, B. Li, S. Sun, J. Xu, "Rate control for screen content coding in HEVC," IEEE Int. Symposium on Circuits and Systems (ISCAS), Lisbon, Portugal, May 2015.

[Guo_17] H. Guo, C. Zhu, Y. Gao, S. Song, "A frame-level rate control scheme for low delay video coding in HEVC," IEEE 19th Int. Workshop on Multimedia Signal Proc. (MMSP), Luton, UK, Oct. 2017.

[Guo_19] H. Guo, C. Zhu, S. Li, Y. Gao, "Optimal bit allocation at frame level for rate control in HEVC," IEEE Trans. Broadcasting, vol. 65, no. 2, pp. 270-281, Jun. 2019.

[Guo_20] H. Guo, C. Zhu, M. Xu, S. Li, "Inter-block dependency-based CTU level rate control for HEVC," IEEE Trans. Broadcasting, vol. 66, no. 1, pp. 113-126, Mar. 2020.

[Guot_20] T. Guo, J. Wang, Z. Cui, Y. Feng, Y. Ge, B. Bai, "Variable rate image compression with content adaptive optimization," IEEE/CVF Conf. Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, Jun. 2020.

[Hami_22] W. Hamidouche, T. Biatek, M. Abdoli, E. François, F. Pescador, M. Radosavljević, D. Menard, M. Raulet, "Versatile video coding standard: A review from coding tools to consumers deployment," IEEE Consumer Electronics Magazine, vol. 11, no. 5, pp. 10-24, Sep. 2022.

[Hann_13] M. Hannuksela et al., "Multiview-video-plus-depth coding based on the advanced video coding standard," IEEE Trans. on Image Proc., vol. 22, no. 9, pp. 3449-3458, Sep. 2013.

[Haro_10] T. Haroun, F. Labeau, "Robust multiple hypothesis motion compensated prediction within the H.264/AVC standard," 2nd Int. Conf. on Image Proc. Theory, Tools and Applications, Paris, France, Jul. 2010.

[He_02] Z. He, S. Mitra, "Optimum bit allocation and accurate rate control for video coding via ρ-domain source modeling," IEEE Trans. Circuits and Systems for Video Technology, vol. 12, no. 10, pp. 840-849, Oct. 2002.

[He_08] Z. He, D. Wu, "Linear rate control and optimum statistical multiplexing for H.264 video broadcast," IEEE Trans. on Multimedia, vol. 10, no. 7, pp. 1237-1249, Nov. 2008.

[Heid_19] O. Heidari, A. Perez-Gracia, "Virtual reality synthesis of robotic systems for human upper-limb and hand tasks," IEEE Conf. Virtual Reality and 3D User Interfaces (VR), Osaka, Japan, Mar. 2019.

[HEVC] 2D HEVC reference codec available online https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.18.

[HEVC_std] ISO/IEC 23008-2, MPEG-H Part 2, High-Efficiency Video Coding, Edition 4, 2020, also: ITU-T Rec. H.265 Edition 7.0, 2019.

[Hou_10] J. Hou, S. Wan, F. Yang, "Frame rate adaptive rate model for video rate control," Int. Conf. Multimedia Communications, Hong Kong, China, Aug. 2010.

[Hsie_20] J. Hsieh, J. Syu, Z. Zhang, "Coding efficient motion estimation rate control for H.265/HEVC," IEEE 9th Global Conf. on Consumer Electronics (GCCE), Kobe, Japan, Oct. 2020.

[Hu_10] S. Hu, H. Wang, S. Kwong, T. Zhao, "Frame level rate control for H.264/AVC with novel rate-quantization model," IEEE Int. Conf. Multimedia and Expo (ICME), Suntec City, Singapore, Jul. 2010.

[Hu_12] H. Hu, B. Li, W. Lin, W. Li, M. Sun, "Region-based rate control for H.264/AVC for low bit-rate applications," IEEE Trans. on Circuits and Systems for Video Technology, vol. 22, no. 11, pp. 1564-1576, Nov. 2012.

[Hu_13] S. Hu, S. Kwong, Y. Zhang, C. Kuo, "Rate-distortion optimized rate control for depth map-based 3-D video coding," IEEE Trans. Image Proc., vol. 22, no. 2, pp. 585-594, Feb. 2013.

[Huv_23] T. Huu, V. Duong, J. Yim, B. Jeon, "Ray-space motion compensation for lenslet plenoptic video coding," IEEE Trans. Image Proc., vol. 32, pp. 1215-1230, Feb. 2023.

[Hyun_20] M. Hyun, B. Lee, M. Kim, "A frame-level constant bit-rate control using recursive bayesian estimation for versatile video coding," IEEE Access, vol. 8, pp. 227255-227269, 2020.

[Jain_89] A. Jain, "Fundamentals of digital image processing," Prentice-Hall, USA, 1989.

[JCT] JCT-VC, ftp://hevc@ftp.tnt.uni-hannover.de/testsequences.

[Jeon_19] J. Jeong, S. Lee, D. Jang, E. Ryu, "Towards 3DoF+ 360 video streaming system for immersive media," IEEE Access, vol. 7, pp. 136399-136408, 2019.

[Jin_16a] J. Jin, A. Wang, Y. Zhao, C. Lin, B. Zeng, "Region-aware 3D warping for DIBR," IEEE Trans. Multimedia, vol. 18, no. 6, pp. 953-966, Jun. 2016.

[Jin_16b] Z. Jin, T. Tillo, J. Xiao, Y. Zhao, "Multiview video plus depth transmission via virtual-view-assisted complementary down/upsampling," EURASIP Journal Image Video Process., vol. 2016, no. 19, pp. 1-17, 2016.

[Jin_22] D. Jin, J. Lei, B. Peng, W. Li, N. Ling, Q. Huang, "Deep affine motion compensation network for inter prediction in VVC," IEEE Trans. Circuits and Systems for Video Technology, vol. 32, no. 6, pp. 3923-3933, Jun. 2022.

[Jin_23] X. Jin, L. Wu, G. Shen, Y. Chen, J. Chen, J. Koo, C. Hahm, "Enhanced bi-directional motion estimation for video frame interpolation," IEEE/CVF Winter Conf. on Applications of Computer Vision (WACV), Waikoloa, HI, USA, Jan. 2023.

[Joshi_15] M. Joshi, M. Raval, Y. Dandawate, K. Joshi, S. Metkar, "Image and video compression fundamentals, techniques, and applications," Taylor & Francis Group, LLC, 1'st edition, 2015.

[Jun_15] H. Jun-Te, L. Jin-Jang, "Virtual view synthesis for multi-view video plus depth sequences using spatial-temporal information," 3DTV-Conf.: The True Vision – Capture, Transmission and Display of 3D Video (3DTV-CON), Lisbon, Portugal, Jul. 2015.

[Jung_17] A. Jung, "Comparison of video quality assessment methods," Master dissertation, Blekinge Institute of Technology, Sweden, 2017.

[Jung_20] J. Jung, V. Vadakital, D. Mieloch, "Report of the exploration experiments on future MPEG immersive video," ISO/IEC JTC 1/SC 29/WG4, Doc. m55712, Oct. 2020.

[Jung_22] K. Jung, B. Sohn, "Immersive body-hand gesture interfaces for HMD-based virtual environment navigation," Int. Conf. Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, Oct. 2022,

[Kian_20] D. Kianfar, A. Wiggers, A. Said, R. Pourreza, T. Cohen, "Parallelized rate-distortion optimized quantization using deep learning," IEEE 22nd Int. Workshop on Multimedia Signal Proc. (MMSP), Tampere, Finland, Sep. 2020.

[Kim_01] Y. Kim, Z. He, S. Mitra, "A novel linear source model and a unified rate control algorithm for H.263/MPEG-2/MPEG-4," Int. Conf. on Acoustics, Speech, and Signal Proc. (ICASSP), Salt Lake City, UT, USA, May 2001.

[Kim_13] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, M. Gross, "Scene reconstruction from high spatio-angular resolution light fields," ACM Trans. on Graphics, vol. 32, no. 4, pp. 73:1-73:11, Jul. 2013.

[Kim_23] S. Kim, Y. Lee, K. Yoon, "Versatile video coding-based coding tree unit level image compression with dual quantization parameters for hybrid vision," IEEE Access, vol. 11, pp. 34498-34509, 2023.

[Klim_14a] K. Klimaszewski, K. Wegner, M. Domański, "Video and depth bitrate allocation in multiview compression," 21st Int. Conf. Systems, Signals and Image Proc. (IWSSIP), Dubrovnik, Croatia, May 2014.

[Klim_14b] K. Klimaszewski, O. Stankiewicz, K. Wegner, M. Domański, "Quantization optimization in multiview plus depth video coding," IEEE Int. Conf. Image Proc. (ICIP), Paris, France, Oct. 2014.

[Koni_12] J. Konieczny, "Motion information coding for 3D video compression," Ph.D. dissertation, Poznan University of Technology, Poland, 2012.

[Kova_15] P. Kovacs, "[FTV AHG] Big Buck Bunny light-field test sequences," ISO/IEC JTC1/SC29/WG11, Doc. MPEG M35721, Geneva, Switzerland, Feb. 2015.

[Kubo_07] A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, C. Zhang, "Multiview imaging and 3DTV," IEEE Signal Proc. Magazine, vol. 24, no. 6, pp. 10-21, 2007.

[Kune_20] A. Kunert, T. Weissker, B. Froehlich, A. Kulik, "Multi-window 3D interaction for collaborative virtual reality," IEEE Trans. on Visualization and Computer Graphics, vol. 26, no. 11, pp. 3271-3284, Nov. 2020.

[Lafr_16] G. Lafruit, M. Domański, K. Wegner, T. Grajek, T. Senoh, J. Jung, P. Kovács, P. Goorts, L. Jorissen, A. Munteanu, B. Ceulemans, P. Carballeira, S. García, M. Tanimoto, "New visual coding exploration in MPEG: super-multiview and Free navigation in free viewpoint TV," IST Electronic Imaging, Stereoscopic Displays and Applications XXVII, San Francisco, USA, Feb. 2016.

[Lee_11] J. Lee, J. Lee, D. Park, "Effect of synthesized depth view on multi-view rendering quality," 18th IEEE Int. Conf. Image Proc. (ICIP), Brussels, Belgium, Sep. 2011.

[Lee_12] H. Lee, S. Sull, "A VBR video encoding for locally consistent picture quality with small buffering delay under limited bandwidth," IEEE Trans. on Broadcasting, vol. 58, no. 1, pp. 47-56, Mar. 2012.

[Lee_13] A. Lee, D. Jun, J. Kim, J. Seok, Y. Kim, S. Jung, J. Choi, "An efficient inter prediction mode decision method for fast motion estimation in HEVC," Int. Conf. on ICT Convergence (ICTC), Jeju, South Korea, Oct. 2013.

[Lee_15] C. Lee, A. Tabatabai, K. Tashiro, "Free viewpoint video (FVV) survey and future research direction," APSIPA Trans. Signal and Information Proc., vol. 4, no. 15, pp. 1-10, Oct. 2015.

[Lee_98] J. Lee, Y. Ho, "Target bit matching for MPEG-2 video rate control," Int. Conf. on Global Connectivity in Energy, Computer, Communication and Control, New Delhi, India, Dec. 1998.

[Lei_18] J. Lei, X. He, H. Yuan, F. Wu, N. Ling, C. Hou, "Region adaptive R- λ model-based rate control for depth maps coding," IEEE Trans. Circuits and Systems for Video Technology, vol. 28, no. 6, pp. 1390-1405, Jun. 2018.

[Levo_06] M. Levoy, "Light fields and computational imaging," IEEE Computer, vol. 39, no. 8, pp. 46-55, Aug. 2006.

[Levo_96] M. Levoy, P. Hanrahan, "Light field rendering," Proc. ACM SIGGRAPH, Jul. 1996.

[Li_03] Z. Li., F. Pan, K. Lim, G. Feng, X. Lin, S. Rahardja, "Adaptive basic unit layer rate control for JVT," Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, Doc. JVT-G012, Pattaya, Thailand, Mar. 2003.

[Li_12] B. Li, D. Zhang, H. Li, J. Xu, "QP determination by lambda value," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Doc. JCTVC-I0426, Geneva, Switzerland, May 2012.

[Li_13] D. Li, H. Hang, Y. Liu, "Virtual view synthesis using backward depth warping algorithm," Picture Coding Symposium (PCS), San Jose, CA, USA, Dec. 2013.

[Li_14a] Y. Li, H. Jia, P. Ma, C. Zhu, X. Xie, W. Gao, "Inter-dependent rate-distortion modeling for video coding and its application to rate control," IEEE Int. Conf. Multimedia and Expo (ICME), Chengdu, China, Jul. 2014.

[Li_14b] B. Li, H. Li, L. Li, J. Zhang, "λ domain rate control algorithm for high efficiency video coding," IEEE Trans. Image Proc., vol. 23, no. 9, pp. 3841-3854, Sep. 2014.

[Li_17] Y. Li, B. Li, D. Liu, Z. Chen, "A convolutional neural network-based approach to rate control in HEVC intra coding," IEEE Visual Communications and Image Proc. (VCIP), St. Petersburg, FL, USA, Dec. 2017.

[Li_18a] S. Li, C. Zhu, M. Sun, "Hole filling with multiple reference views in DIBR view synthesis," IEEE Trans. on Multimedia, vol. 20, no. 8, pp. 1948-1959, Aug. 2018.

[Li_18b] W. Li, F. Zhao, E. Zhang, P. Ren, "Optimal frame-level bit allocation in HEVC with distortion dependency model," IEEE Int. Conf. on Big Data and Smart Computing (BigComp), Shanghai, China, Jan. 2018.

[Li_19]. T. Li, L. Yu, S. Yu, Y. Chen, "The bit allocation method based on inter-view dependency for multi-View texture video coding," Data Compression Conf. (DCC), Snowbird, UT, USA, Mar. 2019.

[Li_20a] T. Li, L. Yu, H. Wang, Z. Kuang, "A bit allocation method based on inter-view dependency and spatio-temporal correlation for multi-view texture video coding," IEEE Trans. on Broadcasting, Early Access, Oct. 2020.

[Li_20b] L. Li, N. Yan, Z. Li, S. Liu, H. Li, "λ-Domain perceptual rate control for 360-degree video compression," IEEE Journal of Selected Topics in Signal Proc., vol. 14, no. 1, pp. 130-145, Jan. 2020.

[Li_20c] Y. Li, Z. Liu, Z. Chen, S. Liu, "Rate control for versatile video coding," IEEE Int. Conf. Image Proc. (ICIP), Abu Dhabi, United Arab Emirates, Oct. 2020.

[Li_22] T. Li, M. Slavcheva, M. Zollhoefer, S. Green, C. Lassner, C. Kim, T. Schmidt, S. Lovegrove, M. Goesele, R. Newcombe, Z. Lv, "Neural 3D video synthesis from multi-view video," IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, Jun. 2022.

[Lian_13] X. Liang, Q. Wang, Y. Zhou, B. Luo, A. Men, "A novel R-Q model based rate control scheme in HEVC," Visual Communications and Image Proc. (VCIP), Kuching, Malaysia, Nov. 2013.

[Lie_14] W. Lie, Y. Liao, "Rate control technique based on 3D quality optimization for 3D video encoding," IEEE Int. Conf. Image Proc. (ICIP), Paris, France, Oct. 2014.

[Likh_22] S. Likhitha, R. Baskar, "Performance analysis of median filter in comparison with gabor filter for skin cancer MRI images based on PSNR and MSE," Int. Conf. System Modeling & Advancement in Research Trends (SMART), Moradabad, India, Dec. 2022.

[Lim_07] S. Lim, H. Na, Y. Lee, "Rate control based on linear regression for H.264/MPEG-4 AVC," Signal Processing: Image Communication, vol. 22, no. 1, pp. 39-58 Jan. 2007.

[Lin_08] G. Lin, S. Zheng, J. Hu, "A two-stage ρ-domain rate control scheme for H.264 encoder," IEEE Int. Conf. Multimedia and Expo (ICME), Hannover, Germany, Apr. 2008.

[Lin_14] Y. Lin, J. Wu, "Quality assessment of stereoscopic 3D image compression by binocular integration behaviors," IEEE Trans. on Image Proc., vol. 23, no. 4, pp. 1527-1542, Apr. 2014.

[Lin_22] Y. Lin, C. Chen, C. Tang, "VVC based rate control using SKIP CTU predictor," IEEE Int. Conf. Consumer Electronics-Asia (ICCE-Asia), Yeosu, Republic of Korea, Oct. 2022.

[Lins_01] L. Linsen, "Point cloud representation." 2001.

[Liu_11] Y. Liu, Q. Huang, S. Ma, D. Zhao, W. Gao, S. Ci, H. Tang, "A novel rate control technique for multiview video plus depth based 3D video coding," IEEE Trans. Broadcasting, vol. 57, no. 2, pp. 562-571, Jun. 2011.

[Liu_14] H. Liu, G. Wang, "CBR-based resource allocation for multimedia streaming services in QoS enhanced LTE/LTE-A networks," 7th Int. Symposium on Circuits and Systems (ISCAS), Bangkok, Thailand, May 2014.

[Liu_15] Z. Liu, X. Jin, C. Li, Q. Dai, "A workload balanced parallel view synthesis for FTV," IEEE Int. Conf. on Acoustics, Speech and Signal Proc. (ICASSP), Brisbane, QLD, Australia, Apr. 2015.

[Liu_21] Z. Liu, X. Pan, Y. Li, Z. Chen, "A game theory based CTU-level bit allocation scheme for HEVC region of interest coding," IEEE Trans. on Image Proc., vol. 30, pp. 794-805, 2021.

[Liu_22] H. Liu, S. Zhu, B. Zeng, "Inter-frame dependency-based rate control for VVC low-delay coding," IEEE Signal Proc. Letters, vol. 29, pp. 2727-2731, 2022.

[Lu_14] Q. Lu, F. Cen, W. Xu, "Efficient frame complexity-based rate control for H.264/AVC intra-frame," IEEE Int. Conf. on System Science and Engineering (ICSSE), Shanghai, China, Jul. 2014.

[Luca_13] L. Lucas, N. Rodrigues, C. Pagliari, E. Silva, S. Farla, "Predictive depth map coding for efficient virtual view synthesis," IEEE Int. Conf. Image Proc. (ICIP), Melbourne, VIC, Australia, Sep. 2013.

[Luo_05] N. Luo, L. Chen, X. Fang, "Implementation of video encoder with real-time constant- and variable-bit-rate control," IEEE Int. Symposium on Communications and Information Technology (ISCIT), Beijing, China, Oct. 2005.

[Mans_20] I. Mansri, N. Doghmane, N. Kouadria, S. Harize, A. Bekhouch, "Comparative evaluation of VVC, HEVC, H.264, AV1, and VP9 encoders for low-delay video applications," Fourth Int. Conf. on Multimedia Computing, Networking and Applications (MCNA), Valencia, Spain, Oct. 2020.

[Mao_22] Y. Mao, M. Wang, S. Wang, S. Kwong, "High efficiency rate control for versatile video coding based on composite cauchy distribution," IEEE Trans. Circuits and Systems for Video Technology, vol. 32, no. 4, pp. 2371-2384, Apr. 2022.

[Math] Mathworks, https://www.mathworks.com/products/matlab.html.

[Maun_16] H. Maung, S. Aramvith, Y. Miyanaga, "Improved region-of-interest based rate control for error resilient HEVC framework," IEEE Int. Conf. Digital Signal Proc. (DSP), Beijing, China, Oct. 2016.

[Mcmi_95] L. McMillan, G. Bishop, "Plenoptic modeling: An image-based rendering system," The annual conference on Computer graphics and interactive techniques, Los Angeles, California, USA, Aug. 1995.

[Medd_14] M. Meddeb, M. Cagnazzo, B. Pesquet-Popescu, "Region-of-interest based rate control scheme for high efficiency video coding," IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, May 2014.

[Medd_15] M. Meddeb, M. Cagnazzo, B. Pesquet-Popescu, "ROI-based rate control using tiles for an HEVC encoded video stream over a lossy network," IEEE Int. Conf. Image Proc. (ICIP), Quebec City, Canada, Sep. 2015.

[Merk_07] P. Merkle, A. Smolic, K. Müller, T. Wiegand, "Multi-view video plus depth representation and coding," IEEE Int. Conf. Image Proc. (ICIP), San Antonio, TX, USA, Sep. 2007.

[Merk_22] P. Merkle, M. Winken, J. Pfaff, H. Schwarz, D. Marpe, T. Wiegand, "Intra-inter prediction for versatile video coding using a residual convolutional neural network," IEEE Int. Conf. Image Proc. (ICIP), Bordeaux, France, Oct. 2022.

[Mess_15] A. Messac, "Optimization in practical with MATLAB," Cambridge University Press, 2015.

[Meue_20] H. Meuel, J. Ostermann, "Analysis of affine motion-compensated prediction in video coding," IEEE Trans. on Image Proc., vol. 29, pp. 7359-7374, 2020.

[Miel_18a] D. Mieloch, A. Grzelka, "Segmentation-based method of increasing the depth maps temporal consistency," Int. Journal of Electronics and Telecommunications, vol. 64, no. 3, pp. 293–298, 2018.

[Miel_18b] D. Mieloch, "Depth estimation in free-viewpoint television," Ph.D. dissertation, Poznan University of Technology, Poland, 2018.

[Miel_20] D. Mieloch, O. Stankiewicz, M. Domański, "Depth map estimation for free-viewpoint television and virtual navigation," IEEE Access, vol. 8, pp. 5760-5776, 2020.

[Mone_13] M. Moness, A. Mostafa, "An algorithm for parameter estimation of twin-rotor multi-input multi-output system using trust region optimization methods," Journal of Systems and Control Engineering, vol. 227, no. 5, pp. 435–450, 2013.

[Morv_07] Y. Morvan, D. Farin, P. With, "Joint depth/texture bit allocation for multi-view video compression," Picture Coding Symposium (PCS), Lisbon, Portugal, Jan. 2007.

[MPEG_20] MPEG, "Potential improvements of MIV," ISO/IEC JTC1/SC29/WG4, Doc. N00004, Oct. 2020.

[Mull_09] K. Muller, A. Smolic, K. Dix, P. Merkle, T. Wiegand, "Coding and intermediate view synthesis of multiview video plus depth," 16th IEEE Int. Conf. Image Proc. (ICIP), Cairo, Egypt, Nov. 2009.

[Mull_11] K. Muller, P. Merkle, T. Wiegand, "3-D video representation using depth maps," Proc. IEEE, vol. 99, no. 4, pp. 643-656, Apr. 2011.

[Mull_14] K. Müller, A. Vetro, "Common test conditions of 3DV core experiments," ISO/IEC JTC1 SC29/WG11 and ITU-T SG 16 WP 3, Doc. JCT3V G1100, San José, USA, Jan. 2014.

[Muno_23] M. Muñoz, D. Maass, M. Perleberg, L. Agostini, M. Porto, "Hardware design for the affine motion compensation of the VVC standard," IEEE Latin America Symposium on Circuits and Systems (LASCAS), Quito, Ecuador, Mar. 2023.

[MVHEVC] MV-HEVC reference codec available online https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-16.3.

[Nam_22] D. Nam, W. Jung, H. Han, J. Han, "An efficient algorithm to select reference views for virtual view synthesis," IEEE Access, vol. 10, pp. 62306-62321, 2022.

[Nata_11] M. Natali, S. Biasotti, G. Patanè, B. Falcidieno, "Graph-based representations of point clouds," Graphical Models, vol. 73, no. 5, pp. 151-164, Sep. 2011.

[Noor_13] N. Noor, H. Karim, N. Arif, A. Sali, "Multiview plus depth video using high-efficiency video coding method," IEEE Int. Conf. on Signal and Image Processing Applications, Melaka, Malaysia, Oct. 2013.

[Nur_10] G. Nur, S. Dogan, H. Arachchi, A. Kondoz, "Impact of depth map spatial resolution on 3D video quality and depth perception," 3DTV-Conf.: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), Tampere, Finland, Jun. 2010.

[Oh_14] B. Oh, K. Oh, "View synthesis distortion estimation for AVC- and HEVC-compatible 3-D video coding," IEEE Trans. on Circuits and Systems for Video Technology, vol. 24, no. 6, pp. 1006-1015, Jun. 2014.

[Ozak_07] H. Ozaktas, L. Onural, "Three-dimensional television: capture, transmission, and display," Springer, Heidelberg, Germany, Nov. 2007.

[Papa_11] A. Papadakis, K. Zachos, "Subjective and objective video codec evaluation," 15th Panhellenic Conf. on Informatics, Kastonia, Greece, Oct. 2011.

[Pate_15] D. Pate, T. Lad, D. Shah, "Review on intra-prediction in high efficiency video coding (HEVC) standard," Int. Journal of Computer Applications, vol. 132, no.13, pp. 27-30, Dec. 2015.

[Pino_14] P. Piñol, O. López, M. Malumbres, J. Oliver, C. Calafate, "On the performance of video quality assessment metrics under different compression and packet loss scenarios," Hindawi Publishing Corporation, The Scientific World Journal, vol. 2014, Article ID. 743604, pp. 1-18, May 2014.

[Puri_15] A. Purica, M. Cagnazzo, B. Pesquet-Popescu, F. Dufaux, B. Ionescu, "A distortion evaluation framework in 3D video view synthesis," Int. Conf. 3D Imaging (IC3D), Liege, Belgium, Dec. 2015.

[Puri_16] A. Purica, E. Mora, B. Pesquet-Popescu, M. Cagnazzo, B. Ionescu, "Multiview plus depth video coding with temporal prediction view synthesis," IEEE Trans. Circuits and Systems for Video Technology, vol. 26, no. 2, pp. 360-374, Feb. 2016.

[Qin_19] J. Qin, H. Bai, Y. Zhao, "Rate control algorithm in HEVC based on scene-change detection," Data Compression Conf. (DCC), Snowbird, UT, USA, Mar. 2019.

[Raha_19] D. Rahaman, "View synthesis for free-viewpoint video using temporal modeling," Ph.D. dissertation, Charles Sturt University, Australia, Apr. 2019.

[Rama_17] A. Ramanand, I. Ahmad, V. Swaminathan, "A survey of rate control in HEVC and SHVC video encoding," IEEE Int. Conf. Multimedia & Expo Workshops (ICMEW), Hong Kong, China, Jul. 2017.

[Rana_10] P. Rana, M. Flierl, "Depth consistency testing for improved view interpolation," IEEE Int. Workshop on Multimedia Signal Proc., Saint-Malo, France, Oct. 2010.

[Rao_14] K. Rao, D. Kim, J. Hwang, "Video coding standards AVS China, H.264/MPEG-4 part 10, HEVC, VP6, DIRAC and VC-1," by Springer, Inc., 2014.

[Riba_99] J. Ribas-Corbera, S. Lei, "Rate control in DCT video coding for low-delay communications," IEEE Trans. on Circuits and Systems for Video Technology, vol. 9, no. 1, pp. 172-185, Feb. 1999.

[Rich_03a] I. Richardson, "H.264 and MPEG-4 video compression video coding for next-generation multimedia," by John Wiley & Sons, Inc., 2003.

[Rich_03b] I. Richardson, "The H 264 advanced video compression standard," by John Wiley & Sons, Inc., 2nd edition, 2003.

[Riza_18] P. Rivaz, J. Haughton, "AV1 bitstream & decoding process specification," The Alliance for Open Media, 2018.

[Rodr_11] P. Rodrigues, J. Camacho, F. Matos, "The application of trust region method to estimate the parameters of photovoltaic modules through the use of single and double exponential models," Int. Conf. on Renewable Energies and Power Quality, Las Palmas, Spain, Apr. 2011.

[Rogm_09] S. Rogmans, J. Lu, P. Bekaert, G. Lafruit, "Real-time stereo-based view synthesis algorithms: A unified framework and evaluation on commodity GPUs," Signal Processing: Image Communication, vol. 24, no. 1–2, pp. 49-64, Jan. 2009.

[Roys_08] P. Royston, W. Sauerbrei, "Multivariable model-building: a pragmatic approach to regression analysis based on fractional polynomials for modelling continuous variables," JohnWiley & Sons Ltd, England, 2008.

[Sair_22] S. Sairam, P. Muralidhar, "Deep neural network based next-frame prediction in HEVC video sequence," IEEE Int. Symposium on Smart Electronic Systems (iSES), Warangal, India, Dec. 2022.

[Sala_19] B. Salahieh, B. Kroon, J. Jung, M. Domański, "Test model for immersive video," ISO/IEC JTC1/SC29/WG11, Doc. MPEG/N18470, Geneva, Switzerland, Apr. 2019.

[Sald_22] M. Saldanha, G. Sanchez, C. Marcon, L. Agostini, "Versatile video coding (VVC): machine learning and heuristics," Springer Publishing, 2022.

[Salo_07] D. Salomon, G. Motta, D. Bryant, "Data compression: the complete reference," by Springer, Inc., 4th edition, 2007.

[Same_16] J. Samelak, J. Stankowski, M. Domański, "Adaptation of the 3D-HEVC coding tools to arbitrary locations of cameras," Int. Conf. on Signals and Electronic Systems (ICSES), Kraków, Poland, Sep. 2016.

[Sanc_18] V. Sanchez, "Rate control for HEVC intra-coding based on piecewise linear approximations," IEEE Int. Conf. on Acoustics, Speech and Signal Proc. (ICASSP), Calgary, AB, Canada, Apr. 2018.

[Sanz_13] S. Sanz-Rodríguez, T. Schierl, "A rate control algorithm for HEVC with hierarchical GOP structures," IEEE Int. Conf. Acoustics, Speech and Signal Proc. (ICASSP), Vancouver, BC, Canada, May 2013.

[Schr_22] P. Schröppel, J. Bechtold, A. Amiranashvili, T. Brox, "A benchmark and a baseline for robust multi-view depth estimation," Int. Conf. 3D Vision (3DV), Prague, Czech Republic, Sep. 2022.

[Schw_19] S. Schwarz et al., "Emerging MPEG standards for point cloud compression," IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 9, no. 1, pp. 133-148, Mar. 2019.

[Seno_15] T. Senoh, K. Wakunami, H. Sasaki, R. Oi, K. Yamamoto, "Fast depth estimation using non-iterative local optimization for super multi-view images," IEEE Global Conf. on Signal and Information Proc. (GlobalSIP), Orlando, FL, USA, Dec. 2015.

[Ser_11] J. Ser, "Recent advances on video coding," InTech, 2011.

[Sett_22] T. Setti, Á. Csapó, "Quantifying the effectiveness of project-based editing operations in virtual reality," IEEE Int. Conf. Cognitive Aspects of Virtual Reality (CVR), Budapest, Hungary, May 2022.

[Shao_05] F. Shao, G. Jiang, K. Chen, M. Yu, T. Choi, "Ray-space data compression based on prediction technique," Int. Conf. on Computer Graphics, Imaging and Visualization (CGIV'05), Beijing, China, Jul. 2005.

[Shen_16] L. Shen, Q. Hu, Z. Liu, P. An, "A new rate control algorithm based on region of interest for HEVC," Advances in Multimedia Information Proc. PCM 2016: 17th Pacific-Rim Conf. on Multimedia, Xi´ an, China, Sep. 2016.

[Shi_08] Y. Shi, S. Yue, B. Yin, Y. Huo, "A novel ROI-based rate control scheme for H.264," Int. Conf. for Young Computer Scientists, Hunan, China, Nov. 2008.

[Shi_19] Y. Shi, H. Sun, "Image and video compression for multimedia engineering," Taylor & Francis Group, LLC, 3'rd edition, 2019.

[Si_13] J. Si, S. Ma, W. Gao, "Efficient bit allocation and CTU level rate control for high efficiency video coding," Picture Coding Symposium (PCS), San Jose, CA, USA, Dec. 2013.

[Siqu_20] Í. Siqueira, G. Correa, M. Grellert, "Rate-Distortion and complexity comparison of HEVC and VVC video encoders," IEEE 11th Latin American Symposium on Circuits & Systems (LASCAS), San Jose, Costa Rica, Feb. 2020.

[Sjob_12] R. Sjoberg, Y. Chen, K. Fujibayashi, M. Hannuksela, J. Samuelsson, T. Tan, Y. Wang, S. Wenger, "Overview of HEVC high-level syntax and reference picture management," IEEE Trans. on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1858-1870, Dec. 2012.

[Smol_09] A. Smolic, "An overview of 3D video and free viewpoint video," Int. Conf. Computer Analysis of Images and Patterns (CAIP), Münster, Germany, Sep. 2009.

[Son_01] S. Son, "A transmission scheme for streaming variable bit rate video over internet," Springer-Verlag Berlin Heidelberg, pp. 16–28, 2001.

[Song_17] F. Song, C. Zhu, Y. Liu, Y. Zhou, Y. Liu, "A new GOP level bit allocation method for HEVC rate control," IEEE Int. Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Cagliari, Italy, Jun. 2017.

[Stan_13a] O. Stankiewicz, K. Wegner, M. Domański, "AHG14: optimized QP/QD curve for 3D coding with half and full resolution depth maps," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Doc. JCT-3V E0269, Vienna, Austria, Aug. 2013.

[Stan_13b] O. Stankiewicz, K. Wegner, M. Tanimoto, M. Domański, "Enhanced view synthesis reference software (VSRS) for free-viewpoint television," ISO/IEC JTC 1/SC 29/WG11, Doc. M31520, Geneva, Switzerland, Oct. 2013.

[Stan_13c] O. Stankiewicz, K. Wegner, M. Tanimoto, M. Domanski, "Enhanced depth estimation reference software (DERS) for free-viewpoint television," ISO/IEC JTC1/SC29/WG11, Doc. MPEG M31518, Geneva, Switzerland, Oct. 2013.

[Stan_18] O. Stankiewicz, M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, J. Samelak, "A free-viewpoint television system for horizontal virtual navigation," IEEE Trans. Multimedia, vol. 20, no. 8, pp. 2182-2195, Aug. 2018.

[Sull_13] G. Sullivan, J. Ohm, W. Han, T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," IEEE Trans. on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1649-1668, Dec. 2012.

[Sull_18] G. Sullivan, J. Ohm, "Versatile video coding-towards the next generation of video compression," Picture Coding Symposium, San Francisco, CA, USA, Jun. 2018.

[Sull_98] G. Sullivan, T. Wiegand, "Rate-distortion optimization for video compression," IEEE Signal Proc. Magazine, vol. 15, no.6, pp. 74–90, Nov. 1998.

[Sze_14] V. Sze, M. Budagavi, G. Sullivan, "High efficiency video coding (HEVC) algorithms and architectures," by Springer, Inc., 2014.

[Taja_17] B. Tajali, H. Roodaki, "HEVC-based view level rate-distortion model for multiview video," Iranian Conf. on Electrical Engineering (ICEE), Tehran, Iran, May 2017.

[Tan_17] S. Tan, S. Ma, S. Wang, S. Wang, W. Gao, "Inter-view dependency-based rate control for 3D-HEVC," IEEE Trans. Circuits and Systems for Video Technology, vol. 27, no. 2, pp. 337-351, Feb. 2017.

[Tang_19] M. Tang, J. Wen, Y. Han, "A generalized rate-distortion- λ model based HEVC rate control algorithm," Journal of Multimedia, 2019.

[Tani_09] M. Tanimoto, K. Suzuki, "View synthesis algorithm in view synthesis reference software 2.0 (VSRS2.0)," ISO/IEC JTC1/SC29/WG11, Doc. MPEG2008/M16090, Lausanne, Switzerland, Feb. 2009.

[Tani_12a] M. Tanimoto, "FTV (Free-viewpoint Television) for ray and sound reproducing in 3D space," IEEE Int. Conf. on Acoustics, Speech and Signal Proc. (ICASSP), Kyoto, Japan, Mar. 2012.

[Tani_12b] M. Tanimoto, "FTV: free-viewpoint television," Signal Processing: Image Communication, vol. 27, no. 6, pp. 555–570, 2012.

[Tani_12c] M. Tanimoto, M. Panahpour, T. Fujii, T. Yendo, "FTV for 3 D spatial communication," Proc. IEEE, vol. 100, no. 4, pp. 905-917, Feb. 2012.

[Tech_16] G. Tech, Y. Chen, K. Müller, J. Ohm, A. Vetro, Y. Wang, "Overview of the multiview and 3D extensions of high efficiency video coding," IEEE Trans. Circuits and Systems for Video Technology, vol. 26, no. 1, pp. 35-49, Jan. 2016.

[Tian_18] L. Tian, J. Li, Y. Zhou, H. Wang, "Picture quality assessment-based on rate control for variable bandwidth networks," Tsinghua Science and Technology, vol. 23, no. 4, pp. 396-405, Aug. 2018.

[Topi_19] P. Topiwala, M. Krishnan, W. Dai, "Performance comparison of VVC, AV1 and EVC," Proc. SPIE, Applications of Digital Image Proc. XLII, 1113715, 6 Sep. 2019.

[Tour_09] A. Tourapis, "H.264/14496-10 AVC Reference software manual," Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6), Doc. JVT-AE010, London, UK, Jul. 2009.

[Trow_20] I. Trow, "AV1: implementation, performance, and application," SMPTE Motion Imaging Journal, vol. 129, no. 1, pp. 51-56, Feb. 2020.

[Tulv_16] C. Tulvan, R. Mekuria, Z. Li, S. Laserre, "Use cases for point cloud compression," ISO/IEC JTC1/SC29/WG11, Doc. MPEG2015/ N16331, Geneva, Switzerland, Jun. 2016.

[Ueda_07] S. Ueda, Y. Shinzaki, H. Shigeno, K. Okada, "H.264/AVC stream authentication at the network abstraction layer," IEEE SMC Information Assurance and Security Workshop, West Point, NY, USA, Jun. 2007.

[Ugur_13] K. Ugur, A. Alshin, E. Alshina, F. Bossen, W. Han, J. Park, J. Lainema, "Motion-compensated prediction and interpolation filter design in H.265/HEVC," IEEE Journal of Selected Topics in Signal Proc., vol. 7, no. 6, pp. 946-956, Dec. 2013.

[Vada_22] V. Vadakital, A. Dziembowski, G. Lafruit, F. Thudor, G. Lee, P. Alface, "The MPEG immersive video standard-current status and future outlook," IEEE MultiMedia, vol. 29, no. 3, pp. 101-111, Sep. 2022.

[Vetro_11] A. Vetro, T. Wiegand, G. Sullivan, "Overview of the stereo and multiview video coding extensions of the H.264/MPEG-4 AVC standard," in Proc. of the IEEE, vol. 99, no. 4, pp. 626-642, Apr. 2011.

[VVC] 2D VVC reference codec available online https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/tree/VTM-7.3.

[VVC_std] ISO/IEC 23090-3, MPEG-I Part 3, Versatile Video Coding, Edition 1, 2020 also: ITU-T Rec. H.266 Edition 1.0, 2020.

[Wali_22] W. Walidaniy, M. Yuliana, H. Briantoro, "Improvement of PSNR by using shannon-fano compression technique in AES-LSB StegoCrypto," Int. Electronics Symposium (IES), Surabaya, Indonesia, Aug. 2022.

[Wang_09] Z. Wang, S. Dong, L. Xu, W. Gao, Q. Huang, "Estimating the value of θ in the intra frame for ρ-domain rate control algorithms," Picture Coding Symposium (PCS), Chicago, IL, USA, May 2009.

[Wang_12] J. Wang, L. Rønningen, "Real time believable stereo and virtual view synthesis engine for autostereoscopic display," Int. Conf. on 3D Imaging (IC3D), Liège, Belgium, Dec. 2012.

[Wang_13a] S. Wang, S. Ma, S. Wang, D. Zhao, W. Gao, "Quadratic ρ-domain based rate control algorithm for HEVC," IEEE Int. Conf. Acoustics, Speech and Signal Proc., Vancouver, BC, Canada, May 2013.

[Wang_13b] S. Wang, S. Ma, S. Wang, D. Zhao, W. Gao, "Rate-GOP based rate control for high efficiency video coding," IEEE Journal of selected topics in signal proc., vol. 7, no. 6, pp. 1101-1111, Dec. 2013.

[Wang_15] Q. Wang, "An overview of emerging technologies for high efficiency 3D video coding," Journal of Latex Class Files, vol. 14, no. 8, pp. 1-6, Aug. 2015.

[Wang_16] M. Wang, K. Ngan, H. Li, "Low-delay rate control for consistent quality using distortion-based Lagrange multiplier," IEEE Trans. Image Proc., vol. 25, no. 7, pp. 2943-2955, Jul. 2016.

[Wang_18] S. Wang, J. Li, S. Wang, S. Ma, W. Gao, "A frame level rate control algorithm for screen content coding," IEEE Int. Symposium on Circuits and Systems (ISCAS), Florence, Italy, May 2018.

[Wang_20a] M. Wang, S. Wang, J. Li, L. Zhang, Y. Wang, S. Ma, "SSIM motivated quality control for versatile video coding," 2020 Asia-Pacific Signal and Information Proc. Association Annual Summit and Conf. (APSIPA ASC), Auckland, New Zealand, Dec. 2020.

[Wang_20b] X. Wang, X. Ding, Q. Qu, "A new filter nonmonotone adaptive trust region method for unconstrained optimization," MDPI, vol. 12, no. 2, pp. 1-13, 2020.

[Wang_21] Y. Wang, R. Skupin, M. Hannuksela, S. Deshpande, Hendry, V. Drugeon, R. Sjöberg, B. Choi, V. Seregin, Y. Sanchez, J. Boyce, W. Wan, G. Sullivan, "The high-level syntax of the versatile video coding (VVC) standard," IEEE Trans. Circuits and Systems for Video Technology, vol. 31, no. 10, pp. 3779-3800, Oct. 2021.

[Wang_22] X. Wang, M. Lu, Z. Ma, "Block-level rate control for learnt image coding," Picture Coding Symposium (PCS), San Jose, CA, USA, Dec. 2022.

[Wedi_03] T. Wedi, H. Musmann, "Motion- and aliasing-compensated prediction for hybrid video coding," IEEE Trans. on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 577-586, 2003.

[Wegn_17] K. Wegner, O. Stankiewicz, A. Dziembowski, D. Mieloch, M. Domański, "Exploration experiments on omnidirectional 6-DoF/3-DoF+ rendering," ISO/IEC JTC1/SC29/WG11, Doc. M41807, Macao, China, Oct. 2017.

[Wei_05] S. Weisberg, "Applied linear regression," by John Wiley & Sons, Inc., 3rd edition, 2005.

[Weng_19] S. Wenger, B. Choi, S. Liu, "AHG17: On NAL unit header design for VVC," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Doc. JVET-M0520, Marrakech, Morocco, Jan. 2019.

[Wieg_03] T. Wiegand, G. Sullivan, G. Bjontegaard, A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Trans. on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560-576, Jul. 2003.

[Wien_15] M. Wien, "High efficiency video coding tools and specification," by Springer, Inc., 2015.

[Wien_19] M. Wien, J. M. Boyce, T. Stockhammer, W. Peng, "Standardization status of immersive video coding," IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 9, no. 1, pp. 5-17, Mar. 2019.

[Wink_05] S. Winkler, "Digital video quality vision models and metrics," John Wiley & Sons Ltd., Chichester, England, 1'st edition, 2005.

[Wu_11] Z. Wu, S. Xie, K. Zhang, R. Wu, "Rate control in video coding," InTech, 2011.

[Wu_14] C. Wu, P. Su, C. Yeh, H. Hsu, "A joint content adaptive rate-quantization model and region of interest intra coding of H.264/AVC," Int. Conf. Multimedia and Expo (ICME), Chengdu, China, Jul. 2014.

[Wu_15] W. Wu, J. Liu, L. Feng, "A novel rate control scheme for low delay video coding of HEVC," ETRl Journal, vol. 38, no. 1, pp. 186-194, Sep. 2015.

[Xiao_11] C. Xiaojing, D. Lun-hui, "Improved rate control scheme in H.264 for real-time communication," Int. Conf. on Electric Information and Control Engineering, Wuhan, China, Apr. 2011.

[Xu_16] L. Xu, C. Zhu, Y. Zhou, Y. Wang, Y. Gao, "Introducing GOP-level quantization parameter offset in high efficiency video coding," IEEE Int. Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Nara, Japan, Jun. 2016.

[Yama_10] N. El-Yamany, K. Ugur, M. Hannuksela, M. Gabbouj, "Evaluation of depth compression and view synthesis distortions in multiview-video-plus-depth coding systems," 3DTV-Conf.: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), Tampere, Finland, Jun. 2010.

[Yan_09] X. Yan, X. Su, "Linear regression analysis theory and computing," World Scientific Publishing, 2009.

[Yan_20a] T. Yan, I. Ra, H. Wen, M. Weng, Q. Zhang, Y. Che, "CTU layer rate control algorithm in scene change video for free-viewpoint video," IEEE Access, vol. 8, pp. 24549-24560, 2020.

[Yan_20b] T. Yan, I. Ra, H. Wen, M. Weng, Q. Zhang, Y. Che, "Corrections to "CTU layer rate control algorithm in scene change video for free-viewpoint video"," IEEE Access, vol. 8, pp. 68982-68982, 2020.

[Yan_22] J. Yan, J. Li, Y. Fang, Z. Che, X. Xia, Y. Liu, "Subjective and objective quality of experience of free viewpoint videos," IEEE Trans. on Image Proc., vol. 31, pp. 3896-3907, 2022.

[Yang_14] Z. Yang, L. Song, Z. Luo, X. Wang, "Low delay rate control for HEVC," IEEE Int. Symposium on Broadband Multimedia Systems and Broadcasting, Beijing, China, Aug. 2014.

[Yang_20] H. Yang, L. Shen, Y. Yang, W. Lin, "A novel rate control scheme for video coding in HEVC-SCC," IEEE Trans. on Broadcasting, vol. 66, no. 2, pp. 333-345, Jun. 2020.

[Yann_20] T. Yan, I. Ra, Q. Zhang, H. Xu, L. Huang, "A novel rate control algorithm based on ρ model for multiview high efficiency video coding," Electronics, vol. 9, no. 1, pp. 1-13, Jan. 2020.

[Yao_16] L. Yao, Y. Liu, W. Xu, "Real-time virtual view synthesis using light field," EURASIP Journal on Image and Video Proc., 25 (2016), Sep. 2016.

[Yu_13] L. Yu, G. Fu, A. Men, B. Luo, H. Zhao, "A novel motion compensated prediction framework using weighted AMVP prediction for HEVC," Visual Communications and Image Proc. (VCIP), Kuching, Malaysia, Nov. 2013.

[Yu_16] L. Yu, Q. Wang, A. Lu, Y. Sun, "Response to call for evidence on free-viewpoint television: Zhejiang University," ISO/IEC JTC1/SC29/WG11, Doc. MPEG2016/m37608. San Diego, USA, Feb. 2016.

[Zhan_11] F. Zhang, E. Steinbach, "Improved ρ-domain rate control with accurate header size estimation," IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, May 2011.

[Zhan_17] Z. Zhang, T. Jing, J. Han, Y. Xu, F. Zhang, "A new rate control scheme for video coding based on region of interest," IEEE Access, vol. 5, pp. 13677-13688, 2017.

[Zhan_19] M. Zhang, G. Zhang, H. Wei, W. Zhou, Z. Duan, "GOP level quality dependency based frame level rate control algorithm," IEEE Global Conf. on Signal and Information Proc. (GlobalSIP), Ottawa, ON, Canada, Nov. 2019.

[Zhan_22] W. Zhang, Y. Wang, Y. Liu, "Generating high-quality panorama by view synthesis based on optical flow estimation," Sensors 22, no.2:470, 2022.

[Zhao_22] Z. Zhao, X. He, S. Xiong, L. He, R. E. Sheriff, "An optimized rate control algorithm in versatile video coding for 360∘ videos," IEEE Signal Proc. Letters, vol. 29, pp. 2303-2307, 2022.

[Zhou_23a] G. Zhou, Z. Luo, M. Hu, D. Wu, "PreSR: neural-enhanced adaptive streaming of VBR-encoded videos with selective prefetching," IEEE Trans. Broadcasting, vol. 69, no. 1, pp. 49-61, March 2023.

[Zhou_23b] M. Zhou, X. Wei, W. Jia, S. Kwong, "Joint decision tree and visual feature rate control optimization for VVC UHD coding," IEEE Trans. Image Proc., vol. 32, pp. 219-234, 2023.

[Zhu_13] C. Zhu, H. Jia, S. Zhang, X. Huang, X. Xie, W. Gao, "On a highly efficient RDO-based mode decision pipeline design for AVS," IEEE Trans. on Multimedia, vol. 15, no. 8, pp. 1815-1829, Dec. 2013.

[Zitn_04] L. Zitnick, S. Kang, M. Uyttendaele, S. Winder, R. Szeliski, "High quality video view interpolation using a layered representation," ACM SIGGRAPH, 2004.

Appendix A

Related to Chapter Four

Appendix A shows the values of parameter *a* of AVC, HEVC, and VVC models that were estimated directly from experimental data of the training set (Table 3.1) for AVC, HEVC, and VVC, respectively by minimization of the square error according to Eq. 4.2. While, the values of parameters *b* and *c* of AVC, HEVC, and VVC models are estimated directly from experimental data for AVC, because parameters *b* and *c* of the model used for codecs have similar values (shown in Section 4.3). The mean relative approximation errors for individual frames of various types, as well as for total bitrate, are also presented in Appendix A. In Appendix B, the approximated curves are plotted using the parameters listed in Appendix A.

Table A.1: The model parameters and the average relative approximation error for I-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
PeopleOnStreet	34342.97	0.94	-0.19	4.27	2.86
Traffic	48707.99	1.06	1.22	3.02	1.75
Kermit	53330.55	1.15	2.99	2.24	1.98
Poznan_Block2	8328.46	1.03	-2.52	3.09	2.35
Poznan_Fencing	10075.77	1.06	1.00	5.61	4.07
BBB.Butterfly	1358.88	0.82	-1.46	5.42	4.57
BBB.Flowers	7786.94	1.01	0.48	4.80	3.23
Ballet	1061.70	0.87	-1.69	3.94	2.74
Breakdancers	864.57	0.81	-3.59	2.40	1.75
Keiba	4702.90	1.07	2.26	1.52	1.23
RaceHorses	36182.55	1.44	26.03	2.54	2.04
Basketball_Drill	6370.55	1.07	1.00	4.28	3.40
Basketball_Pass	1377.00	1.05	1.74	1.37	1.27
BQSquare	6773.02	1.17	15.25	4.10	2.48

Table A.2: The model parameters and the average relative approximation error for P-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
PeopleOnStreet	31913.28	0.96	1.00	3.96	2.11
Traffic	35592.37	1.26	1.00	3.24	2.41
Kermit	27261.32	1.22	-9.01	2.54	1.92
Poznan_Block2	735.97	0.93	-7.00	6.86	6.33
Poznan_Fencing	2581.39	1.03	-4.00	5.67	3.75
BBB.Butterfly	1644.57	1.07	1.00	6.29	5.60
BBB.Flowers	3360.66	1.04	1.00	7.45	5.12
Ballet	588.91	0.96	-1.00	3.14	2.66
Breakdancers	979.68	0.92	-4.58	3.88	2.56
Keiba	5209.49	1.12	2.51	2.15	1.84
RaceHorses	30751.48	1.44	17.81	3.57	3.41
Basketball_Drill	2121.91	1.03	0.19	2.60	2.13
Basketball_Pass	1193.70	1.12	1.00	7.42	4.17
BQSquare	2006.90	1.27	1.00	5.47	4.46

Table A.3: The model parameters and the average relative approximation error for B0-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
PeopleOnStreet	18990.00	0.96	1.00	1.10	0.90
Traffic	4002.00	0.95	1.00	6.97	4.87
Kermit	6415.00	1.07	-5.00	8.72	4.53
Poznan_Block2	100.00	0.77	-5.29	3.19	2.78
Poznan_Fencing	530.99	0.85	-3.94	2.85	2.29
BBB.Butterfly	542.00	0.91	1.00	5.21	3.74
BBB.Flowers	750.00	0.90	1.00	4.54	3.92
Ballet	282.08	0.96	1.00	8.82	5.88
Breakdancers	616.06	0.94	-5.15	5.63	3.45
Keiba	6240.20	1.19	2.50	3.73	2.56
RaceHorses	6800.00	1.20	6.00	3.82	3.95
Basketball_Drill	1421.54	1.05	1.00	3.14	2.42
Basketball_Pass	618.14	1.14	1.00	8.50	4.53
BQSquare	360.00	1.11	-3.00	7.91	5.56

Table A.4: The model parameters and the average relative approximation error for B1-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
PeopleOnStreet	15100.00	0.97	1.00	1.69	1.53
Traffic	1920.00	0.88	1.00	10.58	5.18
Kermit	1397.00	0.85	-5.00	8.98	4.32
Poznan_Block2	87.00	0.81	-5.17	3.87	2.75
Poznan_Fencing	200.00	0.69	-4.43	1.47	1.34
BBB.Butterfly	80.24	0.64	-3.65	2.30	1.61
BBB.Flowers	1095.44	1.04	1.00	11.95	6.83
Ballet	228.20	0.95	1.00	9.96	6.08
Breakdancers	548.63	0.95	-5.80	7.93	4.85
Keiba	4450.51	1.14	2.50	2.08	2.41
RaceHorses	4630.00	1.21	6.00	7.28	4.54
Basketball_Drill	1403.86	1.11	1.00	5.88	3.86
Basketball_Pass	686.13	1.23	1.00	5.42	4.62
BQSquare	350.00	1.22	-5.00	8.49	5.02

Table A.5: The model parameters and the average relative approximation error for B2-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
PeopleOnStreet	12695.00	1.02	1.00	2.46	1.53
Traffic	64.00	0.37	-2.14	7.05	4.84
Kermit	90.05	0.44	-2.63	5.83	4.56
Poznan_Block2	21.00	0.62	-3.62	4.88	3.12
Poznan_Fencing	50.04	0.51	-3.23	1.19	1.09
BBB.Butterfly	22.00	0.47	-2.97	1.96	1.47
BBB.Flowers	21.00	0.40	-2.62	1.60	1.22
Ballet	10.10	0.41	-2.55	1.71	1.31
Breakdancers	73.00	0.61	-4.30	1.39	1.33
Keiba	1140.00	0.92	2.00	4.03	2.69
RaceHorses	225.56	0.73	-7.02	3.13	2.28
Basketball_Drill	620.00	1.03	1.00	9.91	4.65
Basketball_Pass	498.00	1.28	1.00	5.12	4.27
BQSquare	602.00	1.54	-5.00	9.60	7.26

Table A.6: The model parameters and the average relative approximation error for B3-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
PeopleOnStreet	14910.00	1.05	1.00	1.96	1.39
Traffic	35.00	0.27	-2.00	3.71	2.68
Kermit	54.00	0.34	-2.60	3.89	2.59
Poznan_Block2	12.30	0.51	-3.72	2.73	1.84
Poznan_Fencing	31.00	0.43	-3.20	7.00	3.25
BBB.Butterfly	53.15	0.56	-3.42	1.64	1.59
BBB.Flowers	86.00	0.59	-3.61	1.59	1.47
Ballet	10.00	0.41	-2.50	1.31	1.10
Breakdancers	75.98	0.62	-4.56	1.27	1.31
Keiba	210.05	0.65	-4.99	2.10	1.82
RaceHorses	220.00	0.70	-7.00	3.60	3.10
Basketball_Drill	530.00	0.97	1.00	5.53	4.18
Basketball_Pass	450.00	1.23	1.00	4.82	3.03
BQSquare	57.00	0.90	-5.00	9.38	6.20

Table A.7: The model parameters and the average relative approximation error for GOP-level size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
PeopleOnStreet	815.01	0.50	-2.38	5.10	4.28
Traffic	814.02	0.67	-2.43	1.62	1.30
Kermit	530.00	0.62	-2.78	2.05	1.49
Poznan_Block2	200.00	0.79	-4.00	2.40	1.86
Poznan_Fencing	107.02	0.56	-2.55	2.21	2.06
BBB.Butterfly	30.00	0.44	-2.06	3.98	3.45
BBB.Flowers	105.00	0.55	-2.31	3.05	2.81
Ballet	10.10	0.36	-1.90	3.22	2.40
Breakdancers	16.00	0.35	-2.00	5.66	3.63
Keiba	40.00	0.36	-2.00	7.35	4.68
RaceHorses	190.00	0.64	-3.00	6.25	4.53
Basketball_Drill	80.00	0.58	-2.87	4.96	4.49
Basketball_Pass	52.01	0.78	-3.78	4.76	4.39
BQSquare	58.00	0.71	-2.68	3.25	2.49

Table A.8: The model parameters and the average relative approximation error for I-frame size for the HEVC and VVC codecs.

Sequences	HEVC			VVC		
	a	Relative error [%]		*a*	Relative error [%]	
		mean	std. dev		mean	std. dev
PeopleOnStreet	23700.00	0.89	0.66	20799.08	1.48	0.93
Traffic	34561.48	1.94	2.72	30320.00	1.75	1.99
Kermit	38000.00	1.89	1.77	34846.87	1.44	1.08
Poznan_Block2	5200.01	1.65	1.35	4626.05	2.53	2.86
Poznan_Fencing	6590.00	4.19	2.74	5937.08	4.33	3.68
BBB.Butterfly	850.02	1.44	1.15	726.83	1.34	0.95
BBB.Flowers	5000.00	0.64	0.55	4382.05	0.89	0.60
Ballet	696.99	1.11	1.16	590.03	1.52	1.13
Breakdancers	550.04	2.14	1.44	477.06	2.75	2.29
Keiba	3540.00	0.75	0.64	3153.74	1.66	1.67
RaceHorses	25699.98	3.20	2.39	23777.70	4.79	3.14
Basketball_Drill	4053.71	2.19	1.81	3410.00	2.57	1.62
Basketball_Pass	1025.66	0.89	0.66	904.93	2.29	3.26
BQSquare	5565.00	2.95	1.61	5199.97	3.00	2.34

Table A.9: The model parameters and the average relative approximation error for P-frame size for the HEVC and VVC codecs.

Sequences	HEVC			VVC		
	a	Relative error [%]		*a*	Relative error [%]	
		mean	std. dev		mean	std. dev
PeopleOnStreet	22300.00	1.79	1.07	18724.63	1.63	1.92
Traffic	23300.00	1.74	1.23	19378.93	1.88	2.72
Kermit	18600.00	3.86	3.28	13930.55	7.52	8.57
Poznan_Block2	460.27	6.58	4.22	410.00	8.51	5.08
Poznan_Fencing	1634.50	4.84	2.96	1440.00	4.80	3.58
BBB.Butterfly	1189.47	2.25	1.22	1040.00	2.43	1.78
BBB.Flowers	2292.80	1.52	1.08	1970.00	1.80	1.39
Ballet	381.55	1.50	1.42	330.00	1.26	1.09
Breakdancers	640.07	2.38	1.71	562.56	2.88	2.50
Keiba	3804.00	2.00	2.04	3355.88	3.20	4.05
RaceHorses	20317.55	3.02	1.87	17999.97	3.55	2.36
Basketball_Drill	1500.00	2.25	1.82	1298.79	2.34	1.57
Basketball_Pass	932.98	5.99	4.62	840.00	5.17	3.43
BQSquare	1700.00	2.12	1.56	1285.23	6.68	6.05

Table A.10: The model parameters and the average relative approximation error for B0-frame size for the HEVC and VVC codecs.

Sequences	HEVC			VVC		
	a	Relative error [%]		*a*	Relative error [%]	
		mean	std. dev		mean	std. dev
PeopleOnStreet	18927.62	2.37	1.46	15852.84	1.83	1.89
Traffic	2453.93	25.13	17.50	1959.45	27.18	19.37
Kermit	5535.37	11.48	17.31	3671.57	14.78	23.33
Poznan_Block2	109.43	13.79	13.50	100.12	14.64	15.39
Poznan_Fencing	465.42	11.04	9.88	391.81	11.35	11.14
BBB.Butterfly	483.41	8.97	7.02	386.39	10.07	7.63
BBB.Flowers	596.91	12.48	9.70	503.15	12.80	10.17
Ballet	302.00	2.16	2.14	257.94	1.86	2.53
Breakdancers	567.11	3.66	2.41	495.38	3.86	3.09
Keiba	3000.00	2.25	2.05	2465.33	3.79	2.95
RaceHorses	5284.51	13.19	9.40	4326.32	15.05	12.07
Basketball_Drill	1280.00	3.55	2.50	1112.04	3.63	2.97
Basketball_Pass	720.00	5.01	3.23	650.57	5.94	3.80
BQSquare	372.30	10.84	11.38	308.10	11.82	11.25

Table A.11: The model parameters and the average relative approximation error for B1-frame size for the HEVC and VVC codecs.

Sequences	HEVC			VVC		
	a	Relative error [%]		*a*	Relative error [%]	
		mean	std. dev		mean	std. dev
PeopleOnStreet	15638.91	1.79	1.57	13202.83	4.03	6.19
Traffic	873.84	27.66	17.73	620.54	38.27	23.13
Kermit	1029.24	14.53	16.65	684.87	17.96	19.92
Poznan_Block2	83.22	14.47	6.37	75.75	15.19	8.23
Poznan_Fencing	147.67	14.62	7.31	121.28	18.66	14.74
BBB.Butterfly	68.41	11.68	6.20	55.55	14.13	12.34
BBB.Flowers	997.00	1.54	1.52	818.81	1.50	1.37
Ballet	221.00	1.43	0.94	191.72	1.55	1.17
Breakdancers	490.00	2.87	1.57	430.45	3.61	2.89
Keiba	1700.00	2.61	2.43	1424.41	2.44	2.12
RaceHorses	4282.80	6.87	5.96	3583.48	6.99	7.11
Basketball_Drill	1340.00	2.58	1.94	1155.05	4.36	3.45
Basketball_Pass	804.00	5.03	3.66	711.45	4.30	2.24
BQSquare	318.68	8.16	9.03	255.58	10.29	7.21

Table A.12: The model parameters and the average relative approximation error for B2-frame size for the HEVC and VVC codecs.

Sequences	HEVC			VVC		
	a	Relative error [%]		*a*	Relative error [%]	
		mean	std. dev		mean	std. dev
PeopleOnStreet	13973.90	3.59	2.91	11213.64	13.88	19.85
Traffic	17.57	38.48	30.23	8.78	59.43	25.04
Kermit	50.71	26.99	21.28	33.25	37.35	23.84
Poznan_Block2	16.94	19.36	11.48	15.13	22.14	13.25
Poznan_Fencing	35.36	21.20	8.59	28.14	29.01	18.33
BBB.Butterfly	16.11	15.66	6.45	13.50	20.27	13.05
BBB.Flowers	16.31	20.15	8.15	12.37	29.23	17.74
Ballet	9.34	14.17	5.57	7.76	20.78	14.66
Breakdancers	65.89	13.74	5.99	55.23	20.44	15.31
Keiba	431.07	14.10	11.86	357.71	16.46	11.49
RaceHorses	204.79	18.31	7.78	165.01	25.05	16.27
Basketball_Drill	585.87	12.11	5.73	475.99	17.56	9.62
Basketball_Pass	615.21	4.28	4.10	527.71	1.63	1.89
BQSquare	826.54	11.55	6.94	591.13	25.09	22.59

Table A.13: The model parameters and the average relative approximation error for B3-frame size for the HEVC and VVC codecs.

Sequences	HEVC			VVC		
	a	Relative error [%]		*a*	Relative error [%]	
		mean	std. dev		mean	std. dev
PeopleOnStreet	7530.82	16.33	6.24	4035.92	44.43	32.78
Traffic	4.40	35.77	29.87	2.19	51.09	26.64
Kermit	9.77	25.09	20.24	5.66	40.17	20.39
Poznan_Block2	4.46	11.77	6.37	4.09	12.07	9.71
Poznan_Fencing	12.29	19.68	7.93	9.45	30.41	18.26
BBB.Butterfly	19.87	12.85	5.34	15.75	17.30	9.16
BBB.Flowers	29.17	18.73	9.89	21.65	25.70	13.83
Ballet	6.56	14.92	6.21	5.11	23.34	12.62
Breakdancers	52.15	13.85	5.13	44.16	21.78	14.09
Keiba	62.43	14.79	8.05	49.29	23.13	12.74
RaceHorses	81.35	24.59	12.85	63.09	36.67	20.59
Basketball_Drill	199.58	19.44	14.28	155.03	25.68	15.01
Basketball_Pass	213.33	6.79	4.23	178.16	12.00	14.21
BQSquare	17.33	7.62	3.98	13.23	23.20	24.80

Table A.14: The model parameters and the average relative approximation error for GOP-level size for the HEVC and VVC codecs.

Sequences	HEVC			VVC		
	a	Relative error [%]		*a*	Relative error [%]	
		mean	std. dev		mean	std. dev
PeopleOnStreet	618.85	10.41	5.43	446.71	24.38	17.04
Traffic	458.63	19.34	10.57	385.49	23.00	15.54
Kermit	312.19	19.24	11.99	246.84	25.33	16.93
Poznan_Block2	127.14	6.42	3.81	111.38	9.48	7.07
Poznan_Fencing	68.33	13.08	6.21	57.50	20.42	17.18
BBB.Butterfly	17.51	11.31	6.41	14.21	18.85	11.81
BBB.Flowers	62.46	13.41	5.89	51.35	19.48	12.62
Ballet	7.09	10.74	4.42	5.85	19.13	14.12
Breakdancers	11.19	13.74	7.46	8.93	25.71	16.14
Keiba	17.07	17.52	8.74	13.91	28.72	17.51
RaceHorses	111.16	23.14	12.06	90.95	35.36	18.75
Basketball_Drill	52.59	11.71	5.02	43.09	19.99	15.43
Basketball_Pass	42.47	6.90	3.95	35.71	12.25	9.82
BQSquare	48.35	7.98	9.34	44.03	8.70	13.62

Appendix B

Related to Chapter Four

In Appendix B, the approximate curves are plotted using the parameters listed in Appendix A. The difference between the experimental and approximated curves can be noticed in the figures shown in Appendix B.

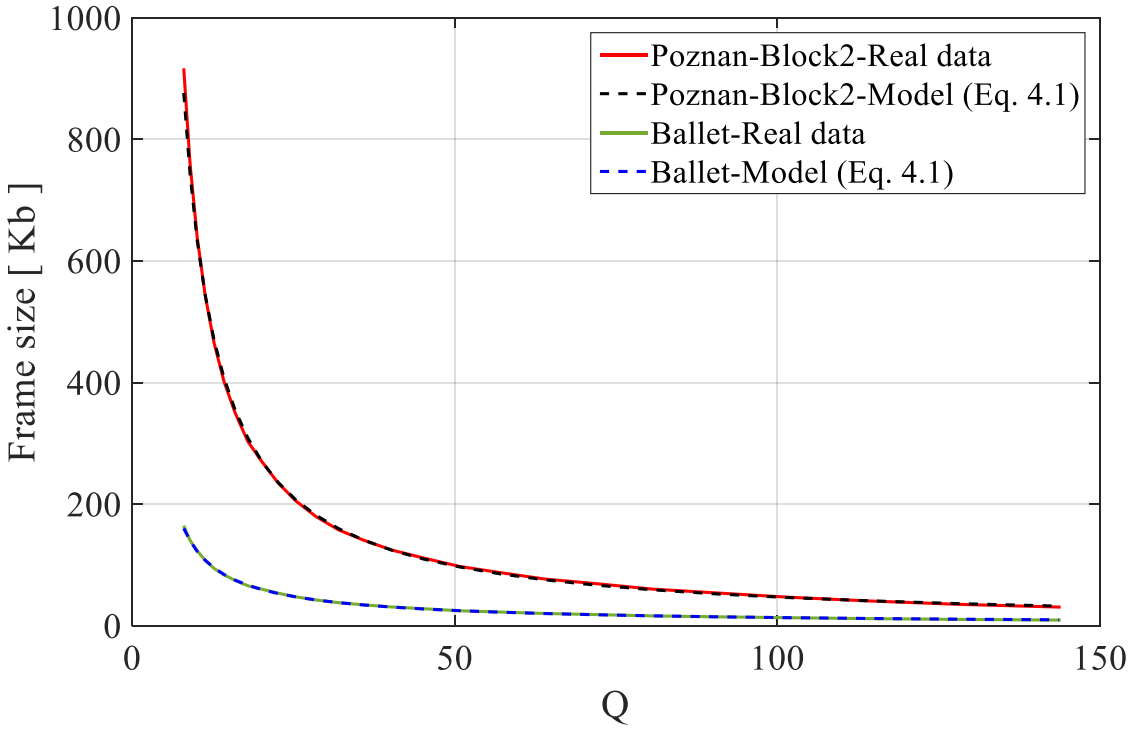


Fig. B.1. The experimental and approximated curves to frame size for I-frame for *Poznan_Block2* and *Ballet* sequences for the HEVC coding.

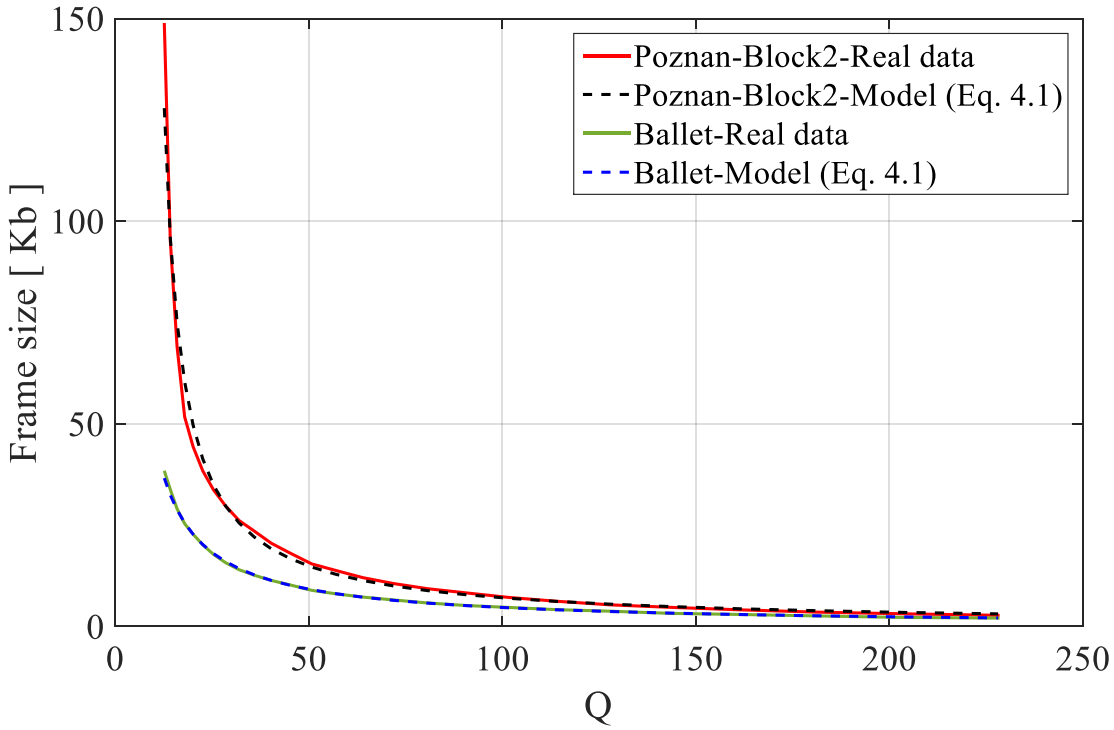


Fig. B.2. The experimental and approximated curves to frame size for P-frame for *Poznan_Block2* and *Ballet* sequences for the HEVC coding.

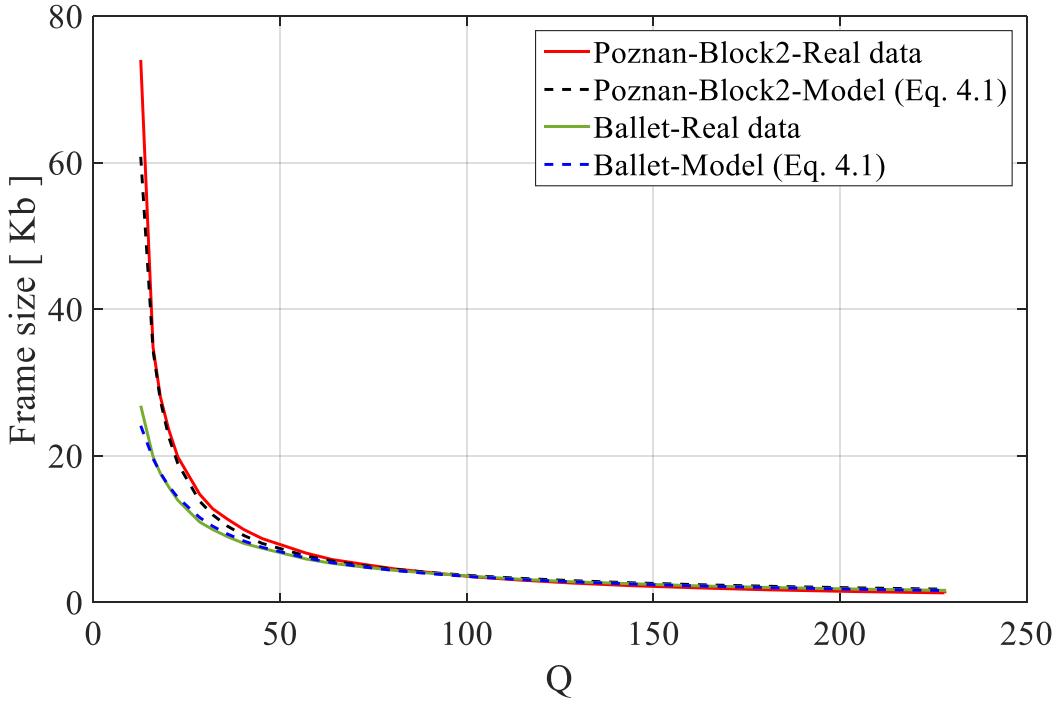


Fig. B.3. The experimental and approximated curves to frame size for B0-frame for *Poznan_Block2* and *Ballet* sequences for the HEVC coding.

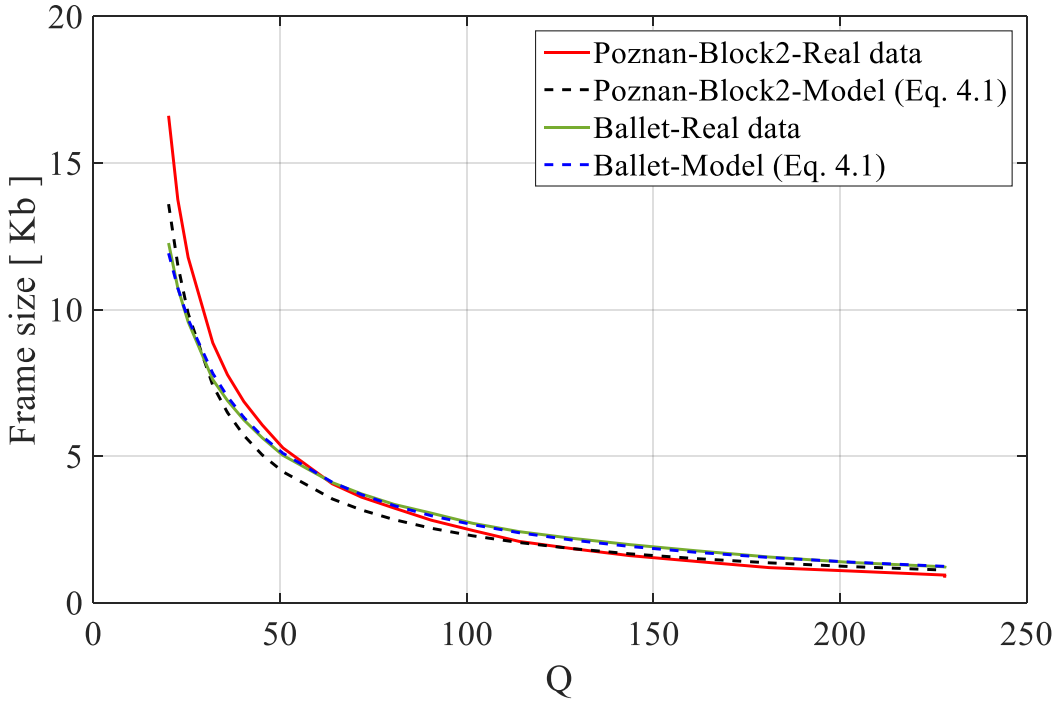


Fig. B.4. The experimental and approximated curves to frame size for B1-frame for *Poznan_Block2* and *Ballet* sequences for the HEVC coding.

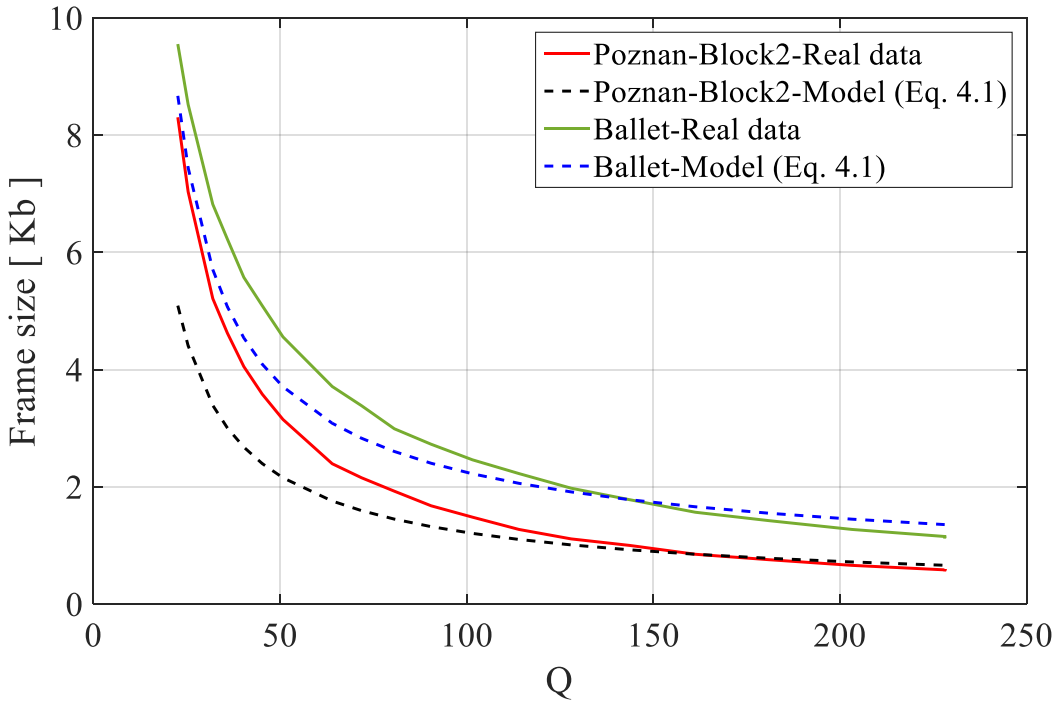


Fig. B.5. The experimental and approximated curves to frame size for B2-frame for *Poznan_Block2* and *Ballet* sequences for the HEVC coding.

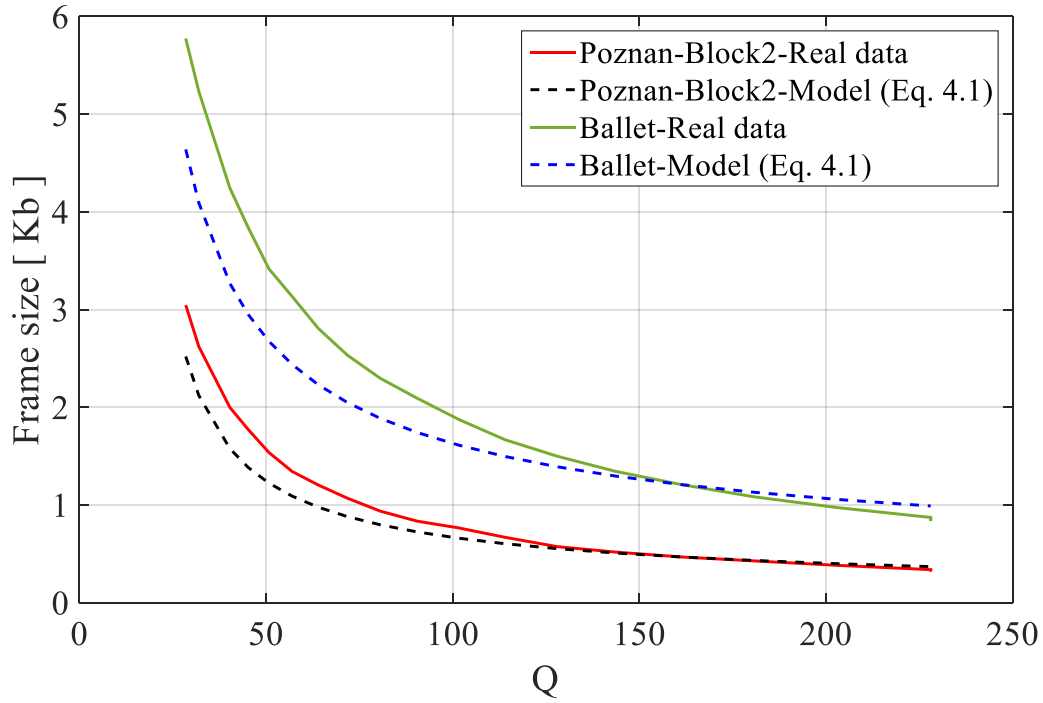


Fig. B.6. The experimental and approximated curves to frame size for B3-frame for *Poznan_Block2* and *Ballet* sequences for the HEVC coding.

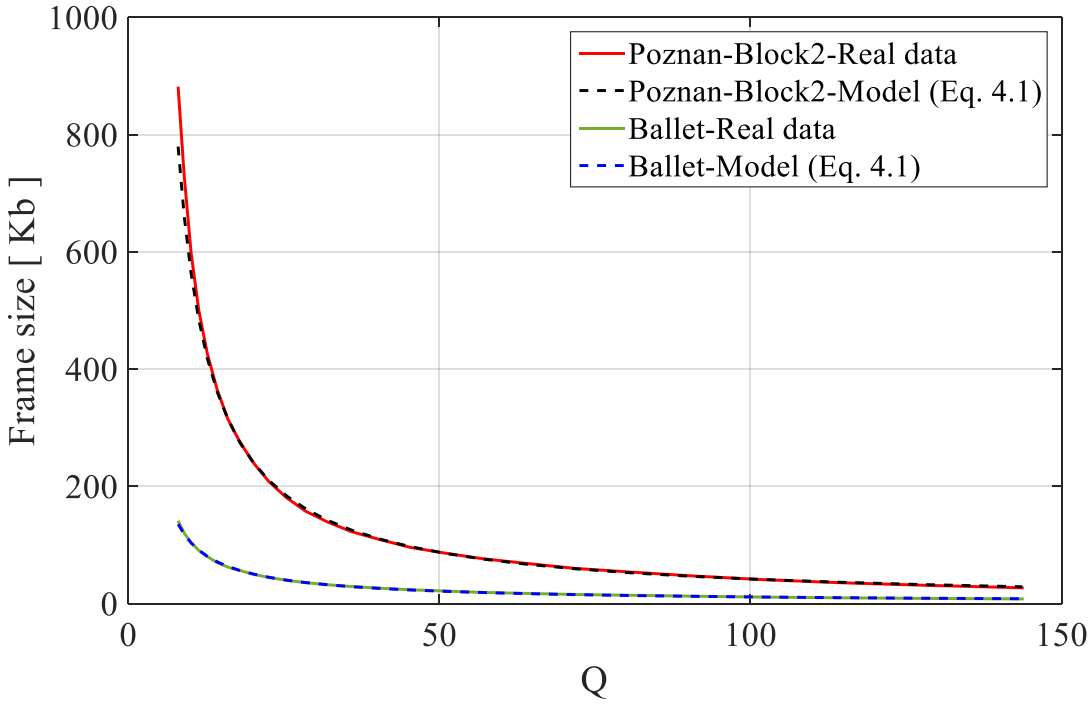


Fig. B.7. The experimental and approximated curves to frame size for I-frame for *Poznan_Block2* and *Ballet* sequences for the VVC coding.

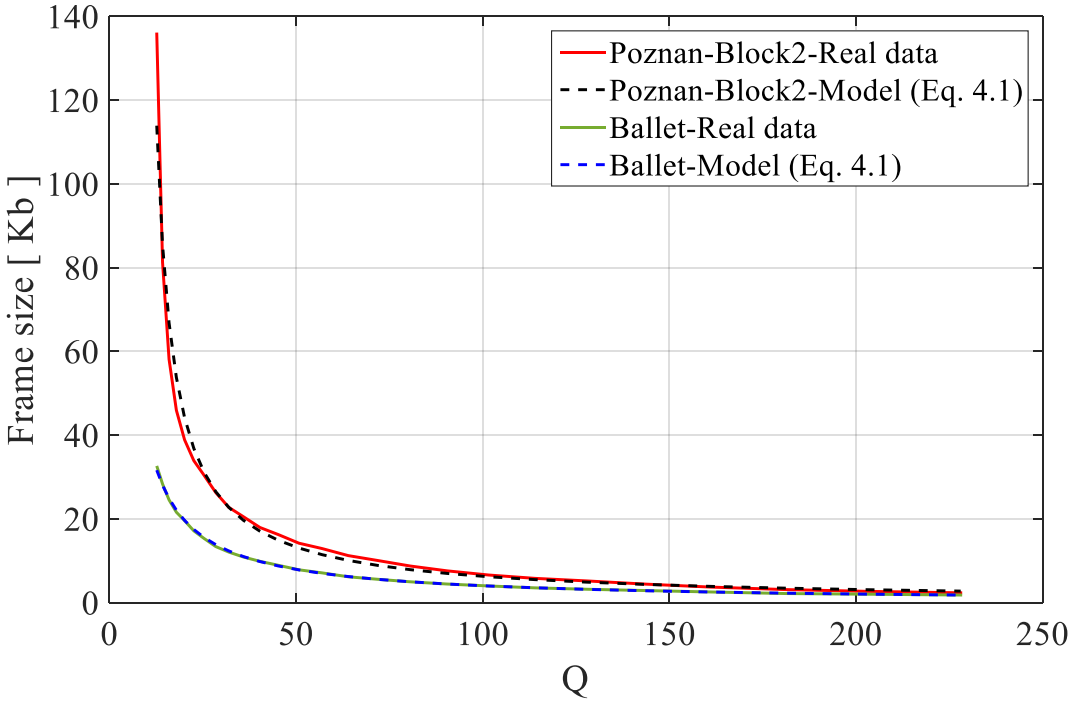


Fig. B.8. The experimental and approximated curves to frame size for P-frame for *Poznan_Block2* and *Ballet* sequences for the VVC coding.

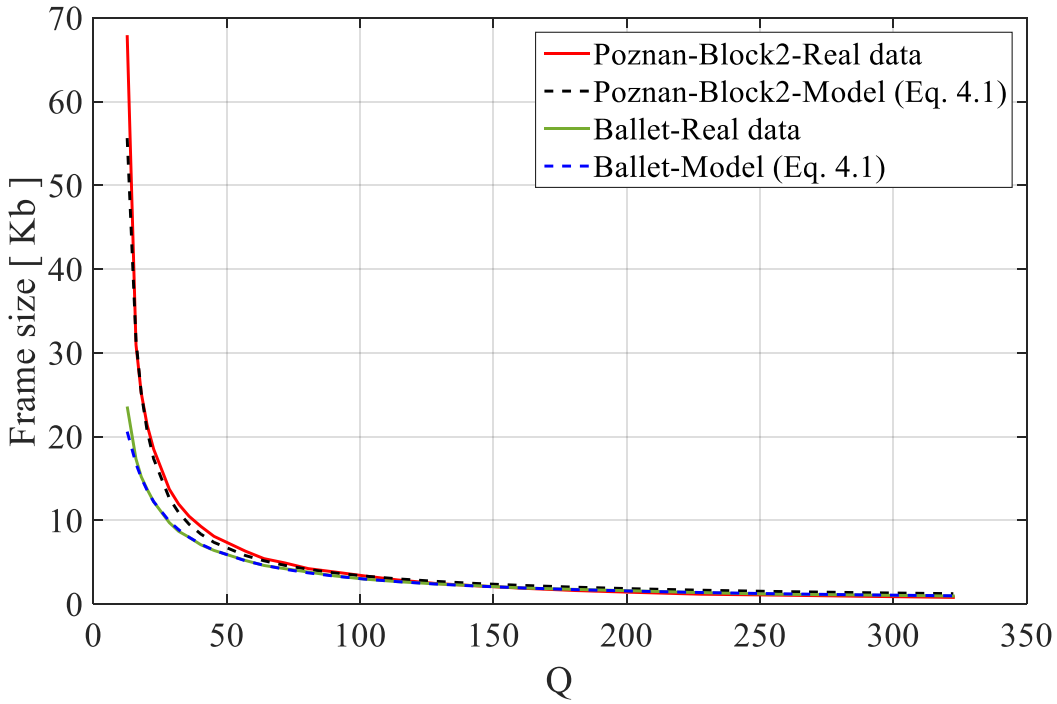


Fig. B.9. The experimental and approximated curves to frame size for B0-frame for *Poznan_Block2* and *Ballet* sequences for the VVC coding.

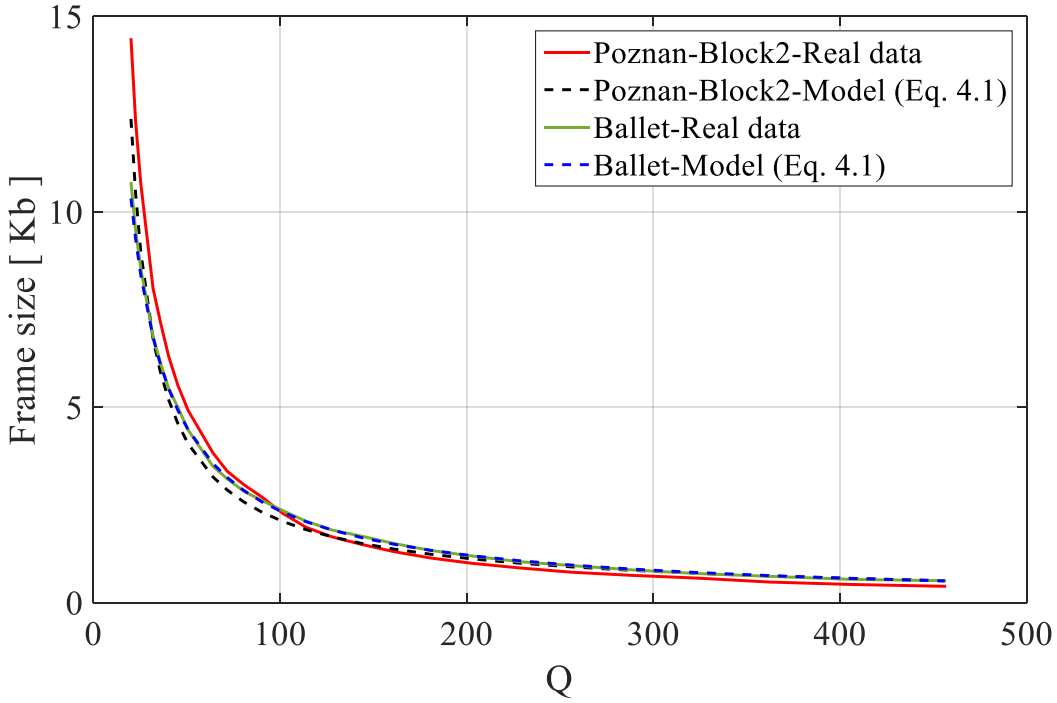


Fig. B.10. The experimental and approximated curves to frame size for B1-frame for *Poznan_Block2* and *Ballet* sequences for the VVC coding.

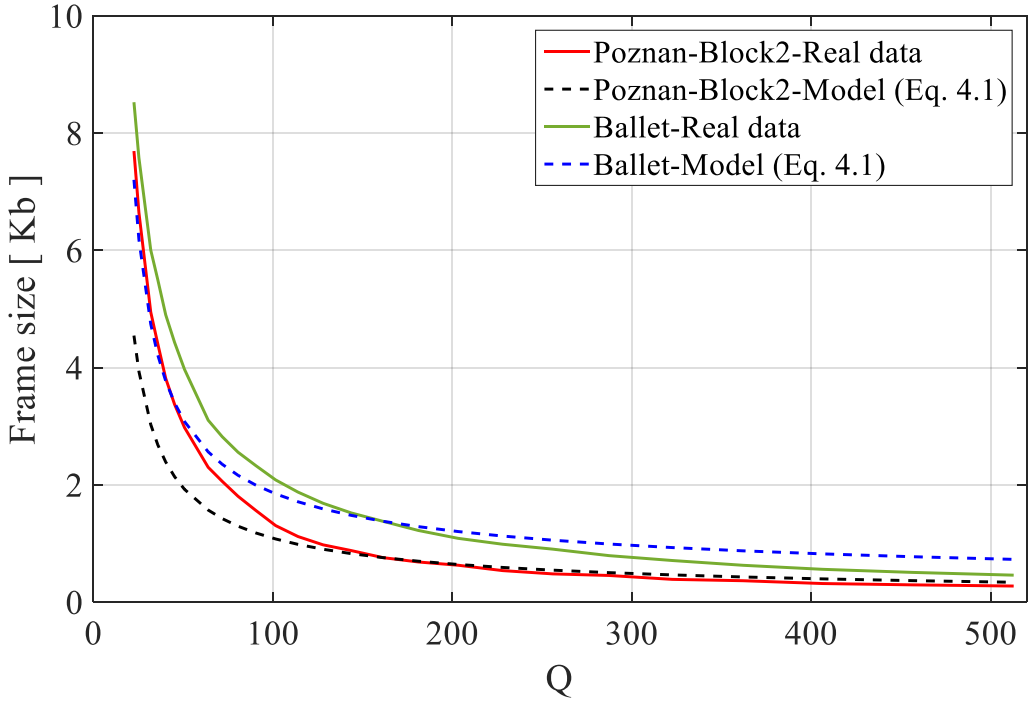


Fig. B.11. The experimental and approximated curves to frame size for B2-frame for *Poznan_Block2* and *Ballet* sequences for the VVC coding.

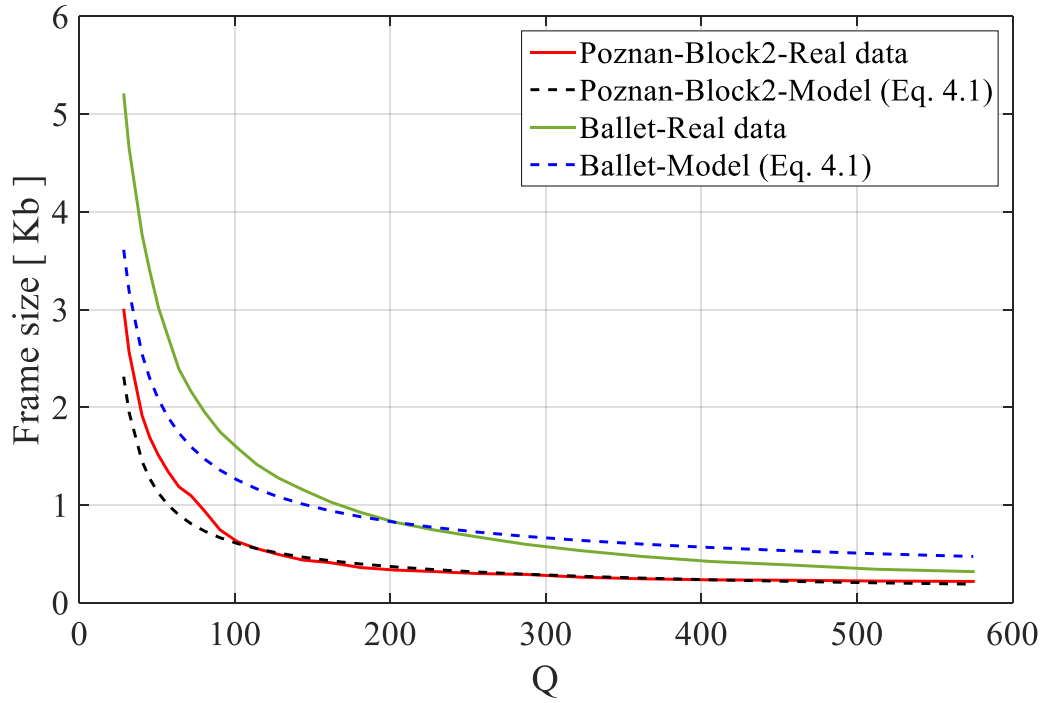


Fig. B.12. The experimental and approximated curves to frame size for B3-frame for *Poznan_Block2* and *Ballet* sequences for the VVC coding.

Appendix C

Related to Chapter Four

Appendix C presents the values of parameters (*a*, *b*, and *c*) of the AVC model that were estimated directly from experimental data of the verification set (Table 3.1) for AVC coding by minimizing the error between the experimental and the curves obtained from the model. In contrast, the values of parameters (*a*, *b*, and *c*) of HEVC and VVC models were estimated according to the method proposed in Section 4.4, i.e. the parameters for HEVC and VVC are calculated from the parameters estimated for AVC. The mean relative approximation errors for individual frames of various types, as well as for total bitrate, are also presented in Appendix C. In Appendix D, the approximated curves are plotted using the parameters listed in Appendix C.

Table C.1: The model parameters and the average relative approximation error for I-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Poznan_CarPark	47943.17	1.30	10.09	3.47	2.94
FourPeople	6457.04	0.97	1.60	3.11	1.76
ChinaSpeed	9416.63	0.89	3.09	1.48	1.11
BQMall	11403.48	1.17	8.07	2.29	1.58
BlowingBubbles	6282.58	1.30	10.85	4.59	2.59

Table C.2: The model parameters and the average relative approximation error for P-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Poznan_CarPark	2477.32	1.03	-6.36	11.04	5.42
FourPeople	1715.78	1.10	-2.17	7.74	4.85
ChinaSpeed	30556.01	1.43	60.00	7.58	5.46
BQMall	6635.60	1.18	1.00	4.48	4.00
BlowingBubbles	4785.88	1.43	8.00	10.35	6.23

Table C.3: The model parameters and the average relative approximation error for B0-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Poznan_CarPark	1493.78	1.13	-6.00	3.05	2.47
FourPeople	418.00	1.02	1.00	5.93	3.40
ChinaSpeed	6635.00	1.23	1.00	2.95	1.84
BQMall	3010.00	1.18	1.00	3.47	2.37
BlowingBubbles	1798.3300	1.41	8.00	5.82	4.62

Table C.4: The model parameters and the average relative approximation error for B1-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Poznan_CarPark	1830.46	1.24	-3.00	5.28	4.85
FourPeople	269.58	1.047	-2.00	6.51	5.91
ChinaSpeed	18623.32	1.51	60.00	2.12	1.45
BQMall	1490.00	1.12	1.00	4.92	3.43
BlowingBubbles	1367.13	1.43	8.00	7.53	7.30

Table C.5: The model parameters and the average relative approximation error for B2-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Poznan_CarPark	2445.93	1.43	-3.00	10.03	6.46
FourPeople	147.00	1.00	10.00	8.92	6.08
ChinaSpeed	14643.15	1.53	60.00	2.91	2.85
BQMall	736.00	1.06	1.00	5.92	4.30
BlowingBubbles	674.54	1.39	8.00	5.89	5.10

Table C.6: The model parameters and the average relative approximation error for B3-frame size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Poznan_CarPark	833.00	1.15	-4.00	4.81	3.02
FourPeople	128.39	0.92	1.02	4.86	3.25
ChinaSpeed	7830.00	1.39	1.00	2.85	1.96
BQMall	1171.00	1.10	1.00	5.22	3.70
BlowingBubbles	585.48	1.29	8.00	7.31	6.78

Table C.7: The model parameters and the average relative approximation error for GOP-level size for the AVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Poznan_CarPark	1782.00	1.11	1.03	2.96	1.85
FourPeople	210.06	0.79	-1.36	1.76	1.31
ChinaSpeed	357.05	0.71	-3.04	4.32	3.16
BQMall	152.12	0.68	-3.08	3.26	3.00
BlowingBubbles	150.00	0.97	-3.60	2.70	2.07

Table C.8: The average relative approximation error for I-frame size for the HEVC and VVC codecs.

Sequence	HEVC		VVC	
	Relative error [%]		Relative error [%]	
	Mean	St.dev	mean	std. dev
Poznan_CarPark	2.5	1.56	3.62	2.58
FourPeople	4.01	2.32	3.64	2.39
ChinaSpeed	7.89	2.02	5.85	1.55
BQMall	6.84	2.10	8.37	1.79
BlowingBubbles	10.63	1.80	12.80	3.63

Table C.9: The average relative approximation error for P-frame size for the HEVC and VVC codecs.

Sequence	HEVC		VVC	
	Relative error [%]		Relative error [%]	
	Mean	St.dev	mean	std. dev
Poznan_CarPark	9.36	8.45	7.36	6.66
FourPeople	16.93	2.57	13.77	3.72
ChinaSpeed	12.86	4.92	13.58	4.19
BQMall	5.15	2.70	6.08	3.64
BlowingBubbles	14.84	3.20	18.60	3.09

Table C.10: The average relative approximation error for B0-frame size for the HEVC and VVC codecs.

Sequence	HEVC		VVC	
	Relative error [%]		Relative error [%]	
	Mean	St.dev	mean	std. dev
Poznan_CarPark	17.79	6.45	21.69	6.83
FourPeople	12.18	9.08	14.60	10.54
ChinaSpeed	12.62	8.82	12.81	9.10
BQMall	7.59	2.30	8.55	1.30
BlowingBubbles	28.76	2.03	33.83	2.38

Table C.11: The average relative approximation error for B1-frame size for the HEVC and VVC codecs.

Sequence	HEVC		VVC	
	Relative error [%]		Relative error [%]	
	Mean	St.dev	mean	std. dev
Poznan_CarPark	22.63	5.01	23.41	8.40
FourPeople	15.27	1.97	18.33	2.28
ChinaSpeed	22.89	8.10	24.91	6.25
BQMall	14.84	4.98	12.81	5.74
BlowingBubbles	27.72	3.34	34.97	4.68

Table C.12: The average relative approximation error for B2-frame size for the HEVC and VVC codecs.

Sequence	HEVC		VVC	
	Relative error [%]		Relative error [%]	
	Mean	St.dev	mean	std. dev
Poznan_CarPark	31.77	5.96	35.13	6.74
FourPeople	13.58	13.90	21.12	13.37
ChinaSpeed	28.18	8.42	32.70	6.78
BQMall	16.90	9.26	15.50	9.80
BlowingBubbles	36.61	1.46	46.31	6.79

Table C.13: The average relative approximation error for B3-frame size for the HEVC and VVC codecs.

Sequence	HEVC		VVC	
	Relative error [%]		Relative error [%]	
	Mean	St.dev	mean	std. dev
Poznan_CarPark	24.34	13.62	24.14	16.94
FourPeople	21.71	9.24	12.72	8.36
ChinaSpeed	49.55	11.98	52.72	12.43
BQMall	10.95	6.94	12.05	6.48
BlowingBubbles	36.93	1.39	50.12	10.22

Table C.14: The average relative approximation error for GOP-level size for the HEVC and VVC codecs.

Sequence	HEVC		VVC	
	Relative error [%]		Relative error [%]	
	Mean	St.dev	mean	std. dev
Poznan_CarPark	8.15	5.51	13.98	7.03
FourPeople	7.82	3.95	12.15	5.32
ChinaSpeed	22.14	8.97	22.78	12.48
BQMall	12.72	7.32	19.53	10.57
BlowingBubbles	23.04	7.32	26.36	11.39

Appendix D

Related to Chapter Four

In Appendix D, the approximate curves for the verification set of video sequences for HEVC and VVC coding were plotted using the parameters listed in Appendix C for the AVC model and the method proposed in Section 4.4, i.e. the parameters for HEVC and VVC are calculated from the parameters estimated for AVC.. The figures in Appendix D can show the difference between the experimental and approximated curves.

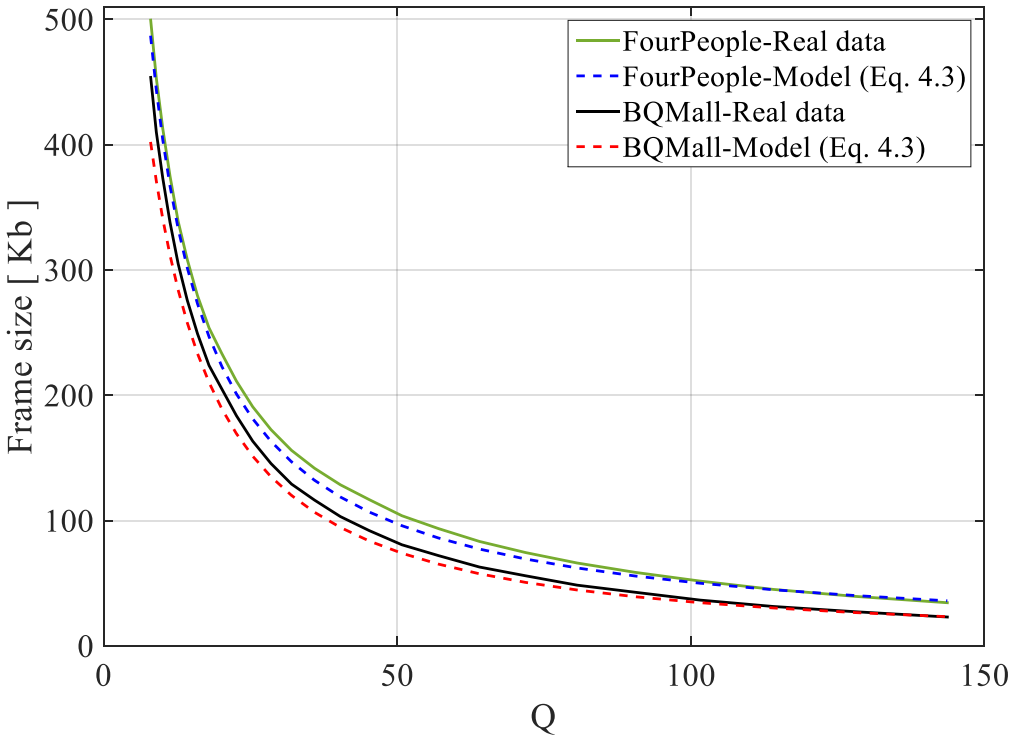


Fig. D.1. The experimental and approximated curves to frame size for I-frame for *FourPeople* and *BQMall* sequences for the HEVC coding.

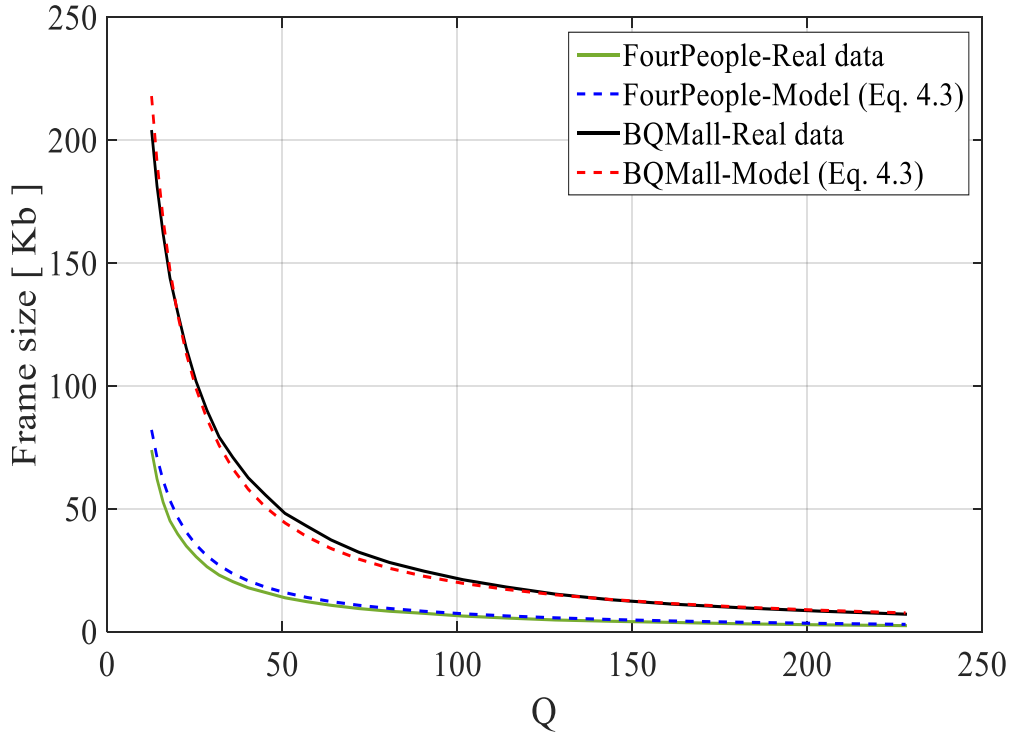


Fig. D.2. The experimental and approximated curves to frame size for P-frame for *FourPeople* and *BQMall* sequences for the HEVC coding.

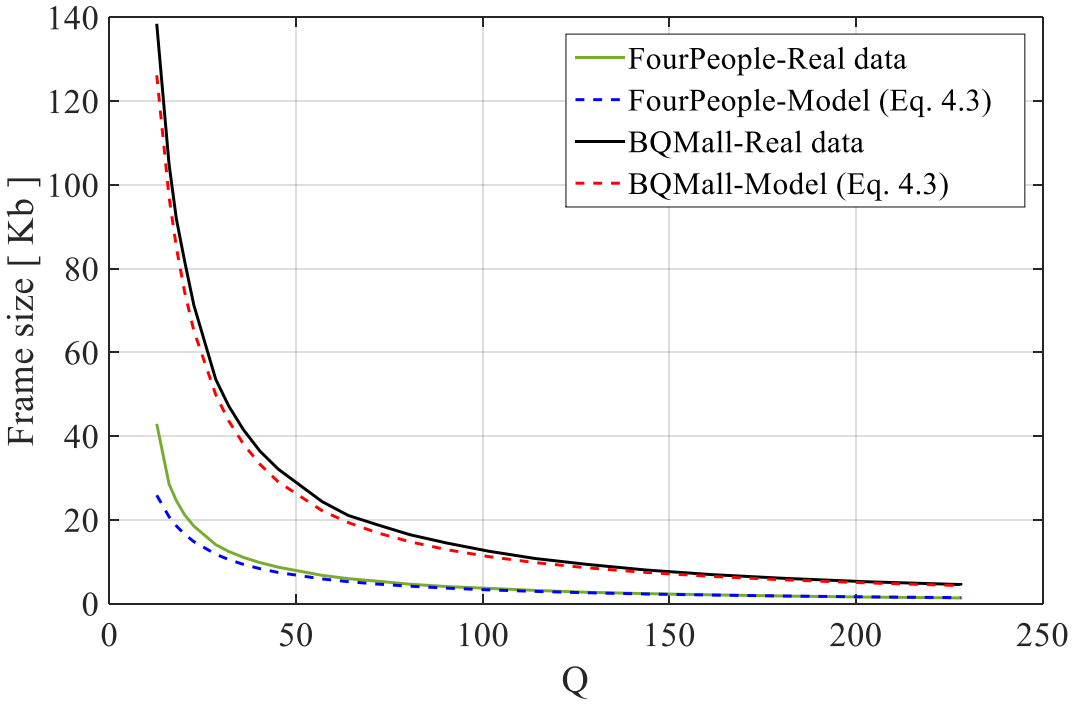


Fig. D.3. The experimental and approximated curves to frame size for B0-frame for *FourPeople* and *BQMall* sequences for the HEVC coding.

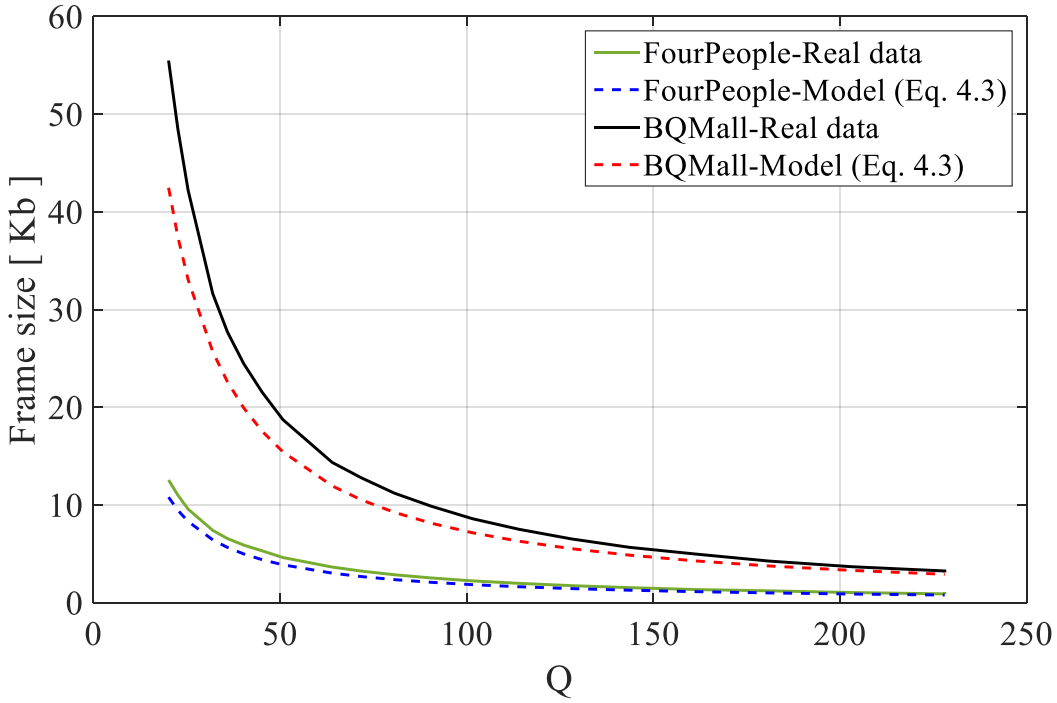


Fig. D.4. The experimental and approximated curves to frame size for B1-frame for *FourPeople* and *BQMall* sequences for the HEVC coding.

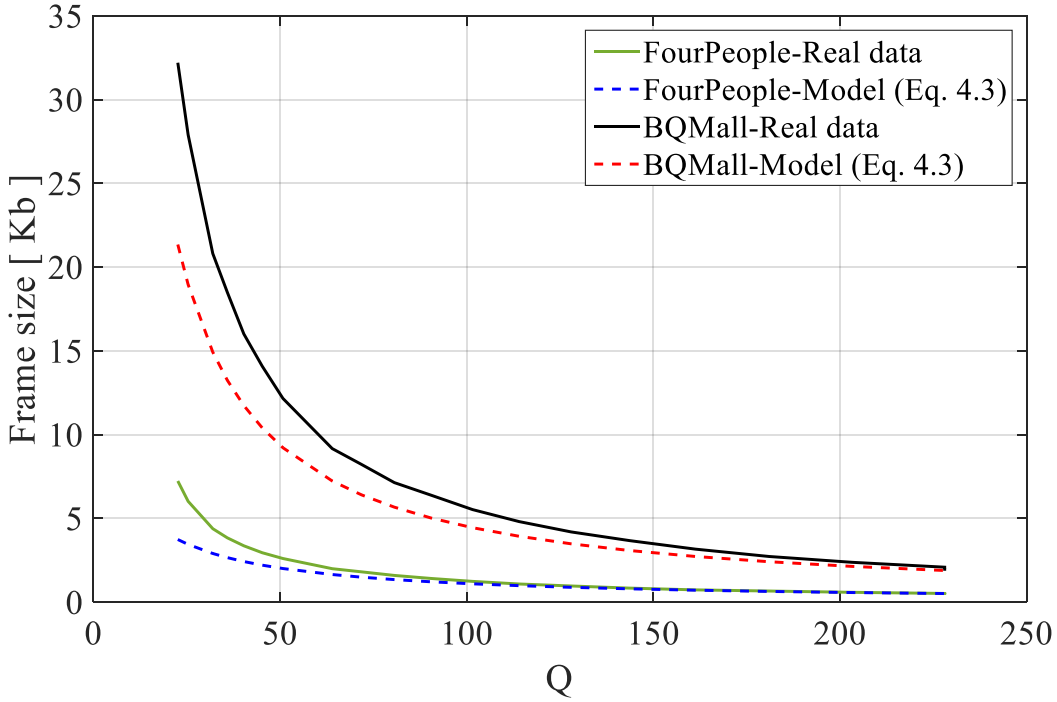


Fig. D.5. The experimental and approximated curves to frame size for B2-frame for *FourPeople* and *BQMall* sequences for the HEVC coding.

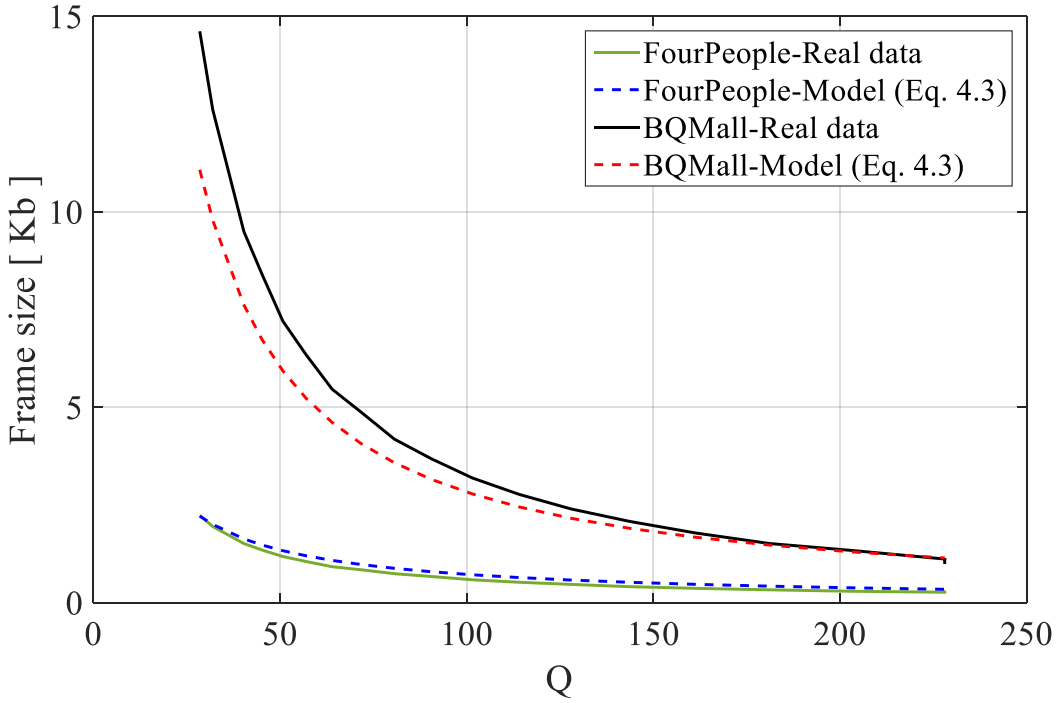


Fig. D.6. The experimental and approximated curves to frame size for B3-frame for *FourPeople* and *BQMall* sequences for the HEVC coding.

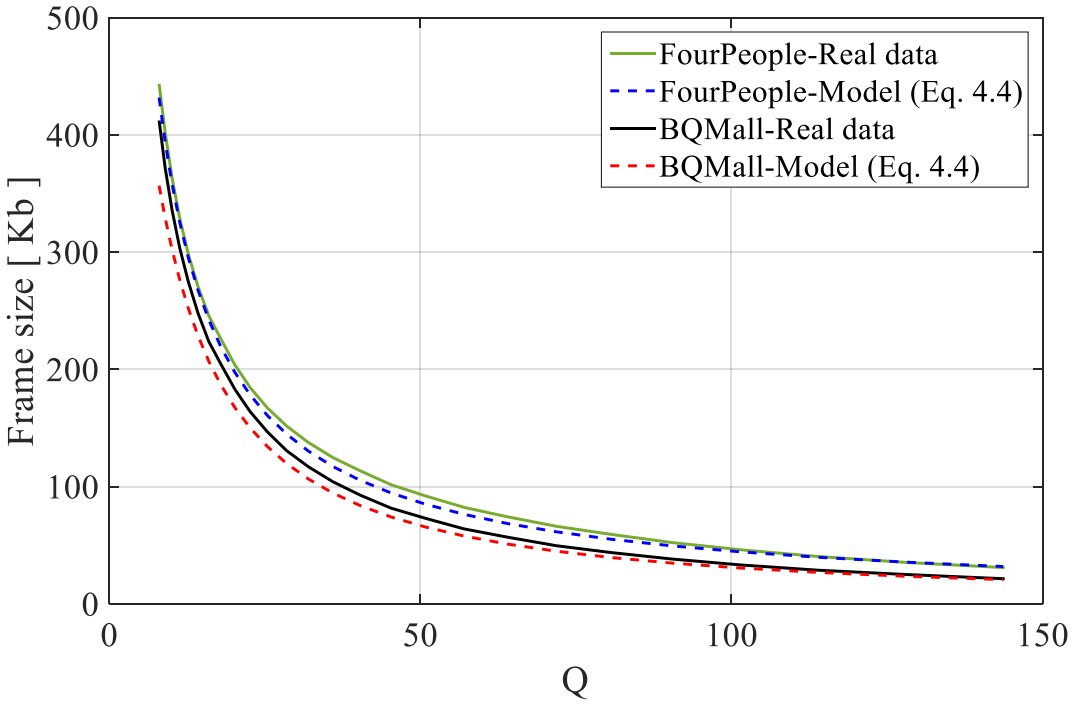


Fig. D.7. The experimental and approximated curves to frame size for I-frame for *FourPeople* and *BQMall* sequences for the VVC coding.

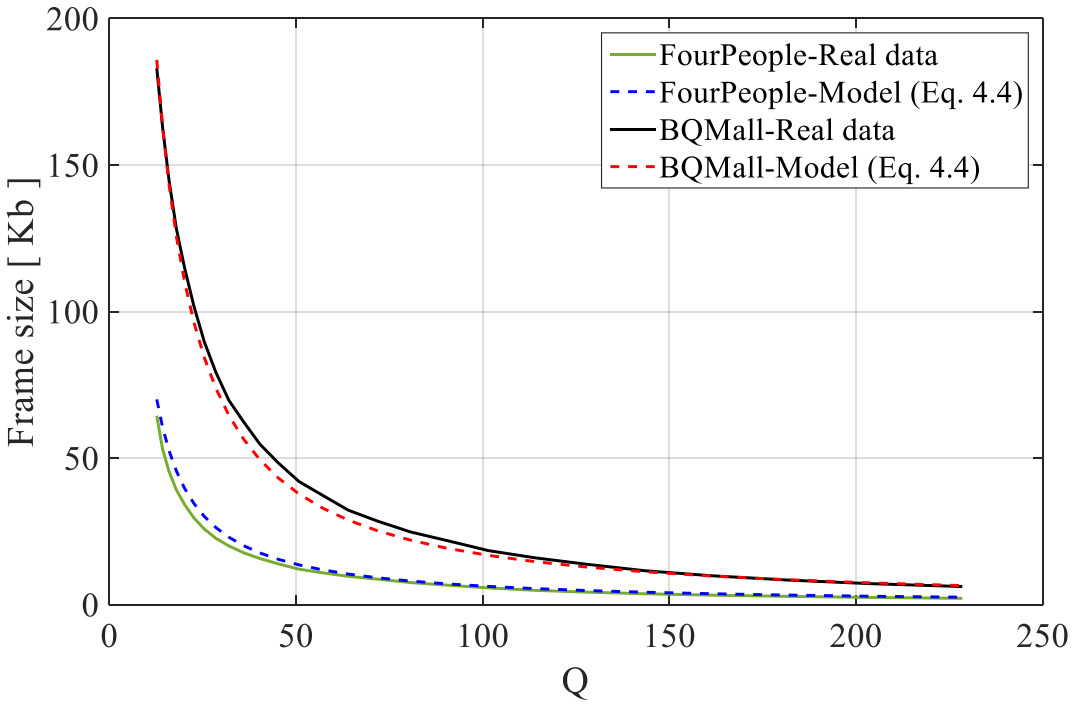


Fig. D.8. The experimental and approximated curves to frame size for P-frame for *FourPeople* and *BQMall* sequences for the VVC coding.

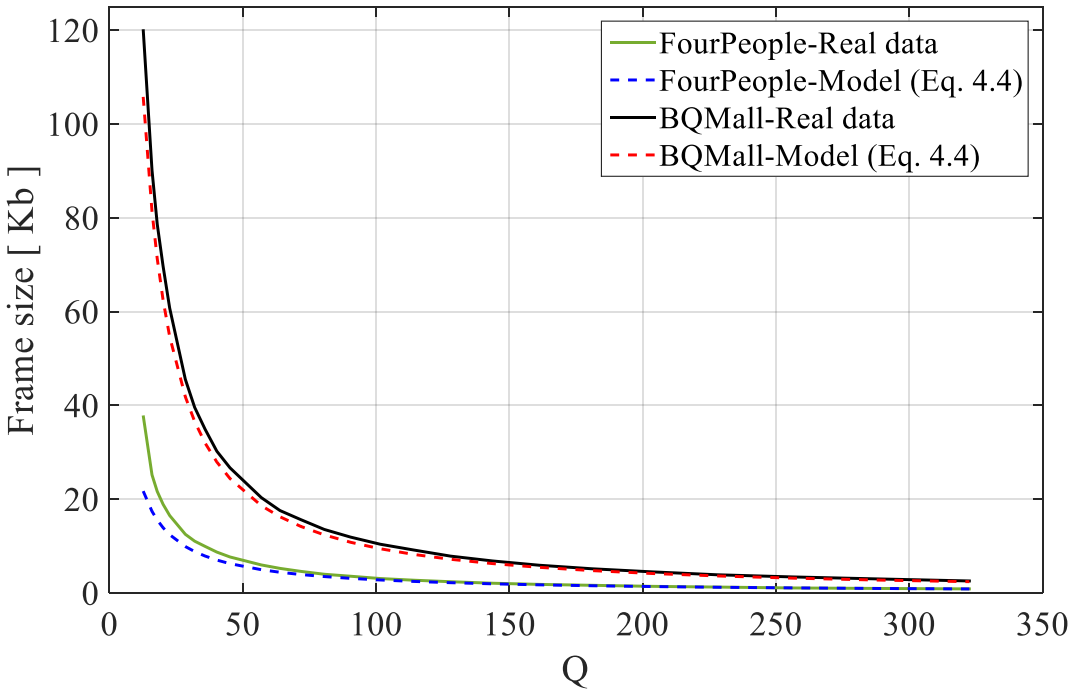


Fig. D.9. The experimental and approximated curves to frame size for B0-frame for *FourPeople* and *BQMall* sequences for the VVC coding.

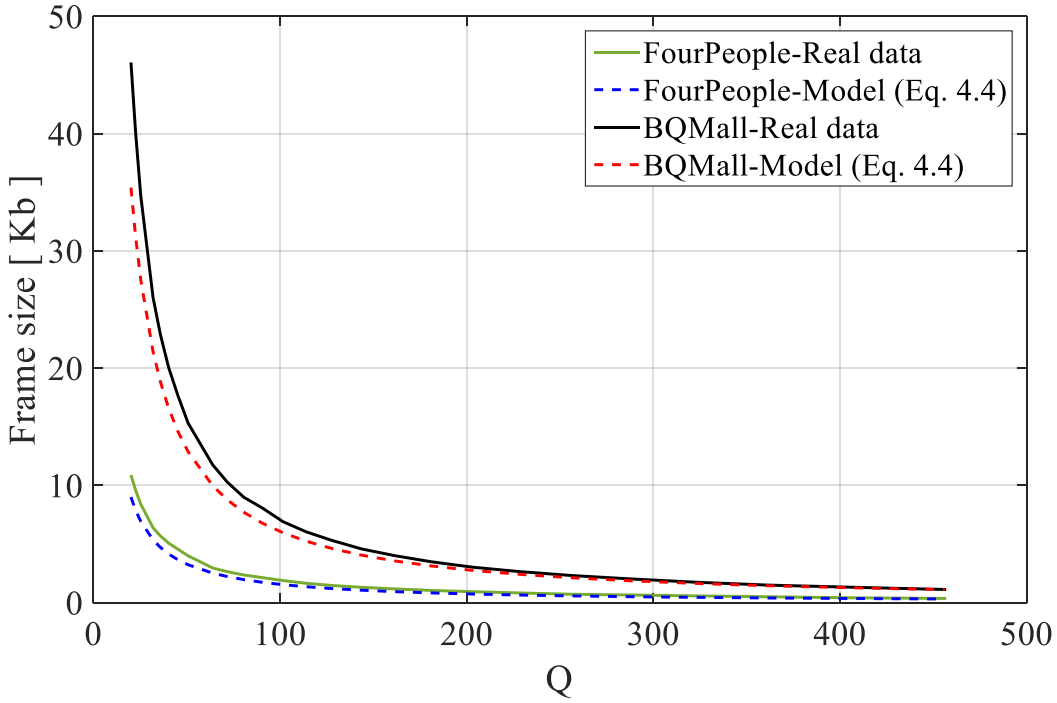


Fig. D.10. The experimental and approximated curves to frame size for B1-frame for *FourPeople* and *BQMall* sequences for the VVC coding.

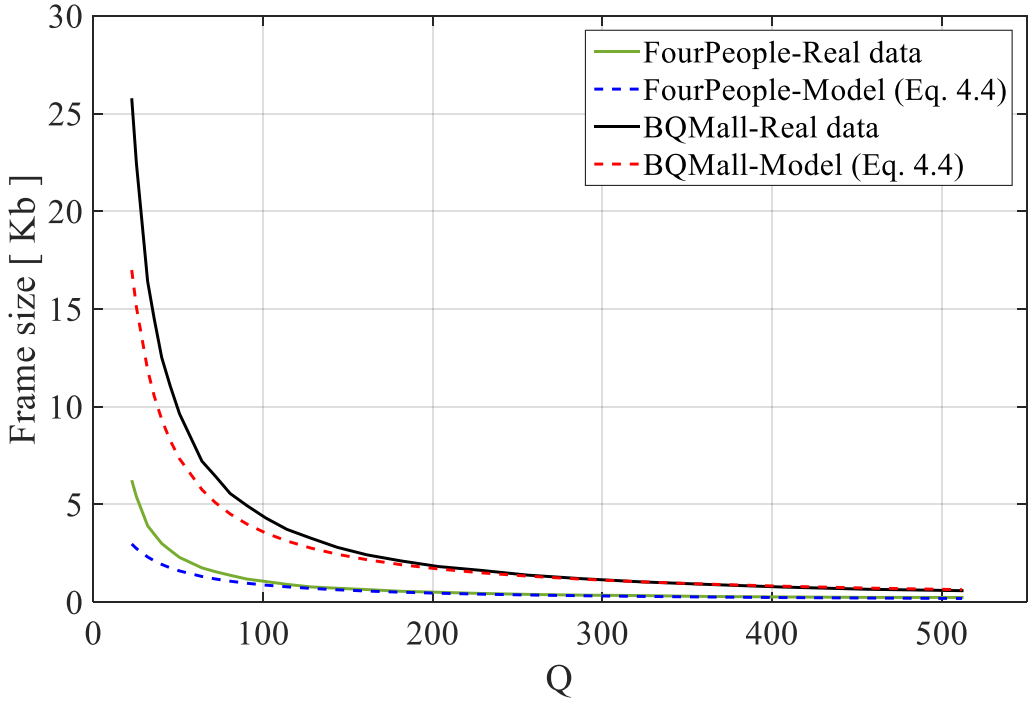


Fig. D.11. The experimental and approximated curves to frame size for B2-frame for *FourPeople* and *BQMall* sequences for the VVC coding.

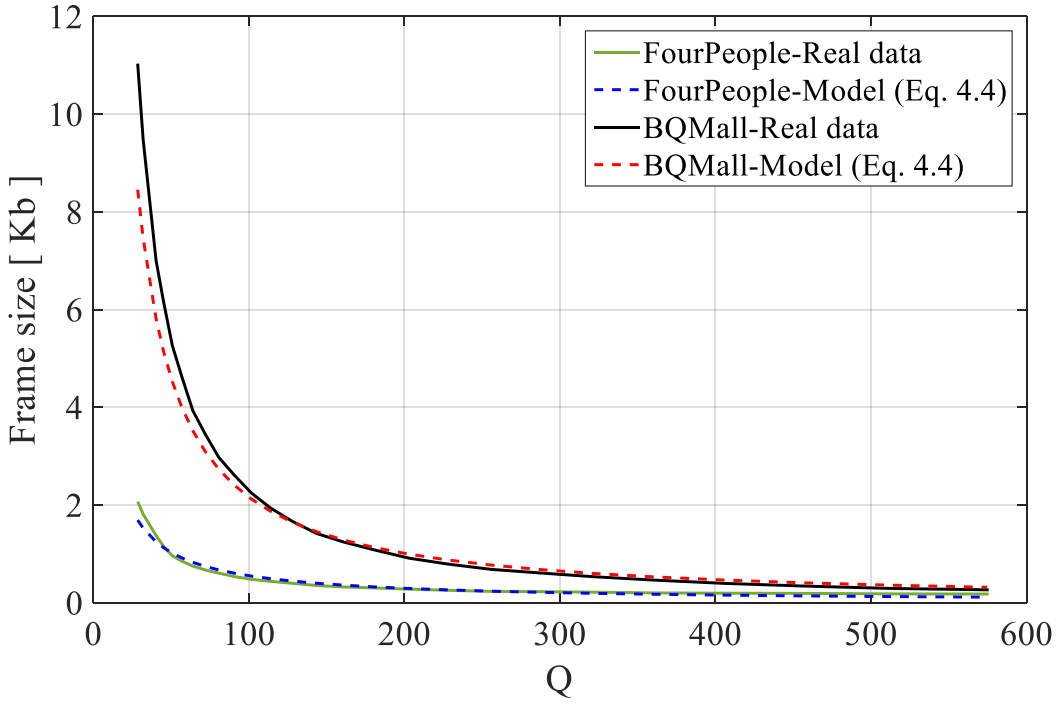


Fig. D.12. The experimental and approximated curves to frame size for B3-frame for *FourPeople* and *BQMall* sequences for the VVC coding.

Appendix E

Related to Chapter Four

In Appendix E, the encoding times are shown for two examples of test sequences: *BQMall* (832 × 480) and *FourPeople* (1280 × 720). The encoding times are estimated for the respective reference software pieces. The values can be considered for a very rough comparison of AVC, HEVC, and VVC encoders. The relation between the encoding times may differ substantially for the practical encoder implementations using different sequences.

The results have been obtained using a personal computer with an Intel Core i7-5820K processor, 15MB cache, up to 3.60GHz frequency, 64 GB RAM, 6 total cores, 12 total threads, and 157 MB/sec disk read speed, 130 MB/sec disk write speed, 75.3MB/sec disk read/write speed.

Table E.1: A comparison of encoding times (T) for AVC, HEVC, and VVC

Sequence	*QP*	Encoding time (sec.)			The ratio of the encoding times	
		AVC	HEVC	VVC	$\frac{HEVC}{AVC}$	$\frac{VVC}{AVC}$
BQMall	26	612.313	1041.270	5270.000	1.70	8.61
	30	604.135	943.530	4219.230	1.56	6.98
	34	585.655	897.377	3419.330	1.53	5.84
	38	557.595	863.171	2657.470	1.55	4.77
	43	533.670	816.296	1997.010	1.53	3.74
	48	494.801	790.603	1260.230	1.60	2.55
FourPeople	26	646.919	1017.980	3720.050	1.57	5.75
	30	604.822	976.708	2681.060	1.61	4.43
	34	462.039	949.857	2028.990	2.06	4.39
	38	436.912	930.472	1469.240	2.13	3.36
	43	419.372	915.262	1040.900	2.18	2.48
	48	417.856	911.437	940.559	2.18	2.25

Appendix F

Related to Chapter Five

In Appendix F, red dots represent the results of virtual view synthesis with the use of views and depth maps encoded according to Fig. 3.2 with all the possible combinations of quantization parameters for views (*QP*) and depth maps (*QD*), each (*QP*, *QD*) pair corresponds to Bitrate(*QP*, *QD*) and PSNR(*QP*, *QD*). The blue line represents optimum (*QP*, *QD*) pairs. The optimum (*QP*, *QD*) pairs form the best R-D (rate-distortion) curve. These curve is the envelope over the cloud of PSNR-bitrate points that.

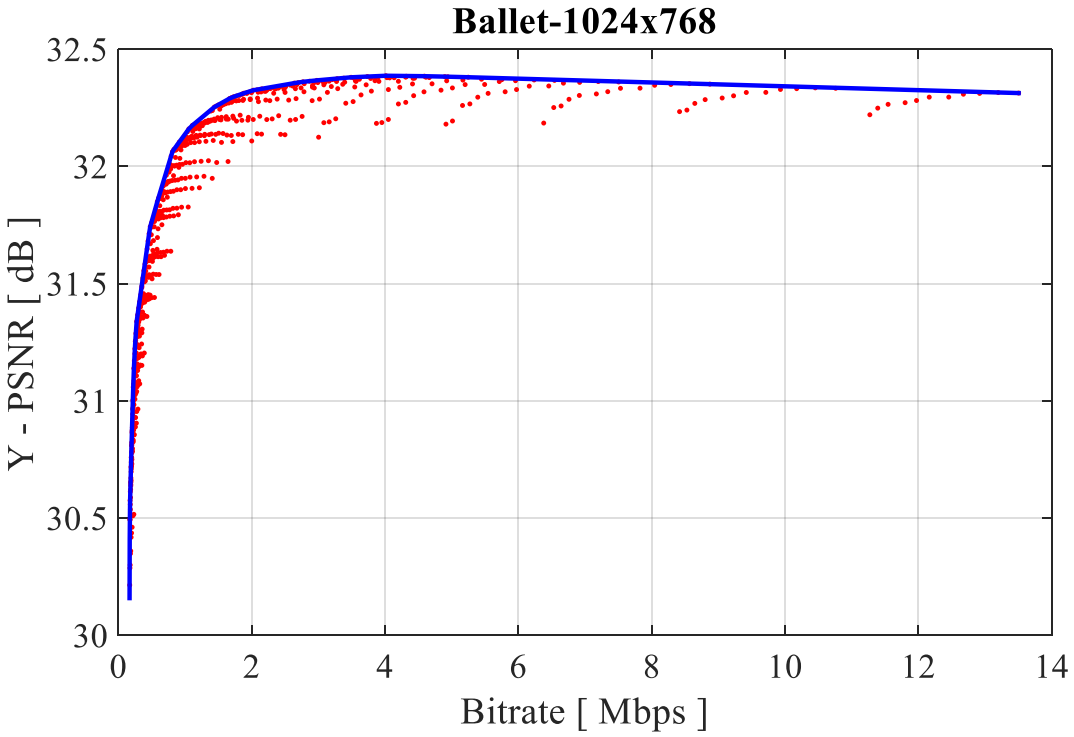


Fig. F.1. The best R-D curve calculated from (*QP*, *QD*) pairs for the HEVC codec for the *Ballet* sequence.

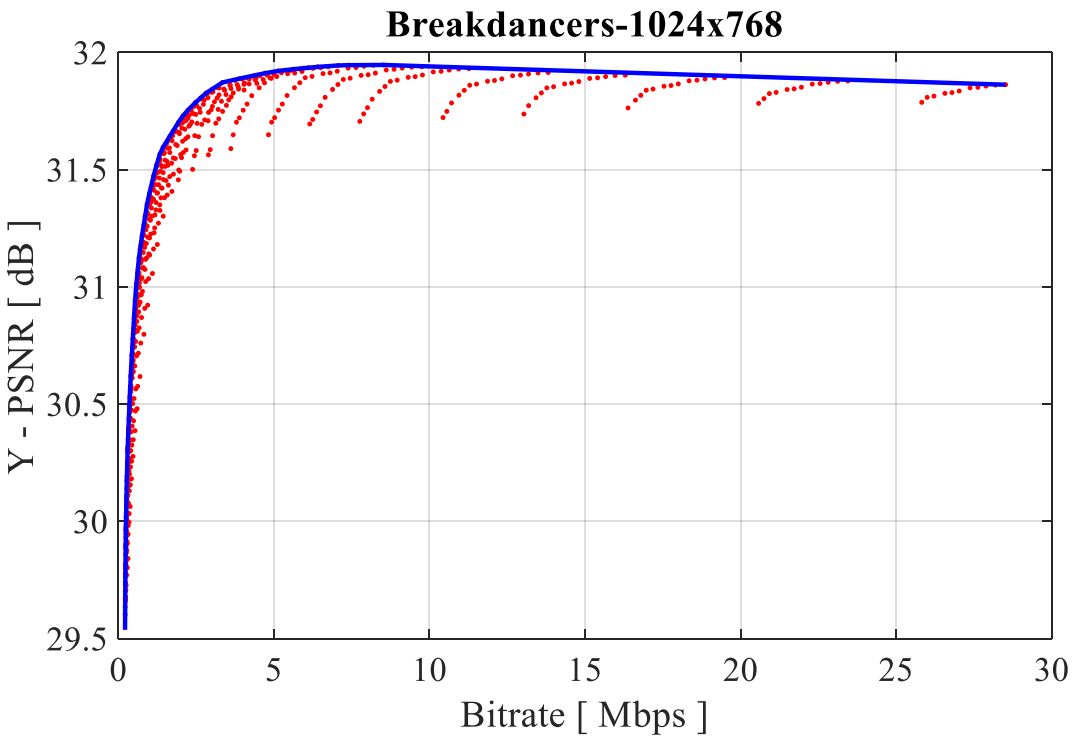


Fig. F.2. The best R-D curve calculated from (*QP, QD)* pairs for the HEVC codec for the *Breakdancers* sequence.

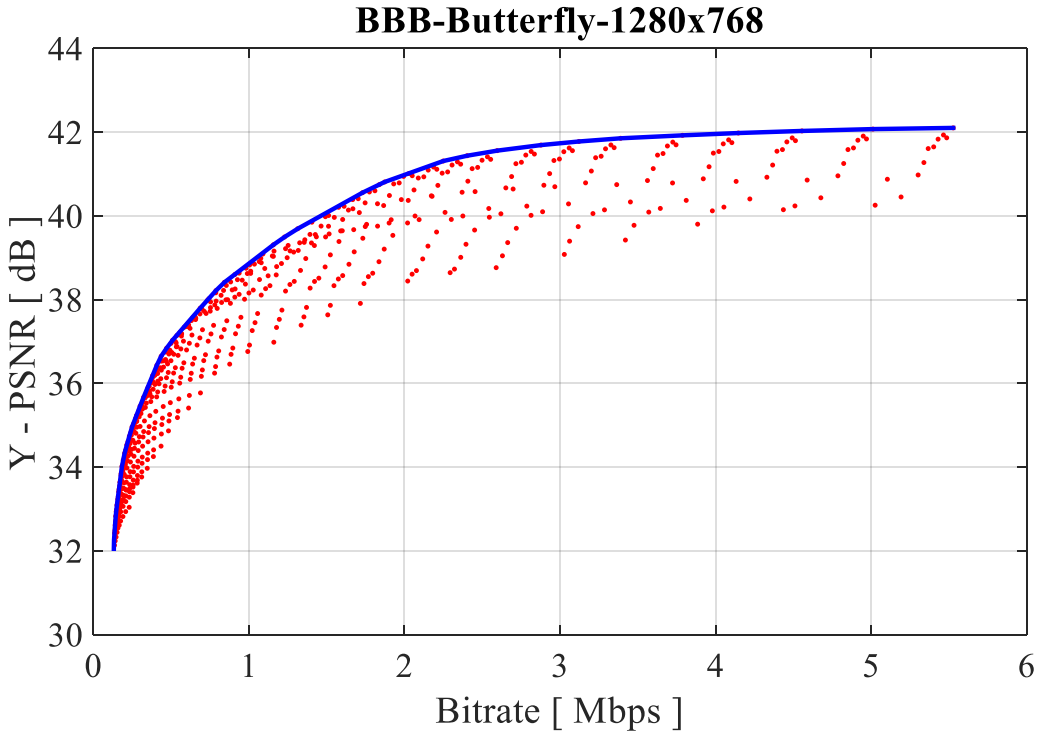


Fig. F.3. The best R-D curve calculated from (*QP, QD)* pairs for the HEVC codec for the *BBB.Butterfly* sequence.

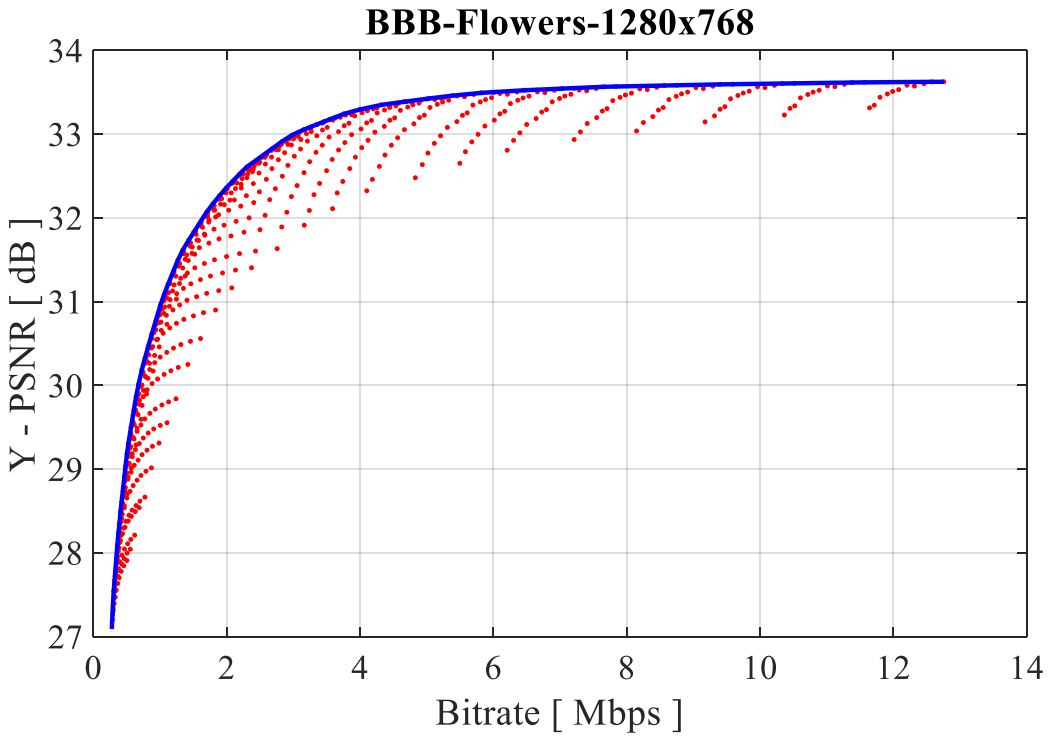


Fig. F.4. The best R-D curve calculated from (*QP, QD)* pairs for the HEVC codec for the *BBB.Flowers* sequence.

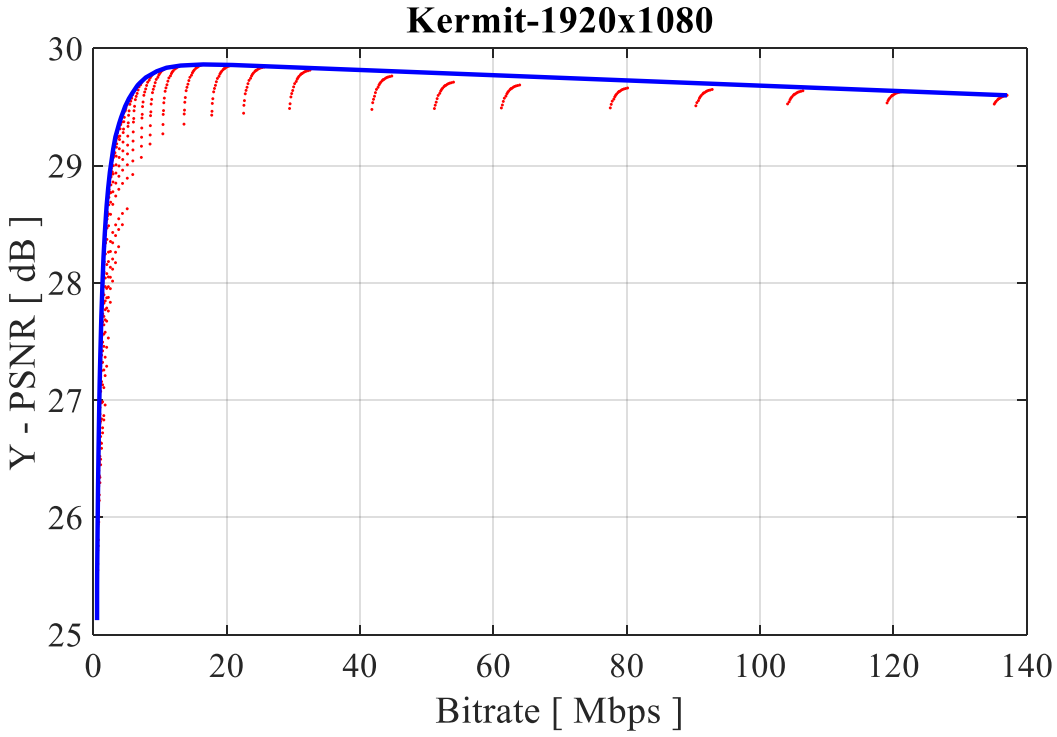


Fig. F.5. The best R-D curve calculated from (*QP, QD)* pairs for the HEVC codec for the *Kermit* sequence.

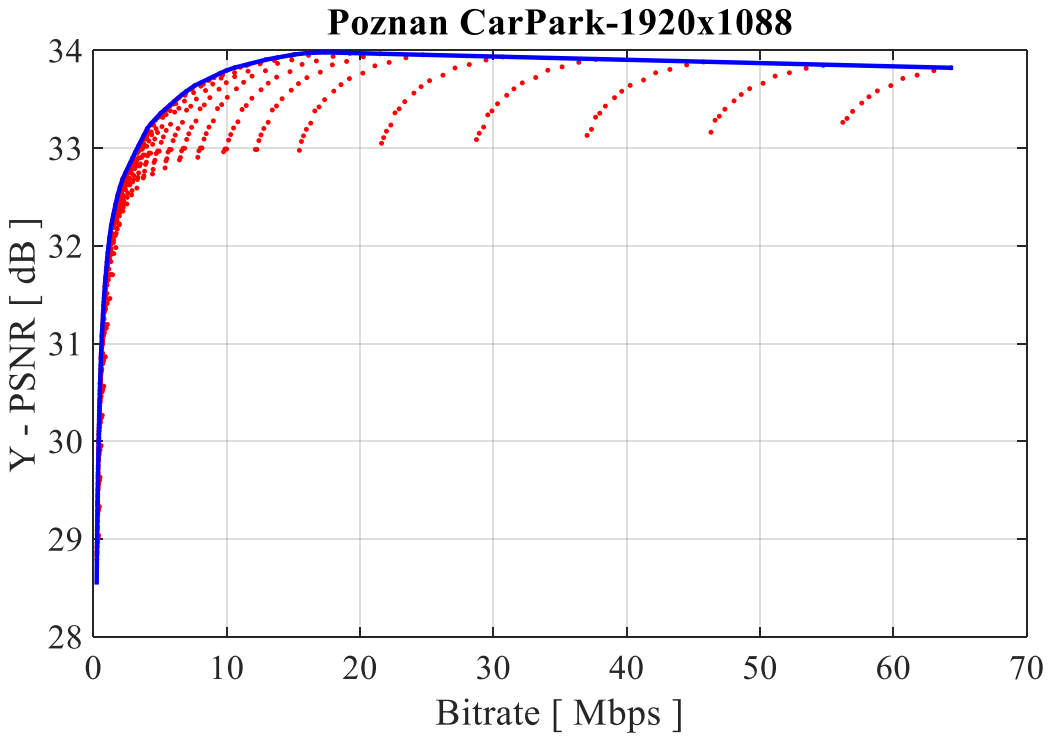


Fig. F.6. The best R-D curve calculated from (*QP, QD)* pairs for the HEVC codec for the *Poznan_CarPark* sequence.

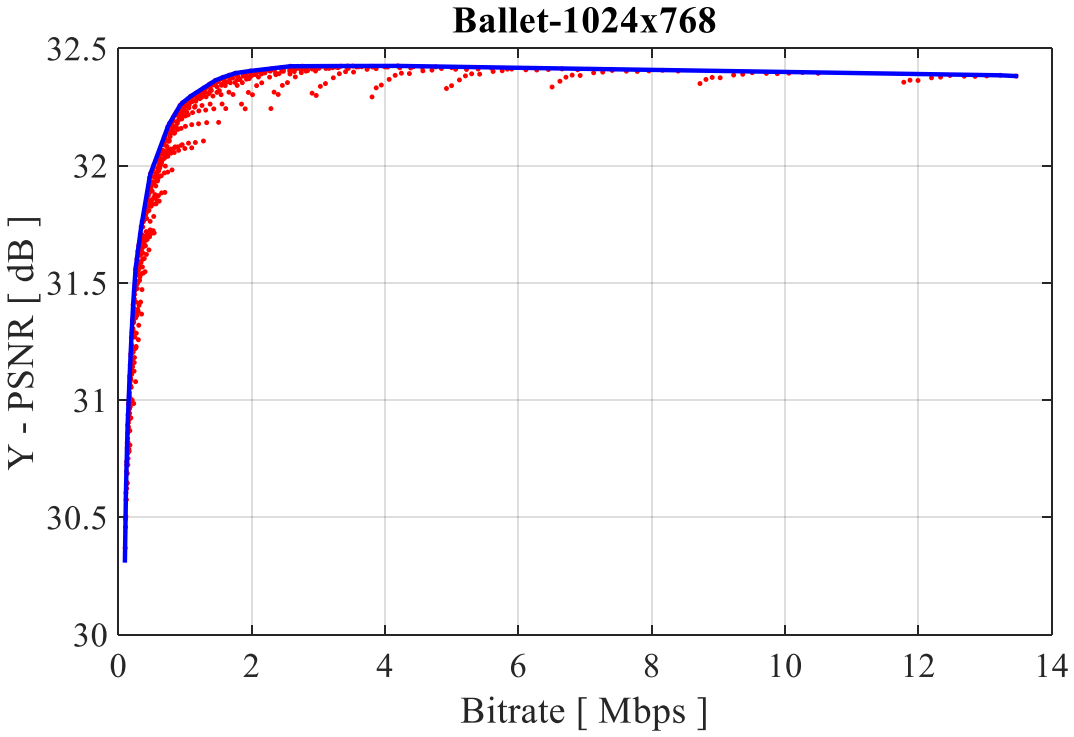


Fig. F.7. The best R-D curve calculated from (*QP, QD)* pairs for the VVC codec for the *Ballet* sequence.

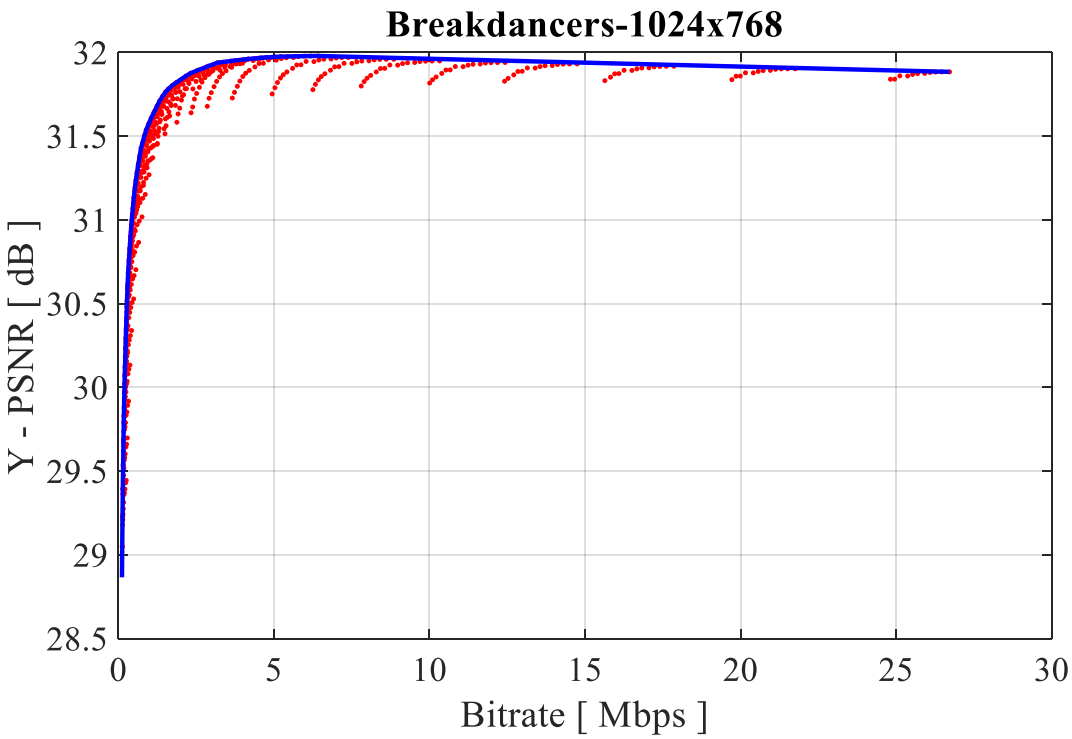


Fig. F.8. The best R-D curve calculated from (*QP*, *QD)* pairs for the VVC codec for the *Breakdancers* sequence.

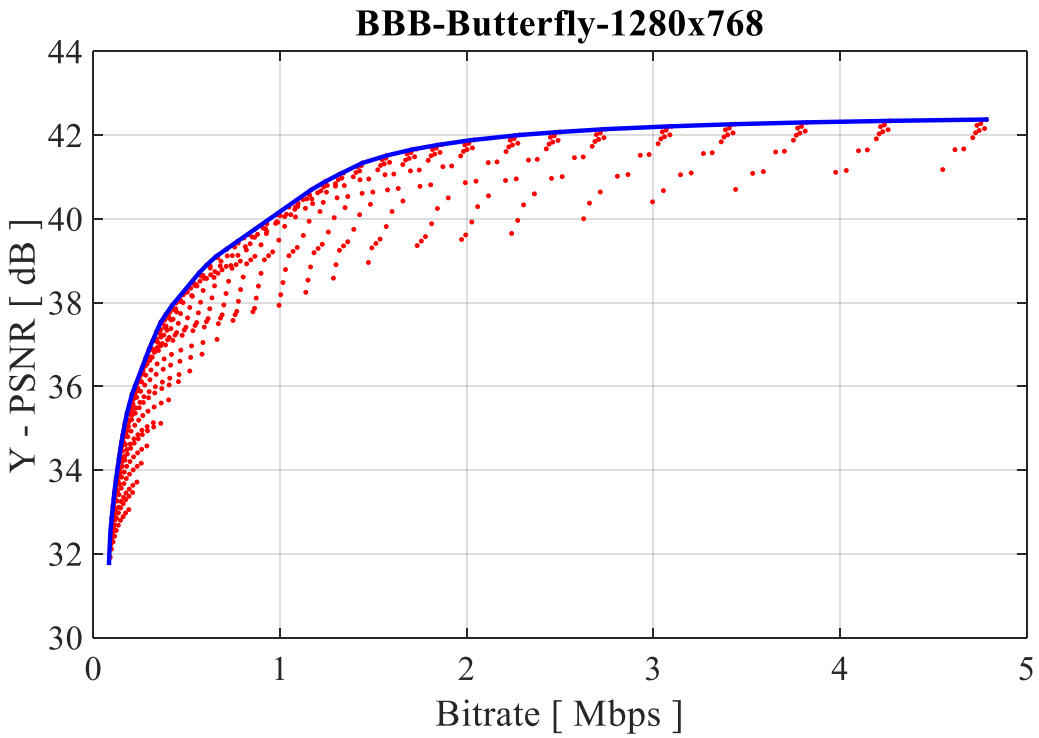


Fig. F.9. The best R-D curve calculated from (*QP*, *QD)* pairs for the VVC codec for the *BBB.Butterfly* sequence.

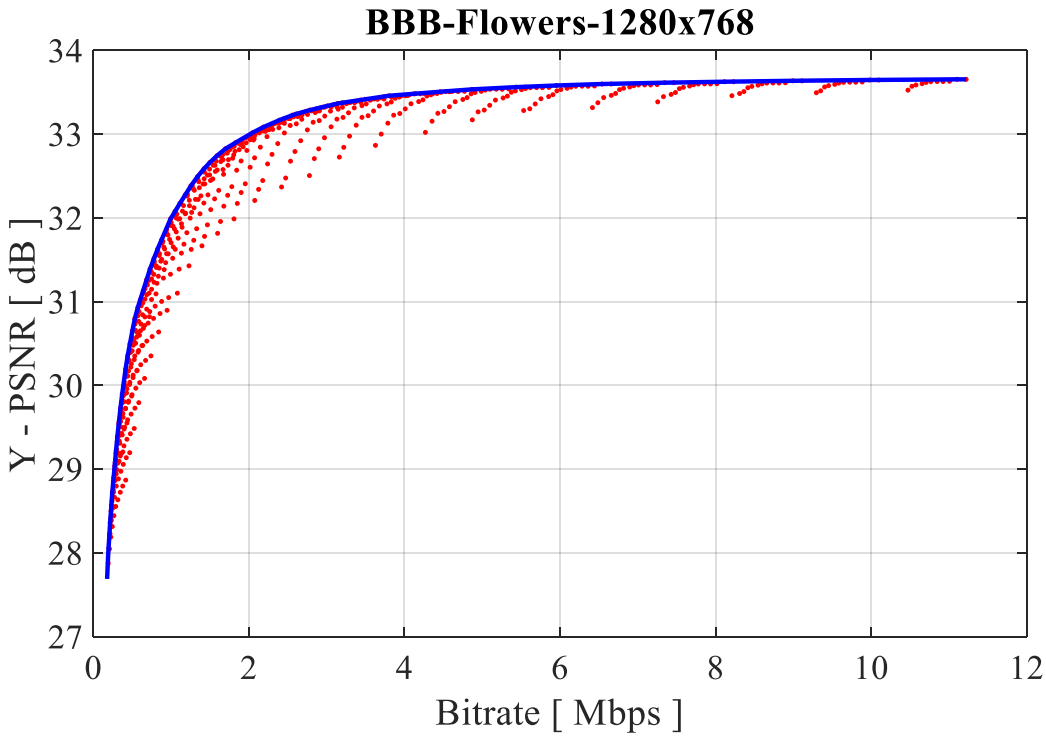


Fig. F.10. The best R-D curve calculated from (*QP, QD)* pairs for the VVC codec for the *BBB.Flowers* sequence.

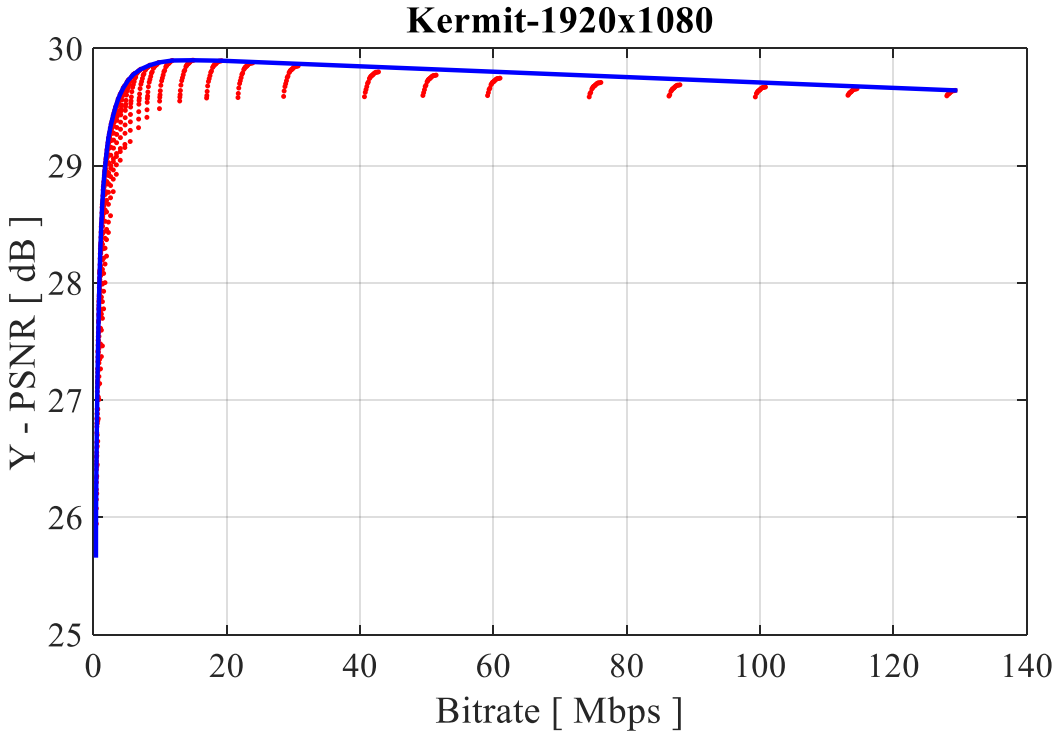


Fig. F.11. The best R-D curve calculated from (*QP, QD)* pairs for the VVC codec for the *Kermit* sequence.

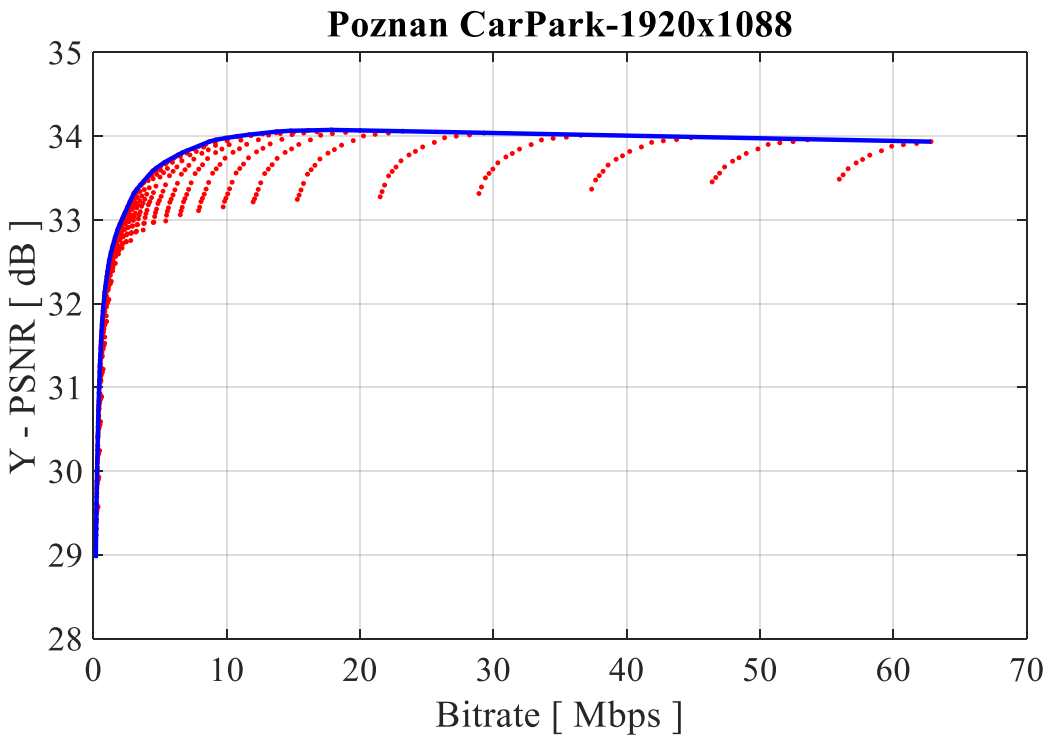


Fig. F.12. The best R-D curve calculated from (*QP, QD)* pairs for the VVC codec for the *Poznan_CarPark* sequence.

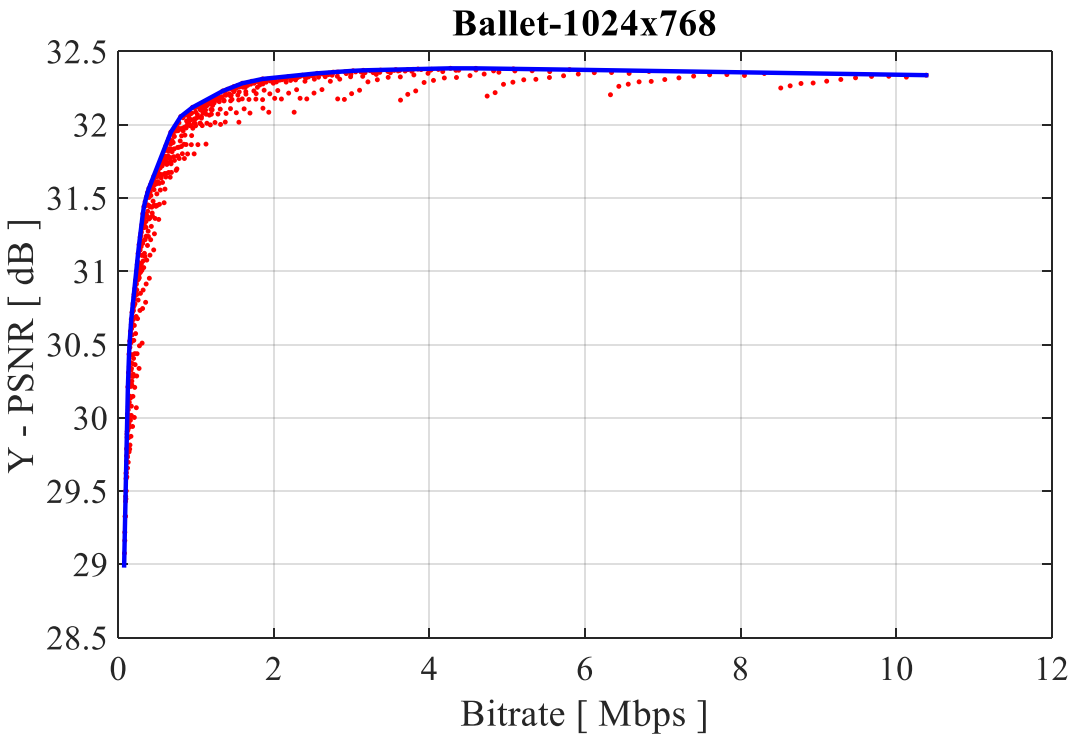


Fig. F.13. The best R-D curve calculated from (*QP, QD)* pairs for the MV-HEVC codec for the *Ballet* sequence.

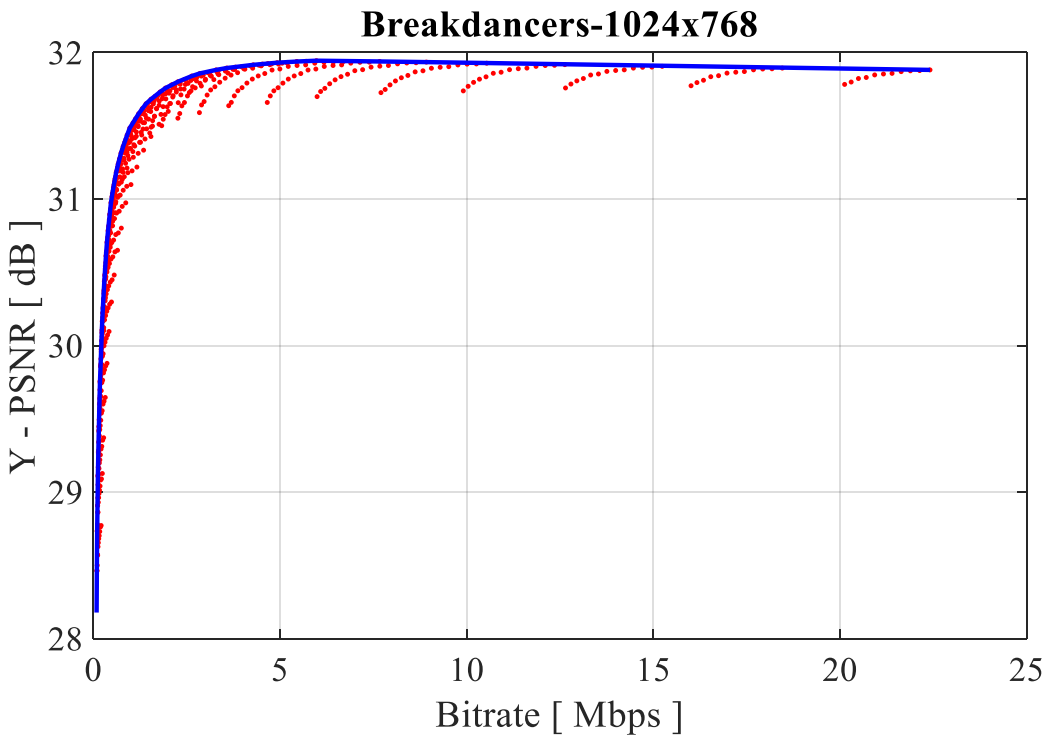


Fig. F.14. The best R-D curve calculated from (*QP, QD)* pairs for the MV-HEVC codec for the *Breakdancers* sequence.

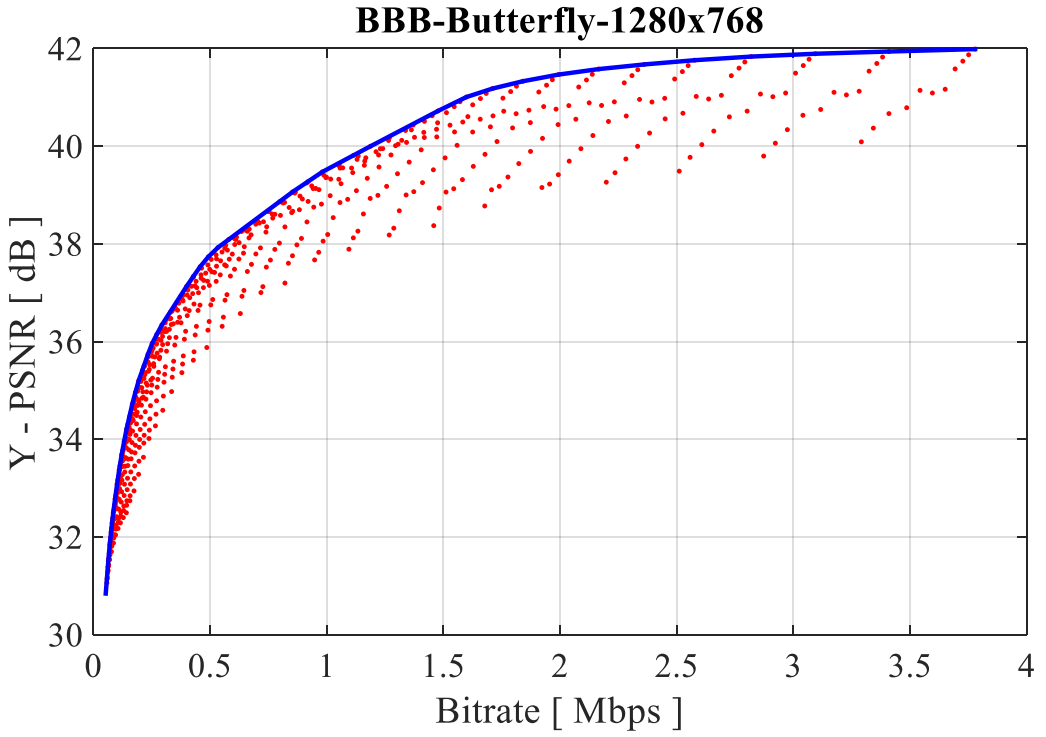


Fig. F.15. The best R-D curve calculated from (*QP, QD)* pairs for the MV-HEVC codec for the *BBB.Butterfly* sequence.

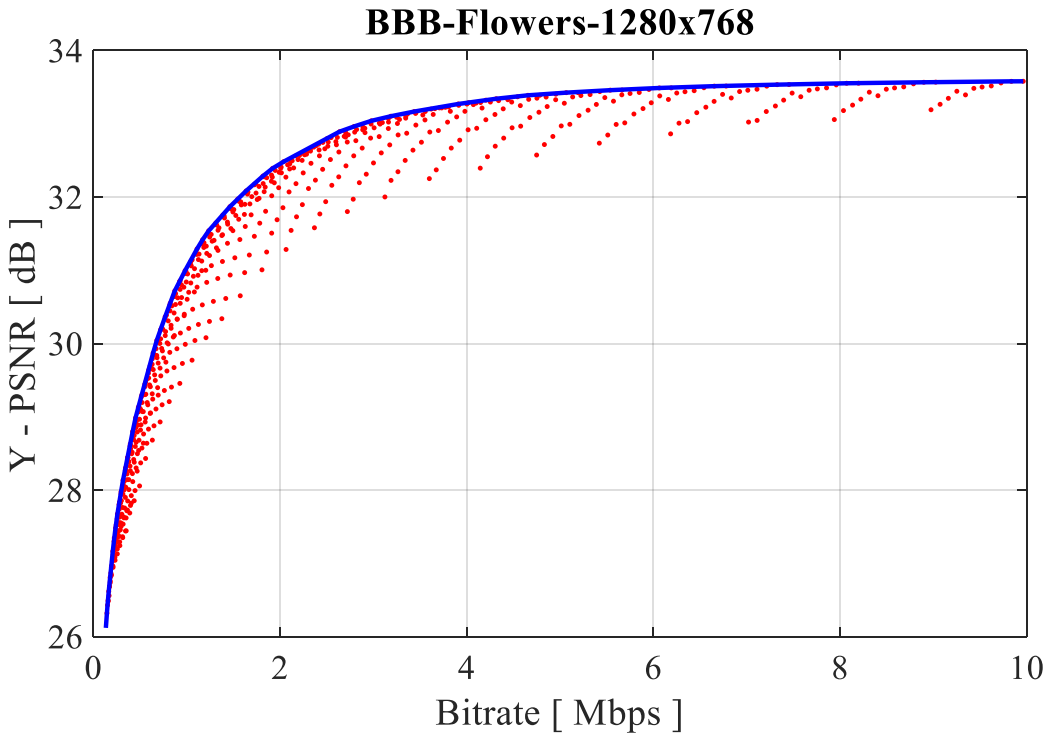


Fig. F.16. The best R-D curve calculated from (*QP, QD)* pairs for the MV-HEVC codec for the *BBB.Flowers* sequence.

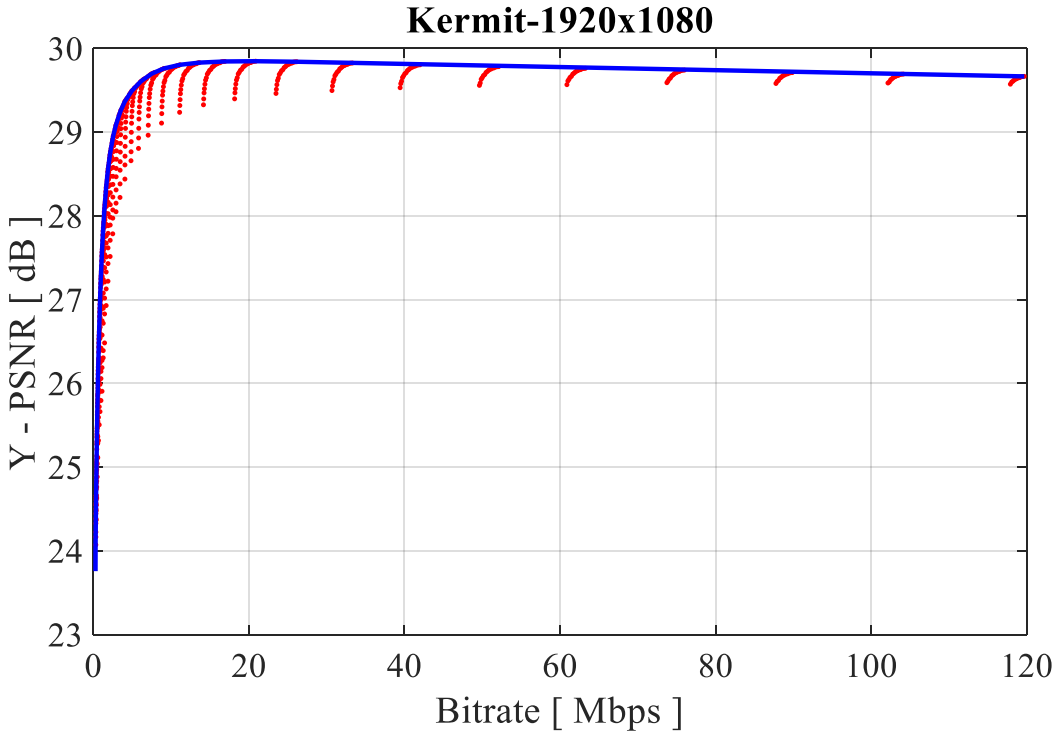


Fig. F.17. The best R-D curve calculated from (*QP, QD)* pairs for the MV-HEVC codec for the *Kermit* sequence.

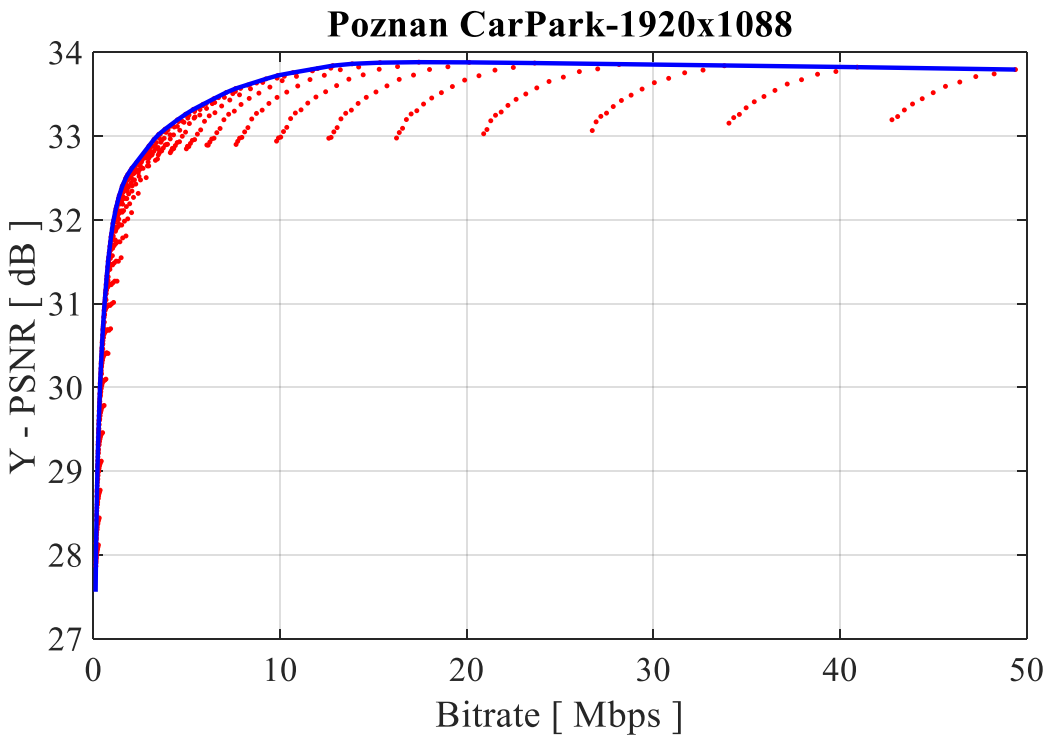


Fig. F.18. The best R-D curve calculated from (*QP, QD)* pairs for the MV-HEVC codec for the *Poznan_CarPark* sequence.

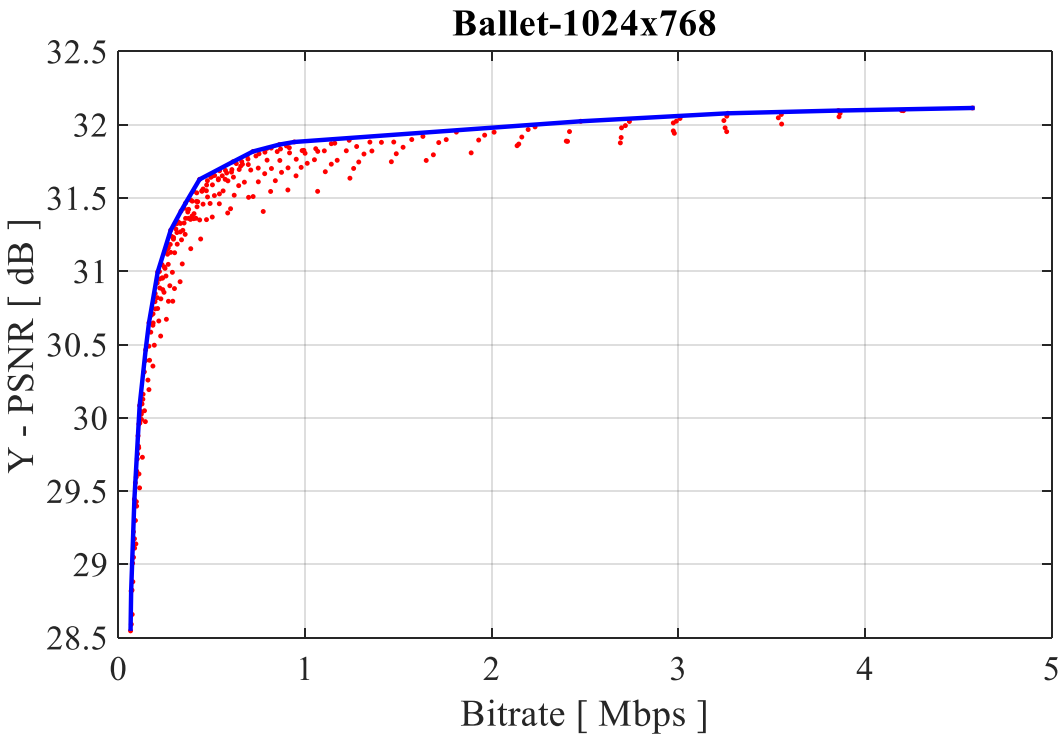


Fig. F.19. The best R-D curve calculated from (*QP, QD)* pairs for the 3D-HEVC codec for the *Ballet* sequence.

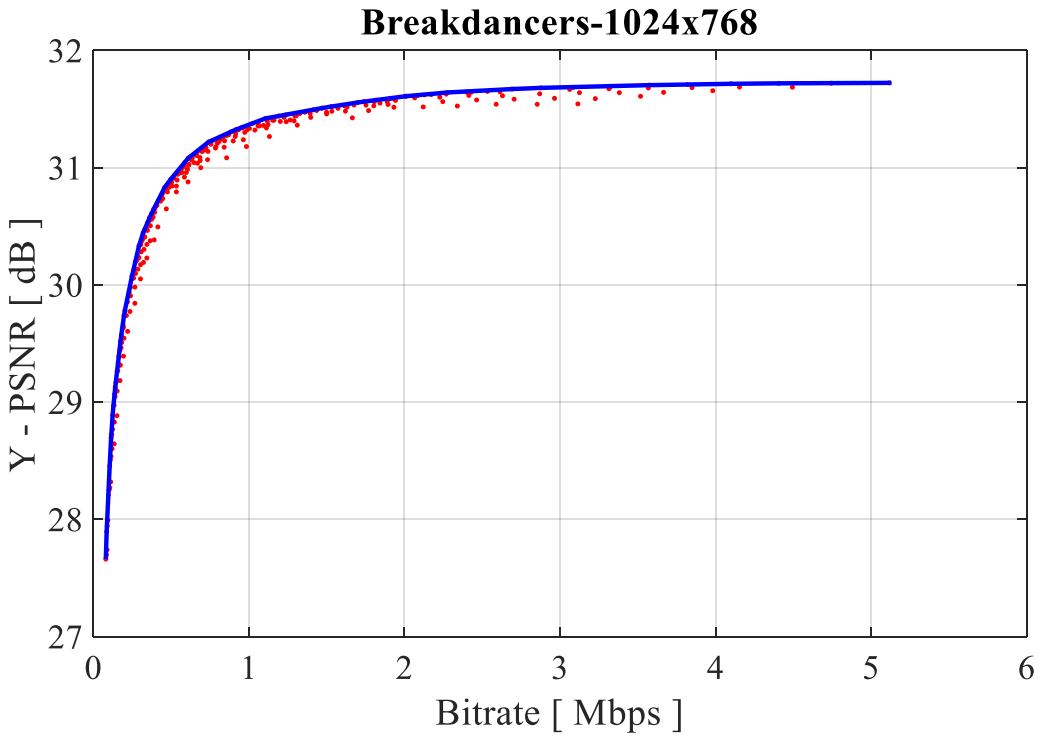


Fig. F.20. The best R-D curve calculated from (*QP*, *QD)* pairs for the 3D-HEVC codec for the *Breakdancers* sequence.

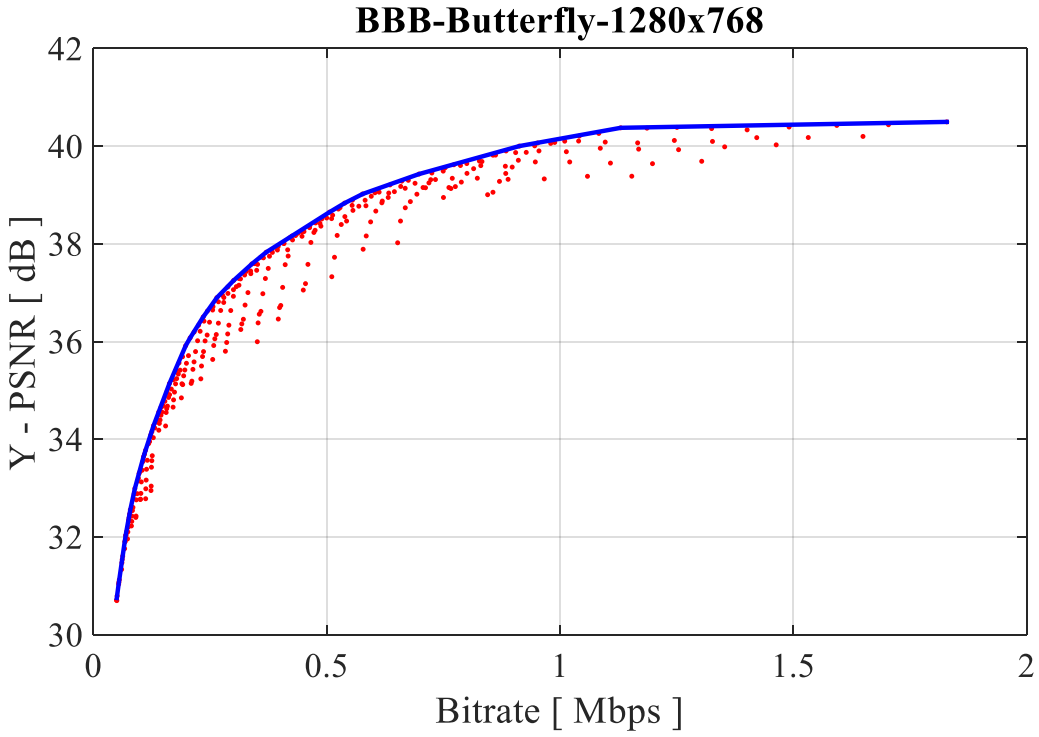


Fig. F.21. The best R-D curve calculated from (*QP*, *QD)* pairs for the 3D-HEVC codec for the *BBB.Butterfly* sequence.

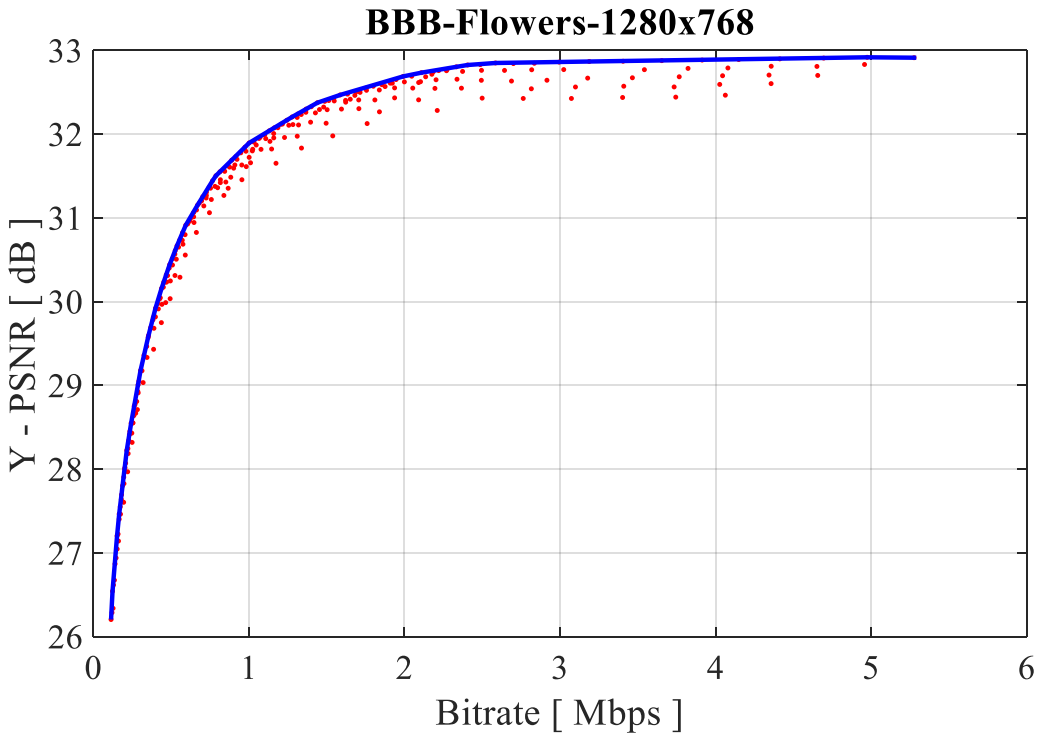


Fig. F.22. The best R-D curve calculated from (*QP*, *QD)* pairs for the 3D-HEVC codec for the *BBB.Flowers* sequence.

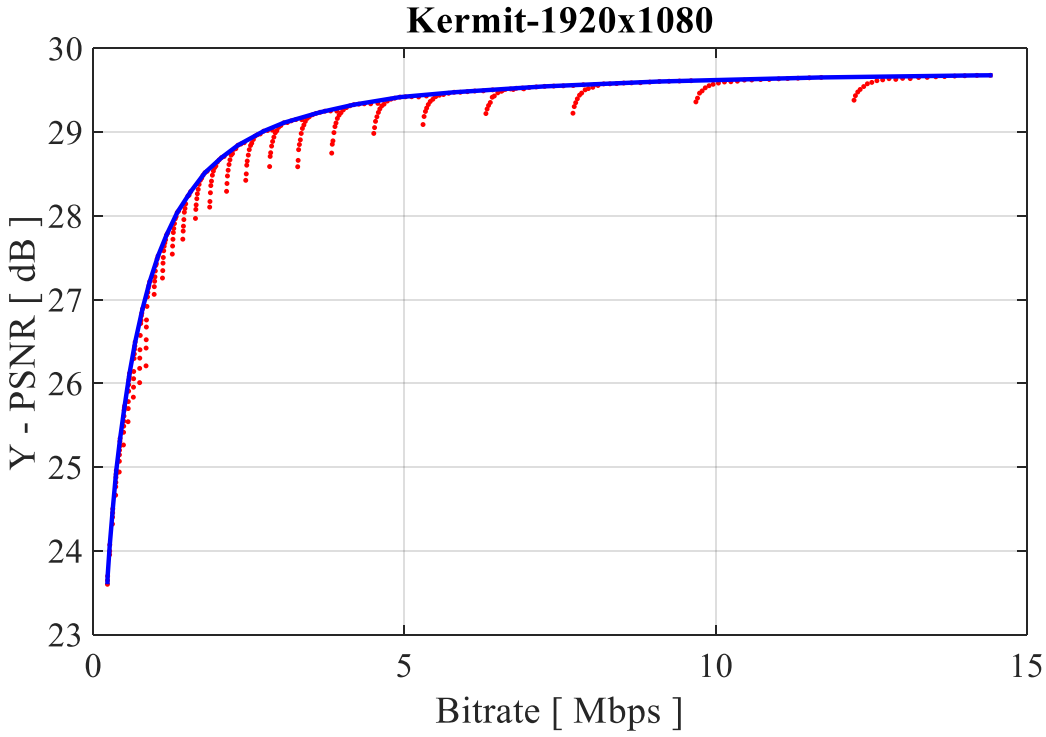


Fig. F.23. The best R-D curve calculated from (*QP*, *QD)* pairs for the 3D-HEVC codec for the *Kermit* sequence.

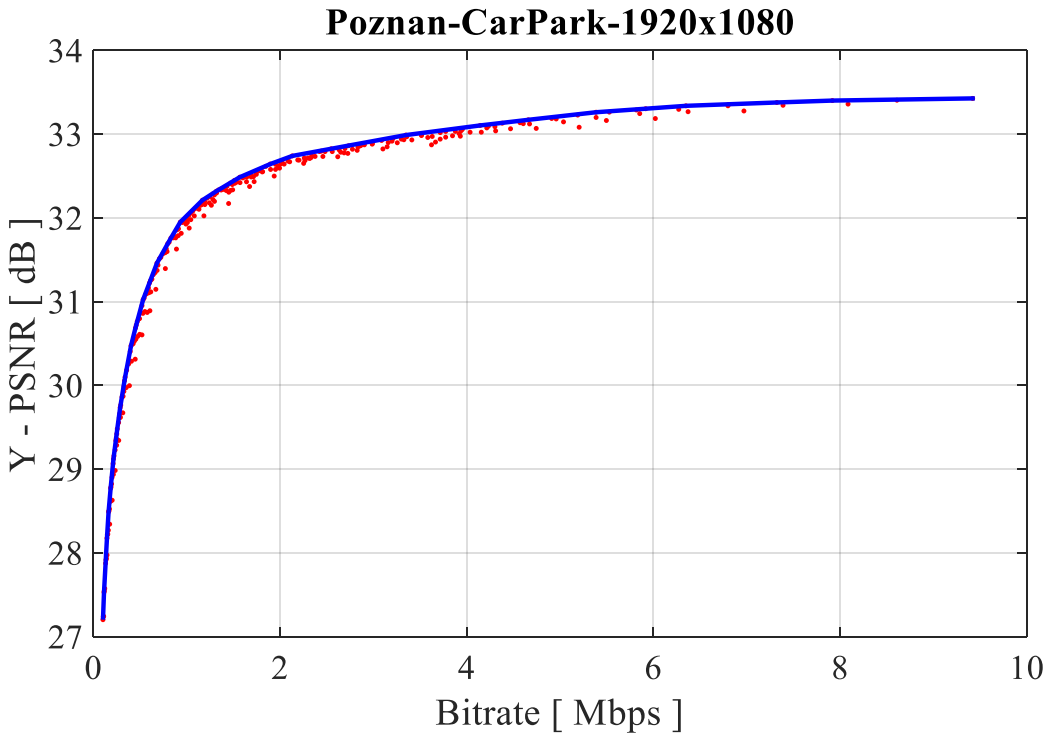


Fig. F.24. The best R-D curve calculated from (*QP*, *QD)* pairs for the 3D-HEVC codec for the *Poznan_CarPark* sequence.

Appendix G

Related to Chapter Five

In Appendix G, the approximate relationship between *QP* and *QD* for the optimum pairs for training sequences (shown in Table 3.2) is shown. The blue dots represent optimum (*QP*, *QD*) pairs for training sequences, while the red line represents the line obtained from a first-order polynomial regression.

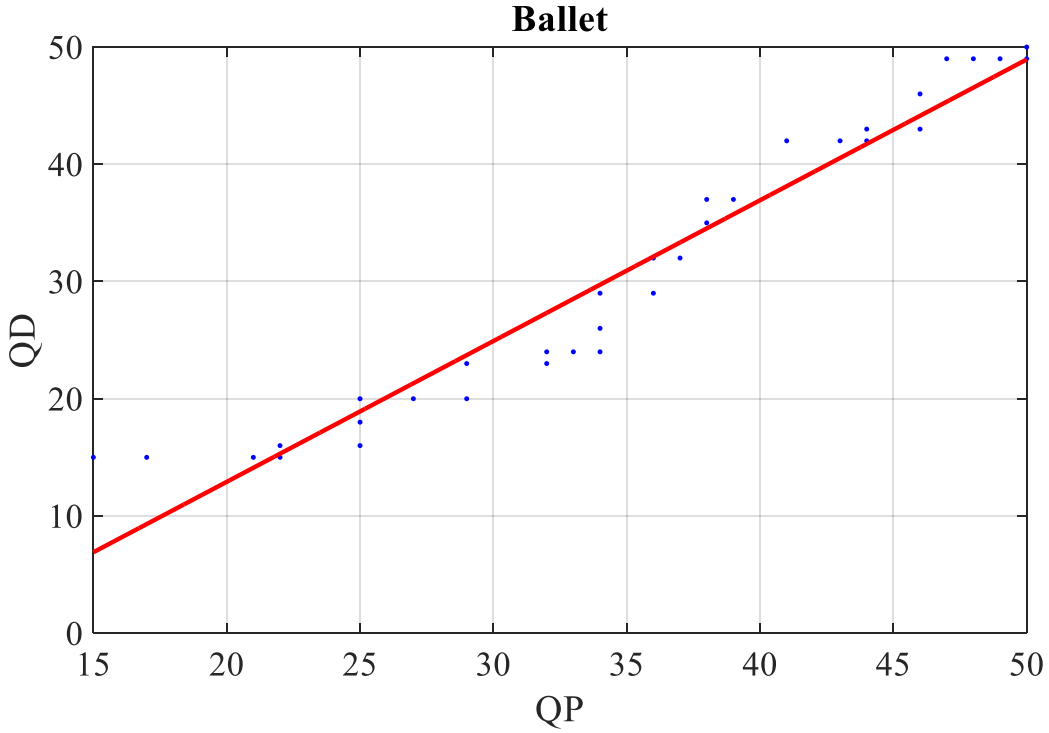


Fig. G.1. The approximate relationship between *QD* and *QP* for the optimum pairs for HEVC encoding for the *Ballet* sequence with the use of the linear model.

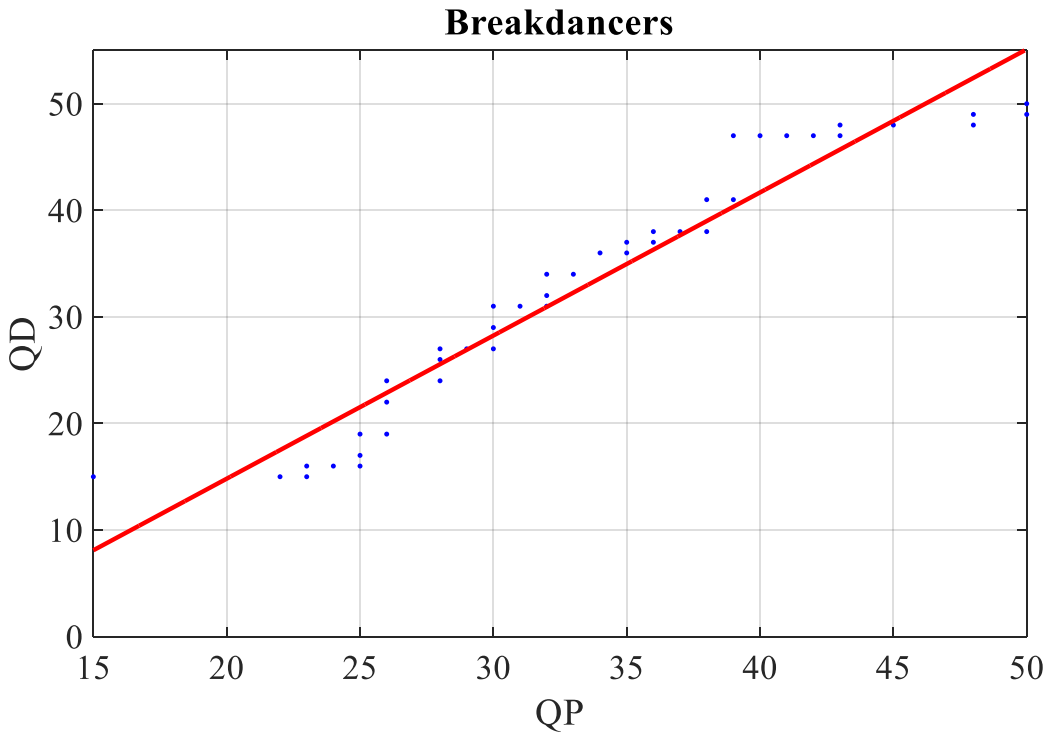


Fig. G.2. The approximate relationship between *QD* and *QP* for the optimum pairs for HEVC encoding for the *Breakdancers* sequence with the use of the linear model.

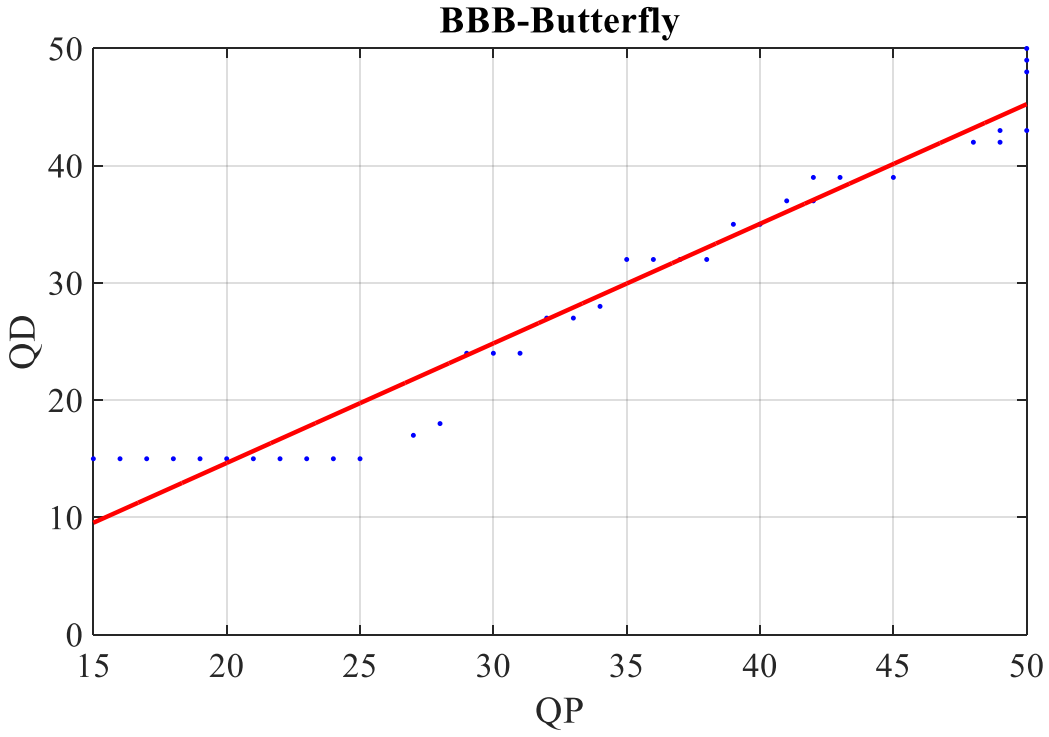


Fig. G.3. The approximate relationship between *QD* and *QP* for the optimum pairs for HEVC encoding for the *BBB.Butterfly* sequence with the use of the linear model.

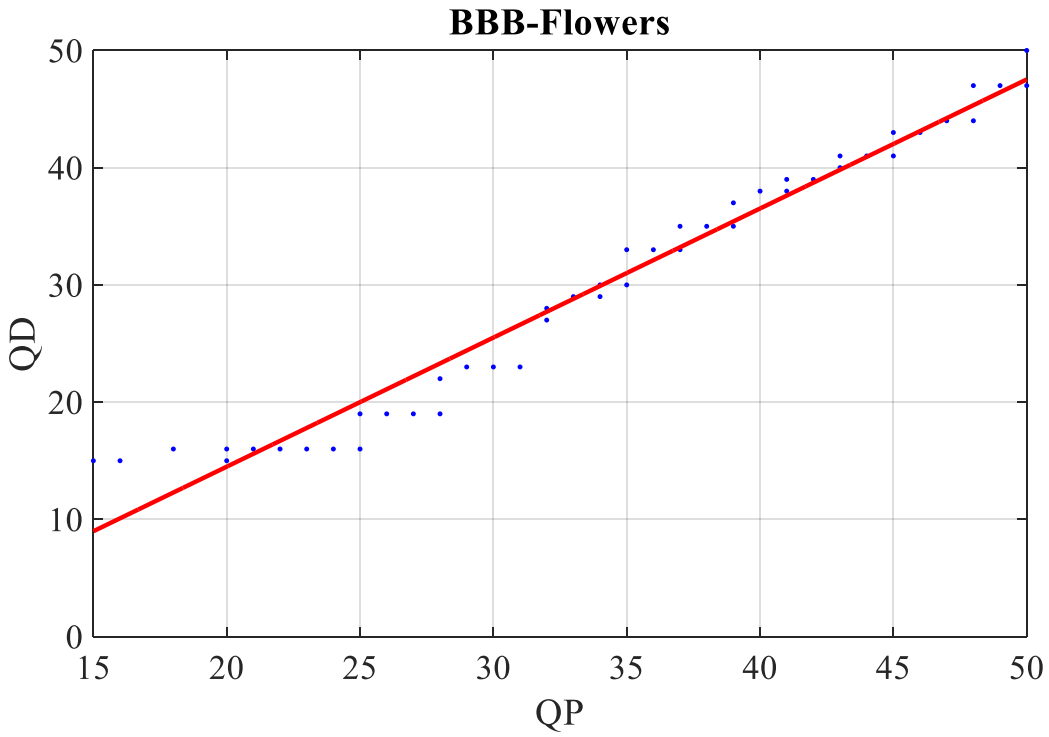


Fig. G.4. The approximate relationship between *QD* and *QP* for the optimum pairs for HEVC encoding for the *BBB.Flowers* sequence with the use of the linear model.

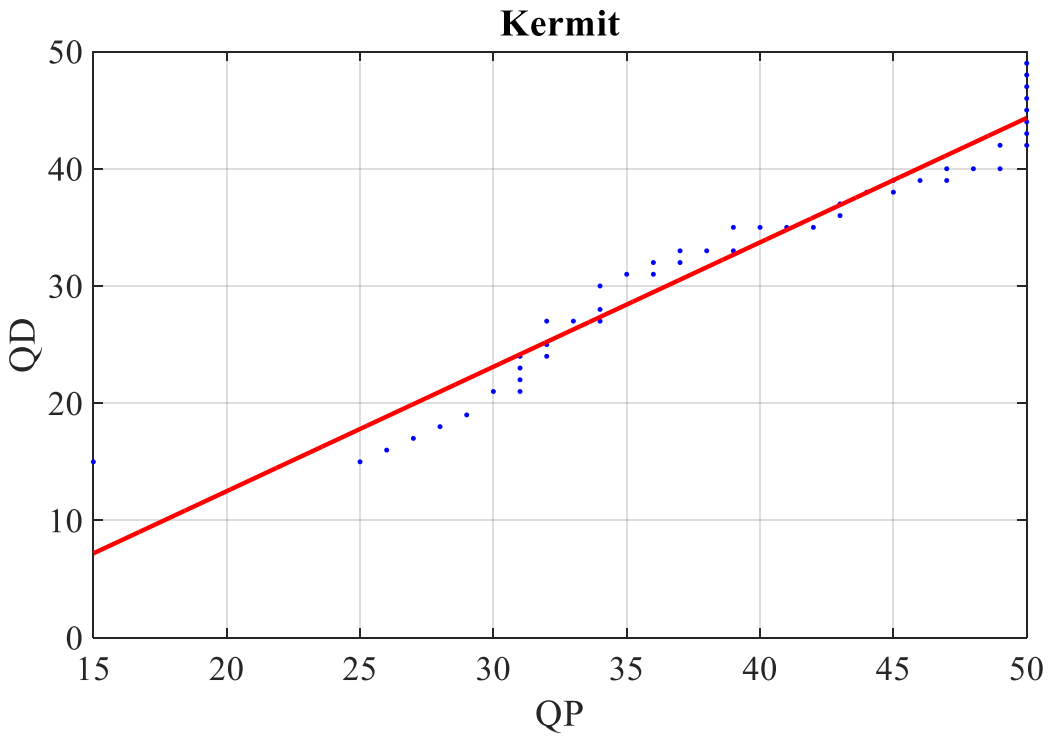


Fig. G.5. The approximate relationship between *QD* and *QP* for the optimum pairs for HEVC encoding for the *Kermit* sequence with the use of the linear model.

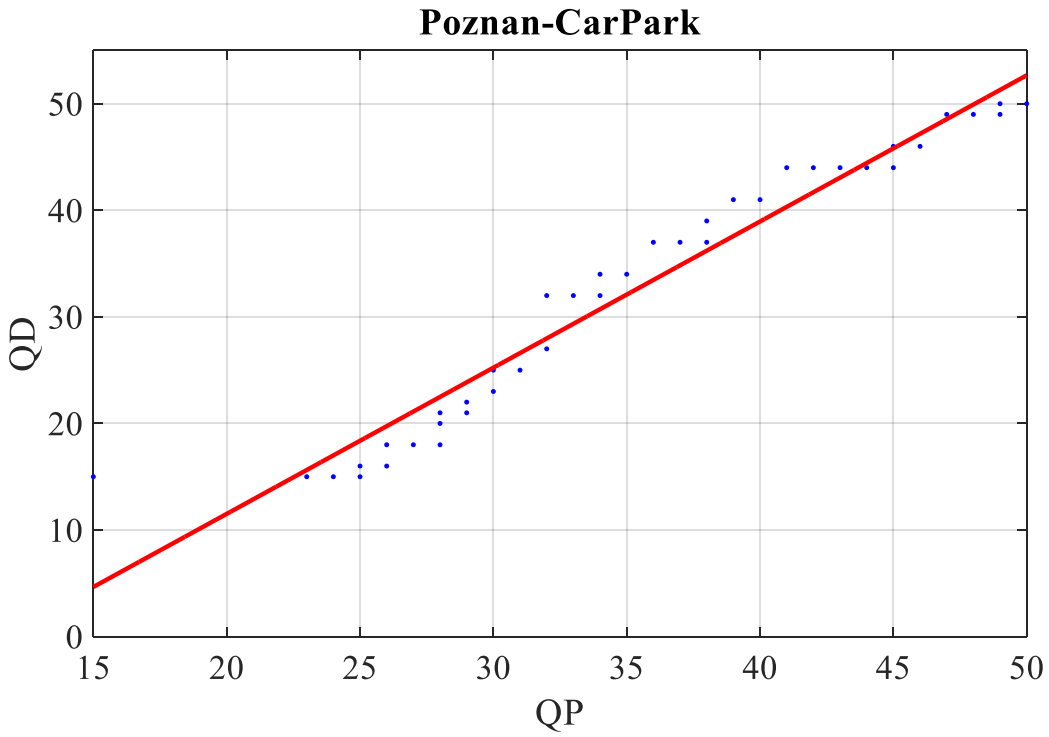


Fig. G.6. The approximate relationship between *QD* and *QP* for the optimum pairs for HEVC encoding for the *Poznan_CarPark* sequence with the use of the linear model.

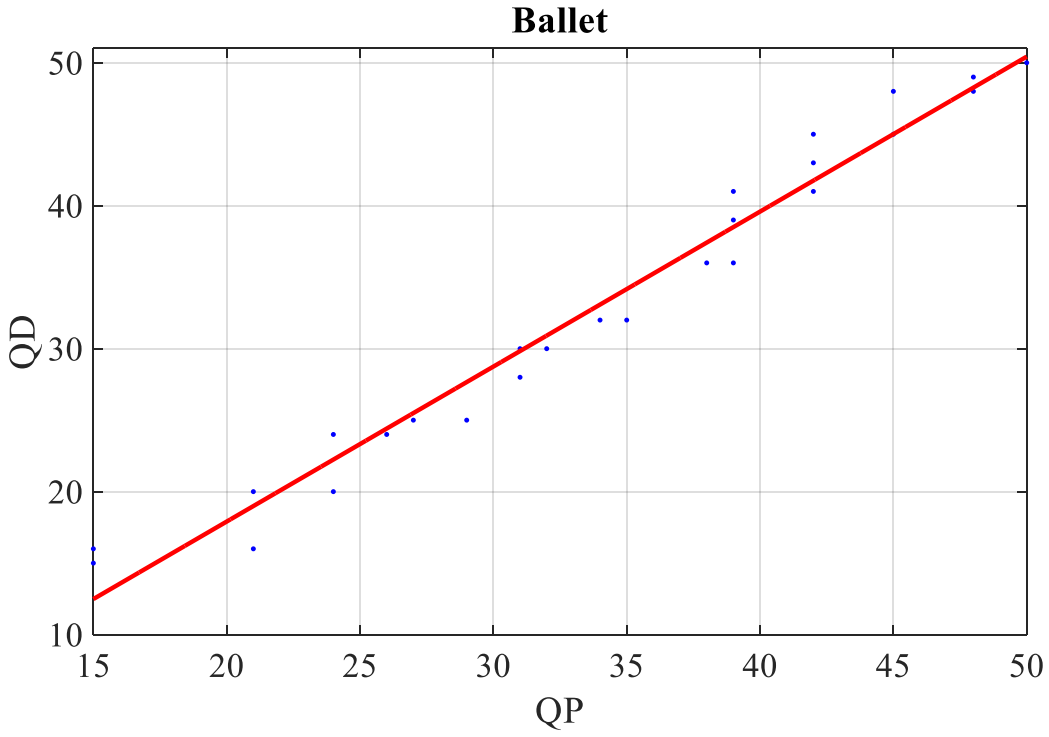


Fig. G.7. The approximate relationship between *QD* and *QP* for the optimum pairs for VVC encoding for the *Ballet* sequence with the use of the linear model.

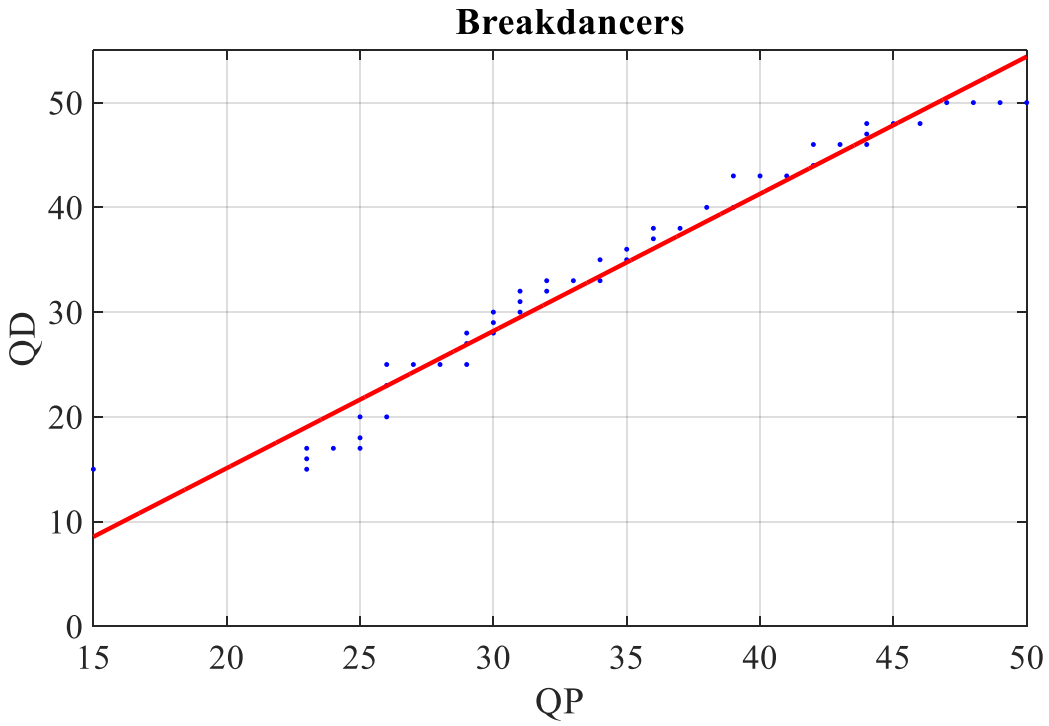


Fig. G.8. The approximate relationship between *QD* and *QP* for the optimum pairs for VVC encoding for the *Breakdancers* sequence with the use of the linear model.

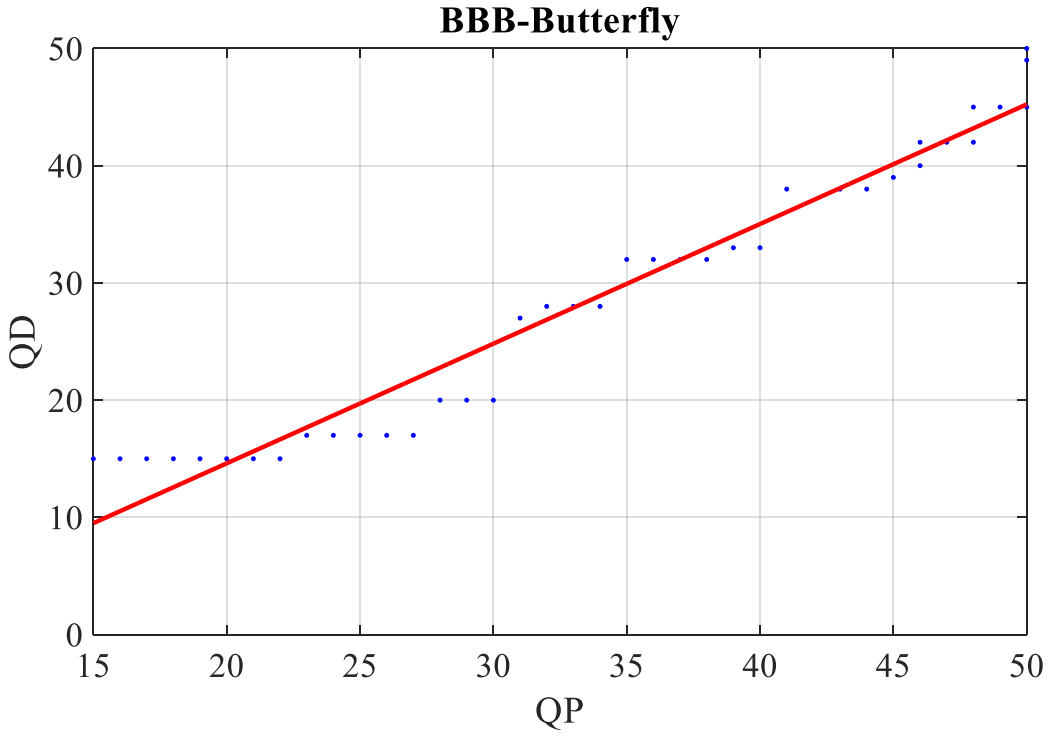


Fig. G.9. The approximate relationship between *QD* and *QP* for the optimum pairs for VVC encoding for the *BBB.Butterfly* sequence with the use of the linear model.

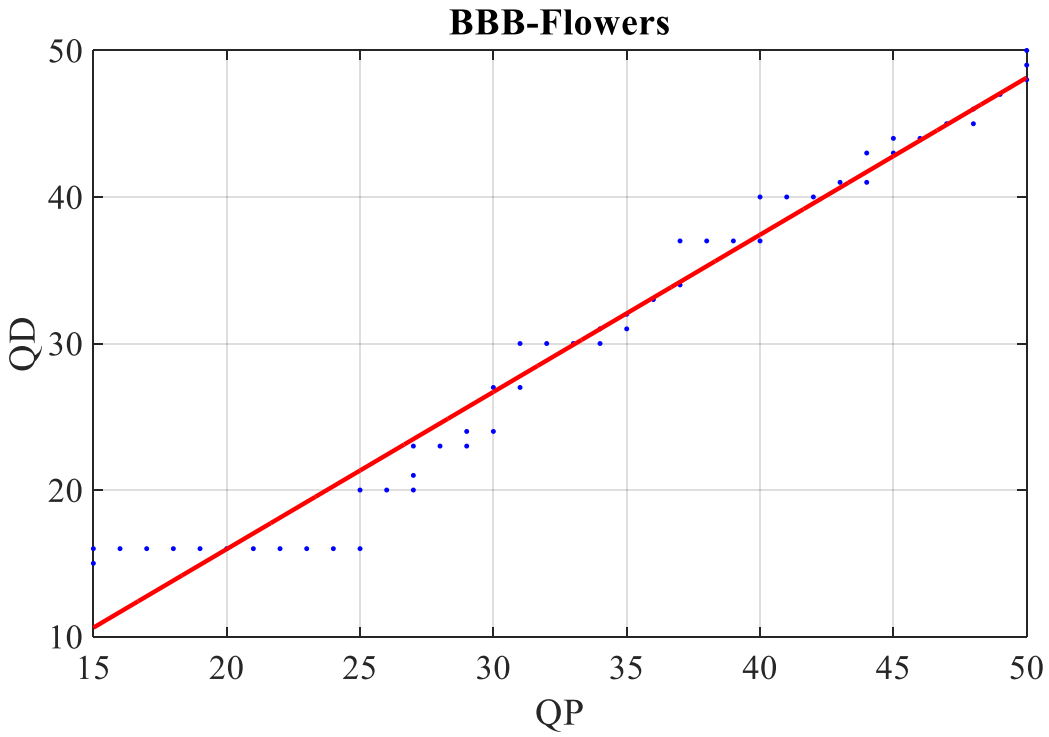


Fig. G.10. The approximate relationship between *QD* and *QP* for the optimum pairs for VVC encoding for the *BBB.Flowers* sequence with the use of the linear model.

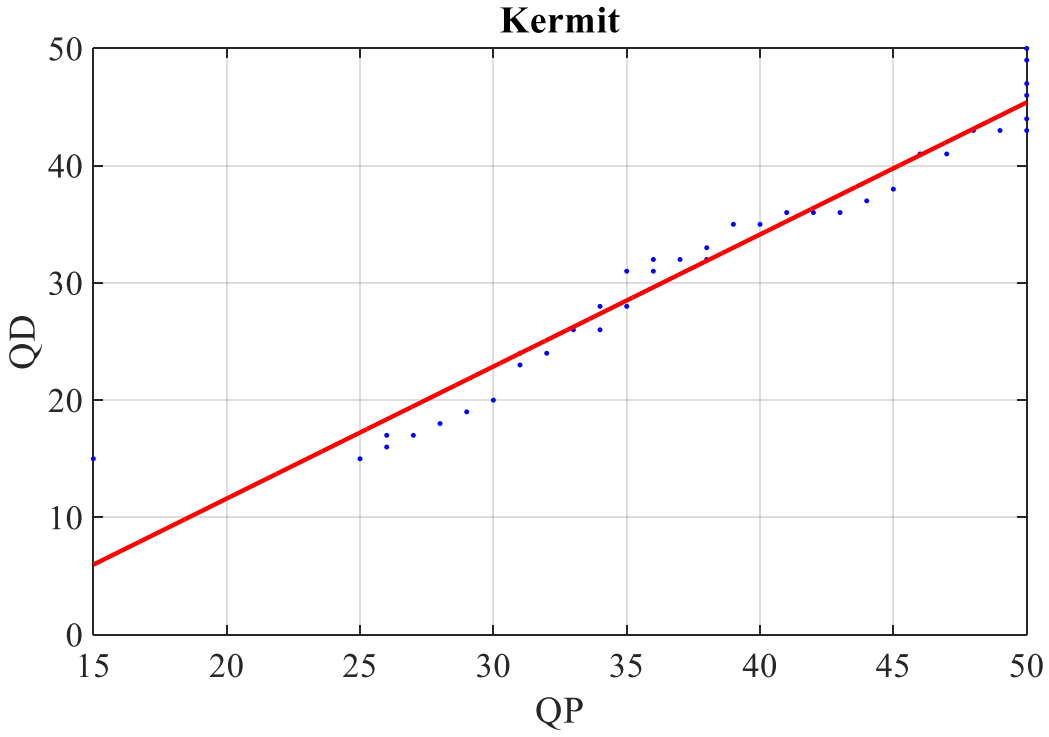


Fig. G.11. The approximate relationship between *QD* and *QP* for the optimum pairs for VVC encoding for the *Kermit* sequence with the use of the linear model.

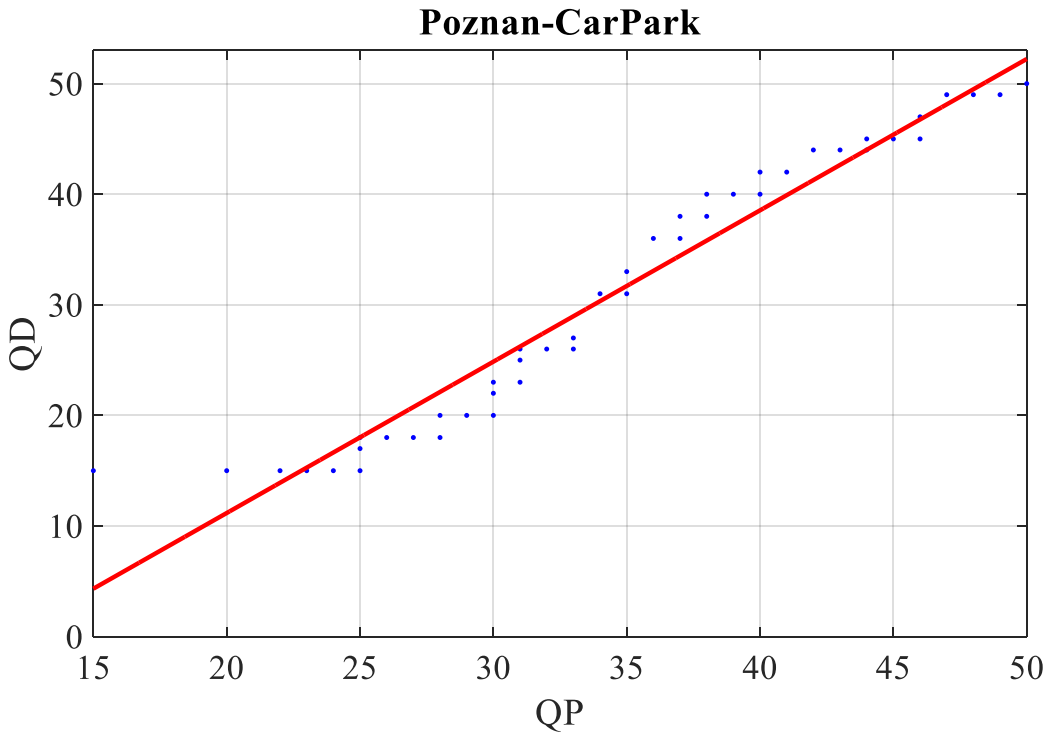


Fig. G.12. The approximate relationship between *QD* and *QP* for the optimum pairs for VVC encoding for the *Poznan_CarPark* sequence with the use of the linear model.

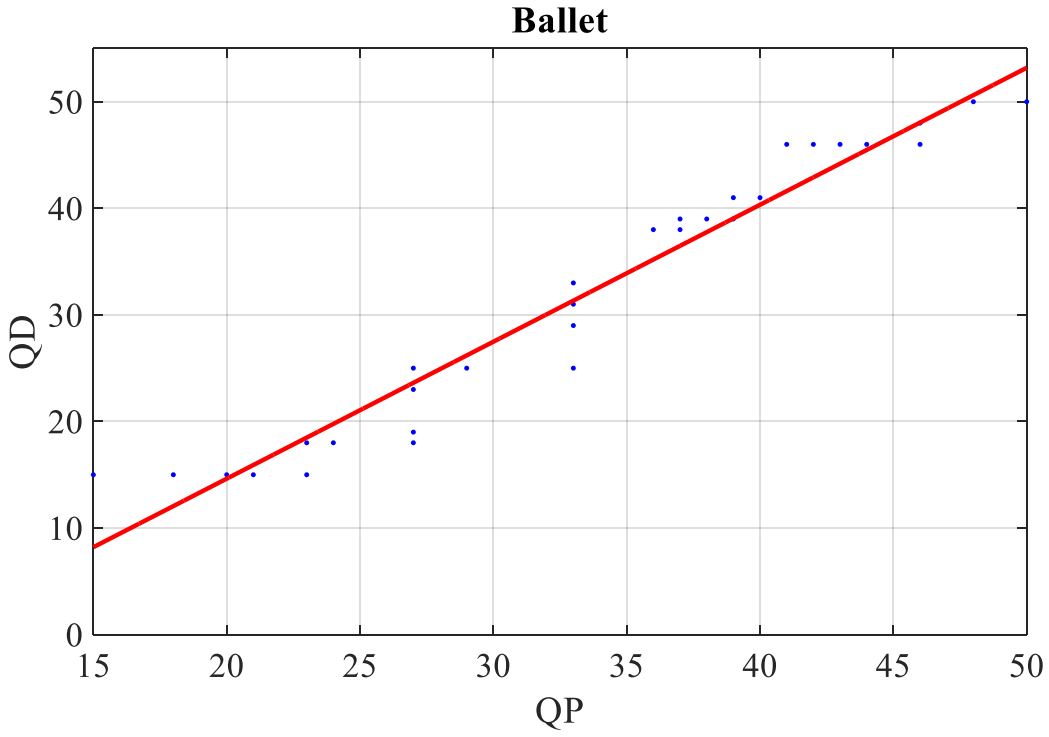


Fig. G.13. The approximate relationship between *QD* and *QP* for the optimum pairs for MV-HEVC encoding for the *Ballet* sequence with the use of the linear model.

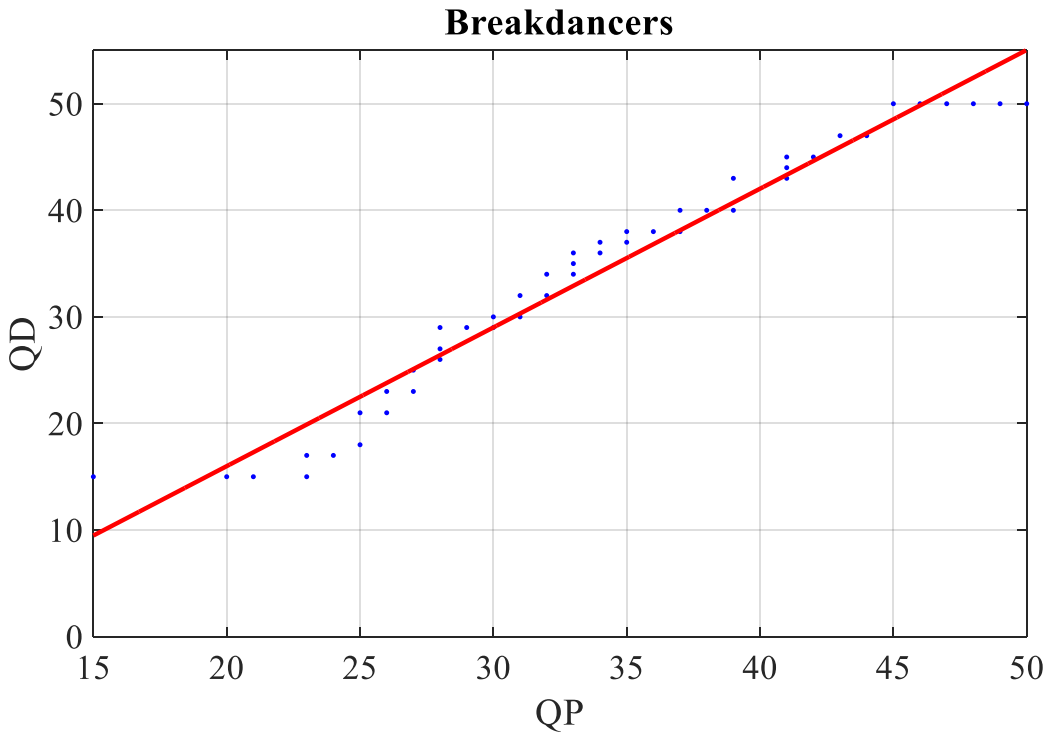


Fig. G.14. The approximate relationship between *QD* and *QP* for the optimum pairs for MV-HEVC encoding for the *Breakdancers* sequence with the use of the linear model.

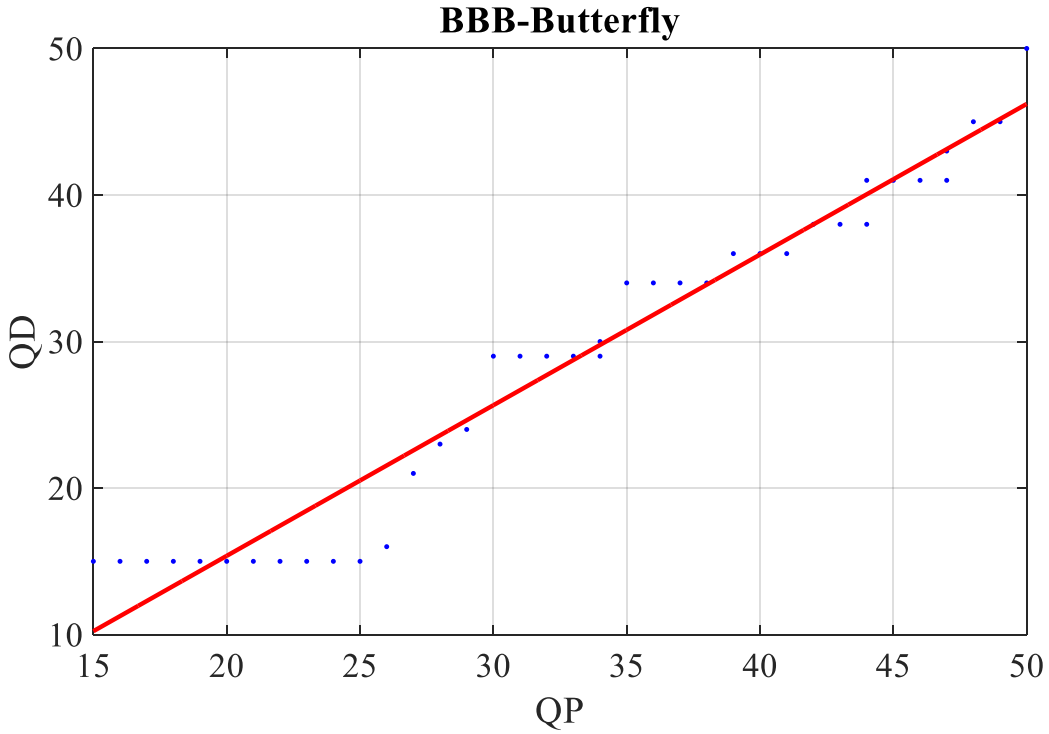


Fig. G.15. The approximate relationship between *QD* and *QP* for the optimum pairs for MV-HEVC encoding for the *BBB.Butterfly* sequence with the use of the linear model.

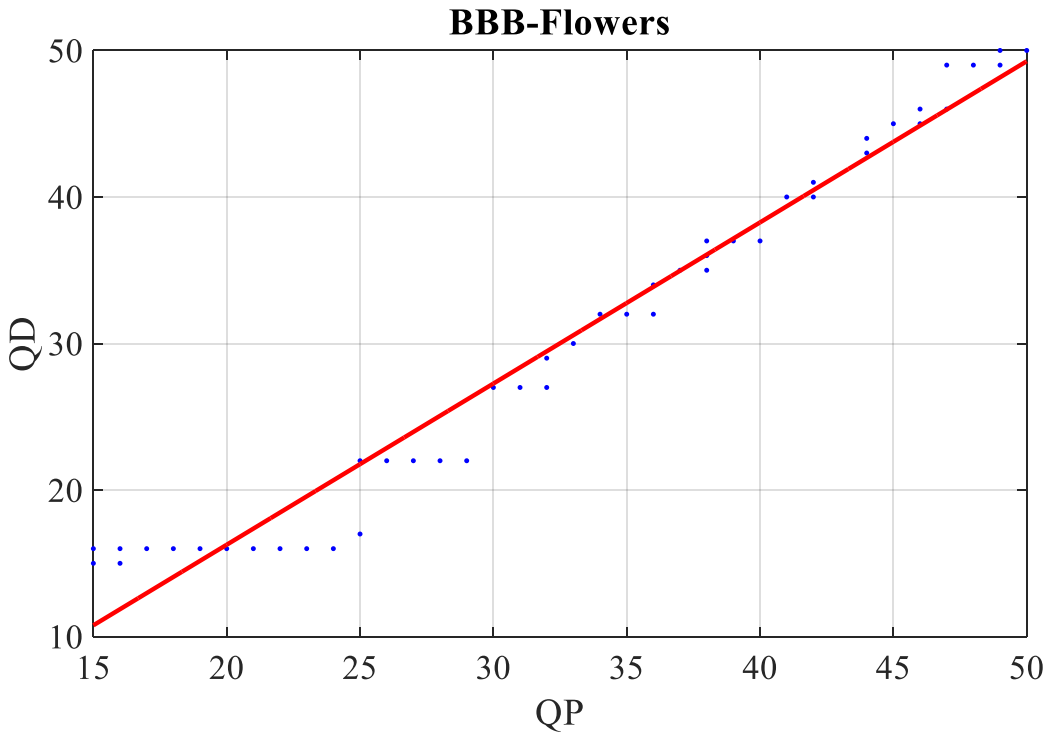


Fig. G.16. The approximate relationship between *QD* and *QP* for the optimum pairs for MV-HEVC encoding for the *BBB.Flowers* sequence with the use of the linear model.

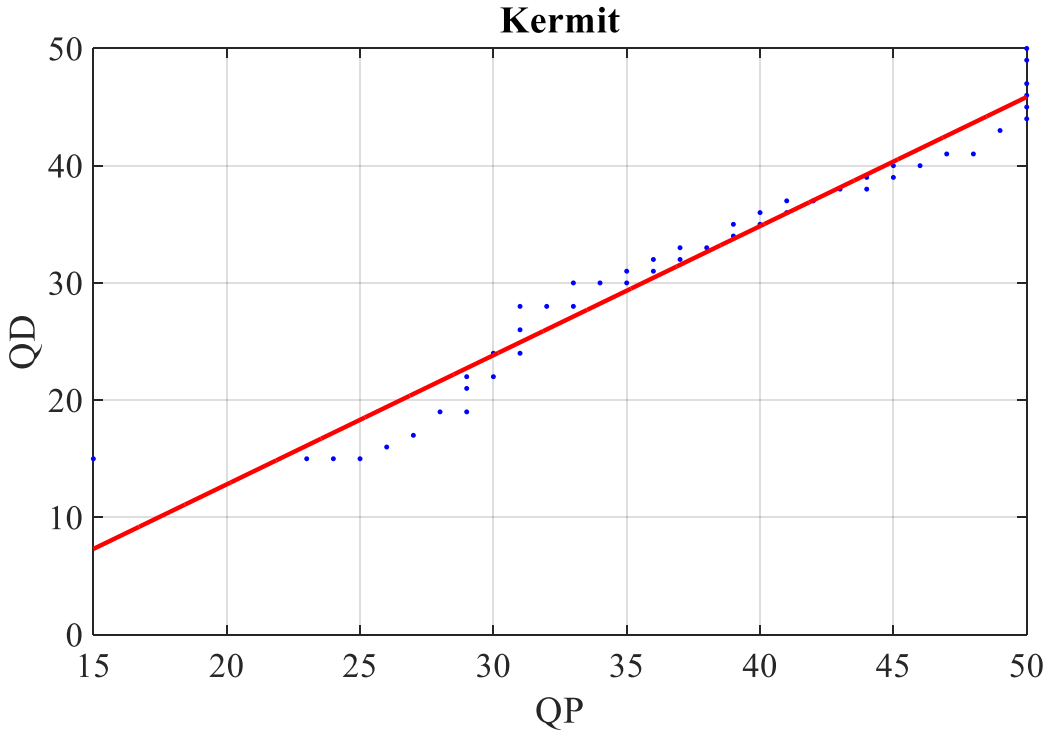


Fig. G.17. The approximate relationship between *QD* and *QP* for the optimum pairs for MV-HEVC encoding for the *Kermit* sequence with the use of the linear model.

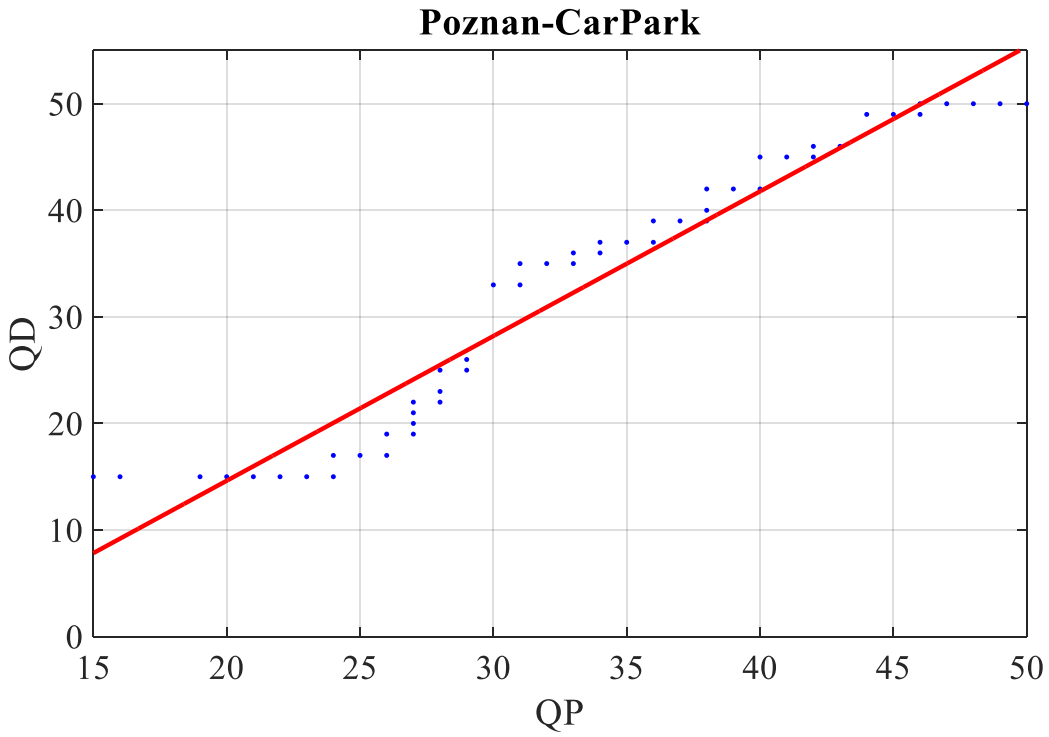


Fig. G.18. The approximate relationship between *QD* and *QP* for the optimum pairs for MV-HEVC encoding for the *Poznan_CarPark* sequence with the use of the linear model.

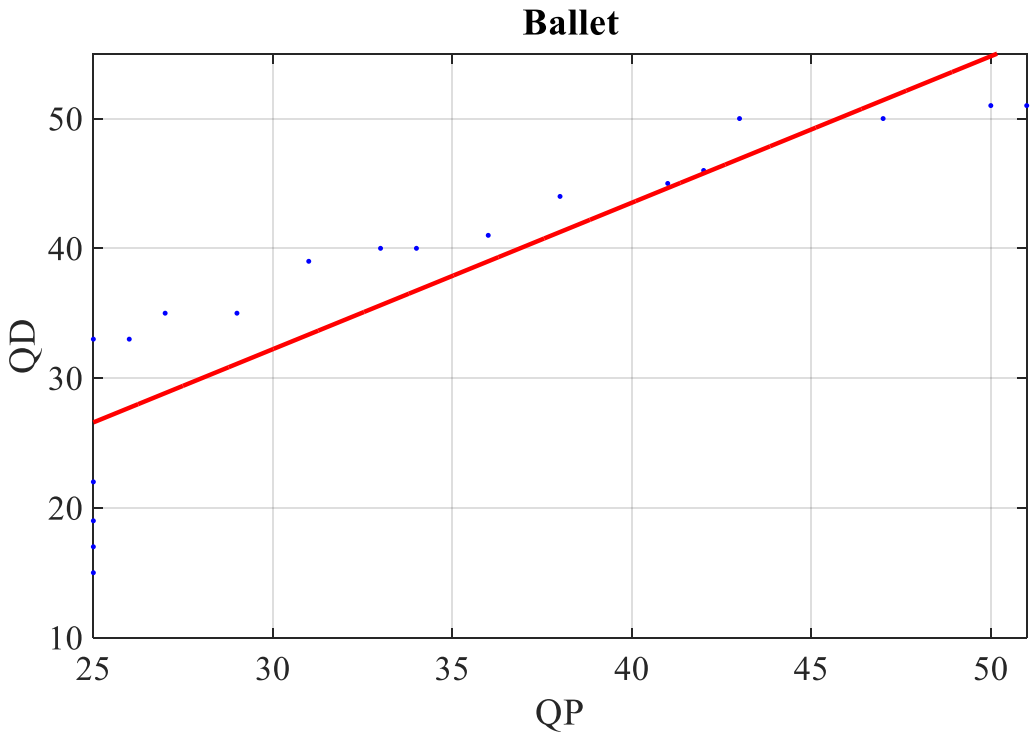


Fig. G.19. The approximate relationship between *QD* and *QP* for the optimum pairs for 3D-HEVC encoding for the *Ballet* sequence with the use of the linear model.

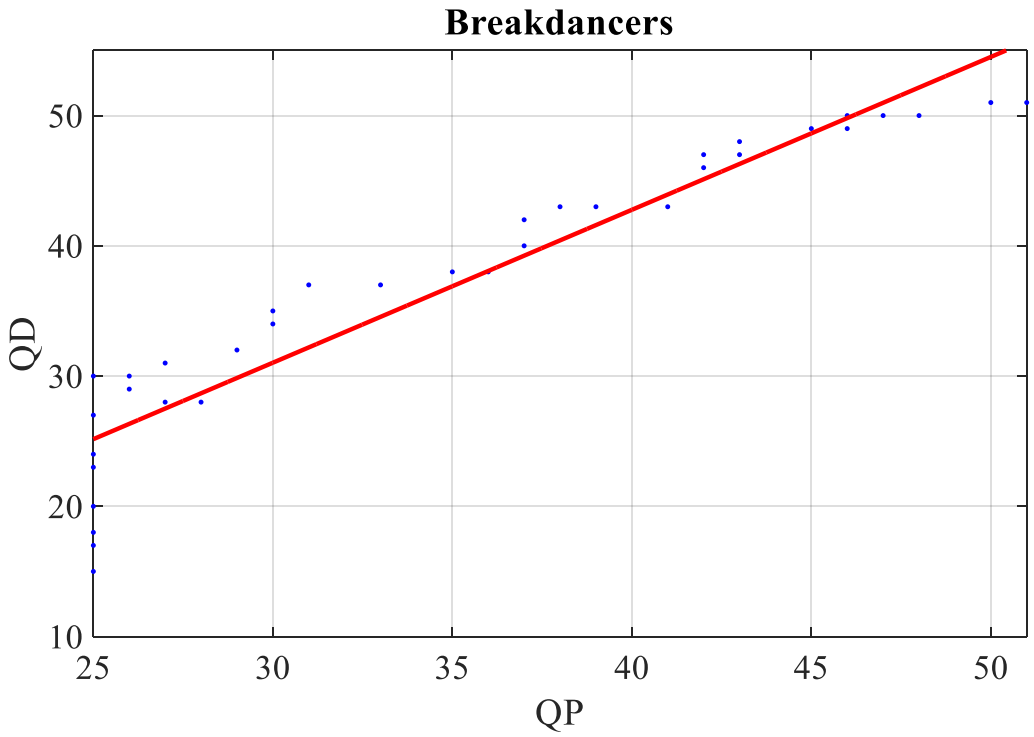


Fig. G.20. The approximate relationship between *QD* and *QP* for the optimum pairs for 3D-HEVC encoding for the *Breakdancers* sequence with the use of the linear model.

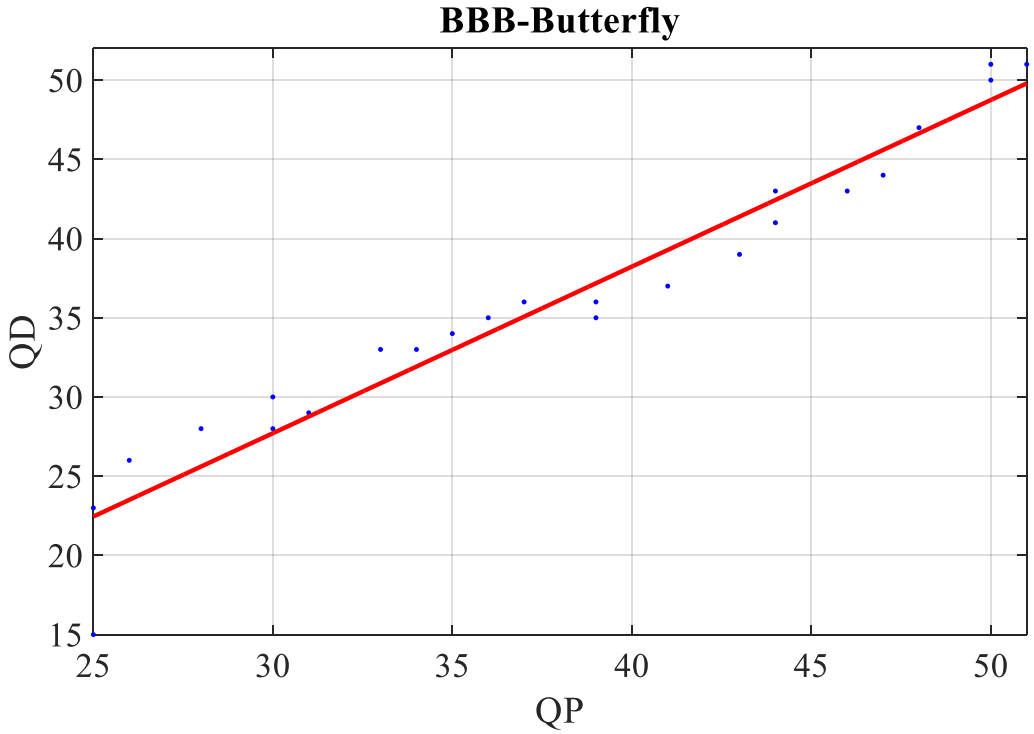


Fig. G.21. The approximate relationship between *QD* and *QP* for the optimum pairs for 3D-HEVC encoding for the *BBB.Butterfly* sequence with the use of the linear model.

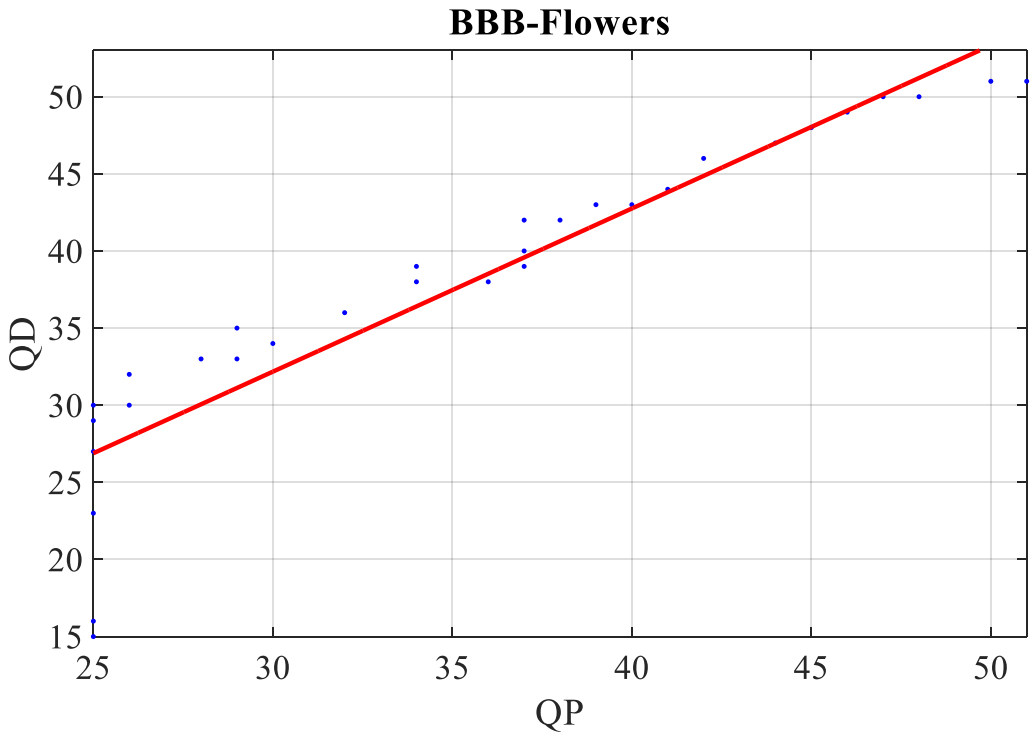


Fig. G.22. The approximate relationship between *QD* and *QP* for the optimum pairs for 3D-HEVC encoding for the *BBB.Flowers* sequence with the use of the linear model.

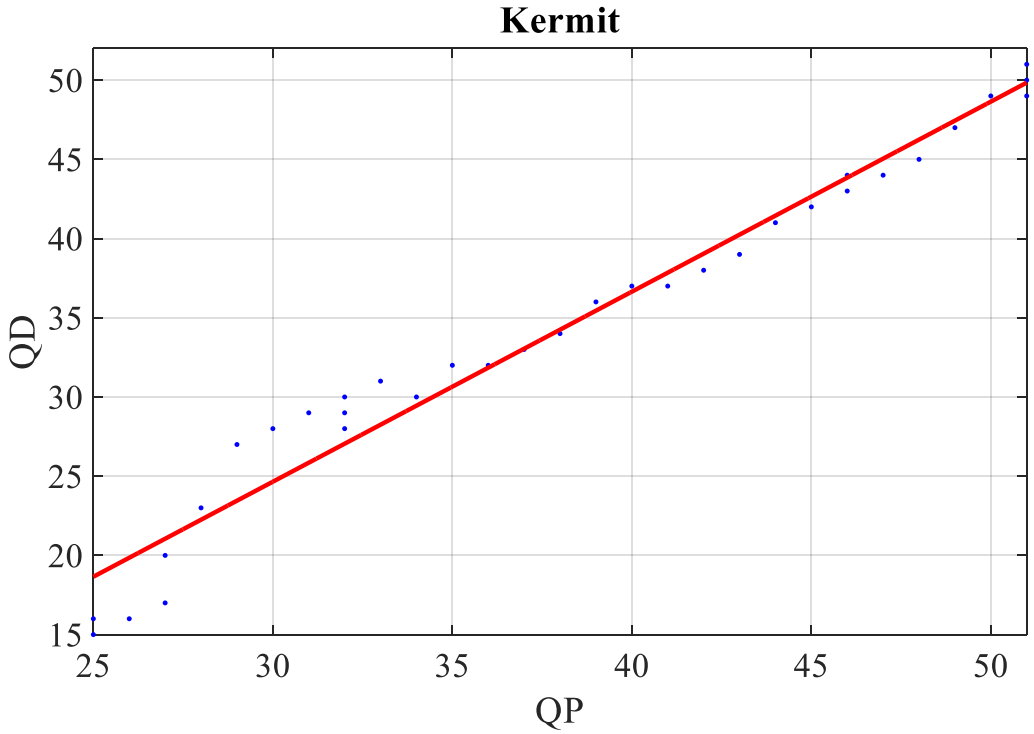


Fig. G.23. The approximate relationship between *QD* and *QP* for the optimum pairs for 3D-HEVC encoding for the *Kermit* sequence with the use of the linear model.

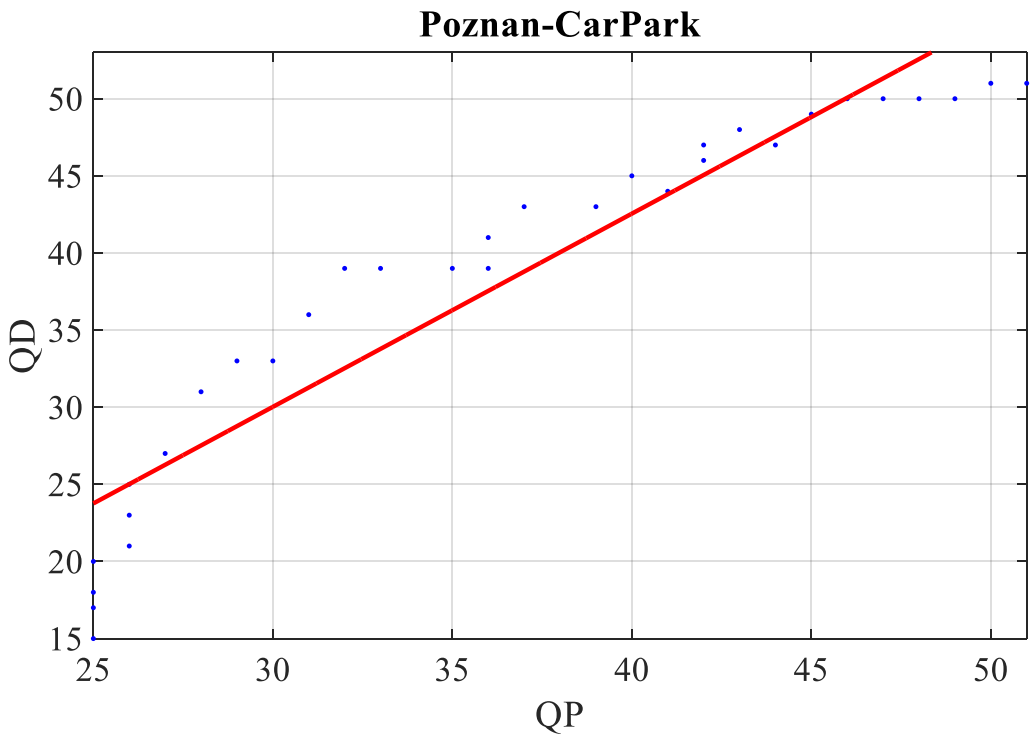


Fig. G.24. The approximate relationship between *QD* and *QP* for the optimum pairs for 3D-HEVC encoding for the *Poznan_CarPark* sequence with the use of the linear model.

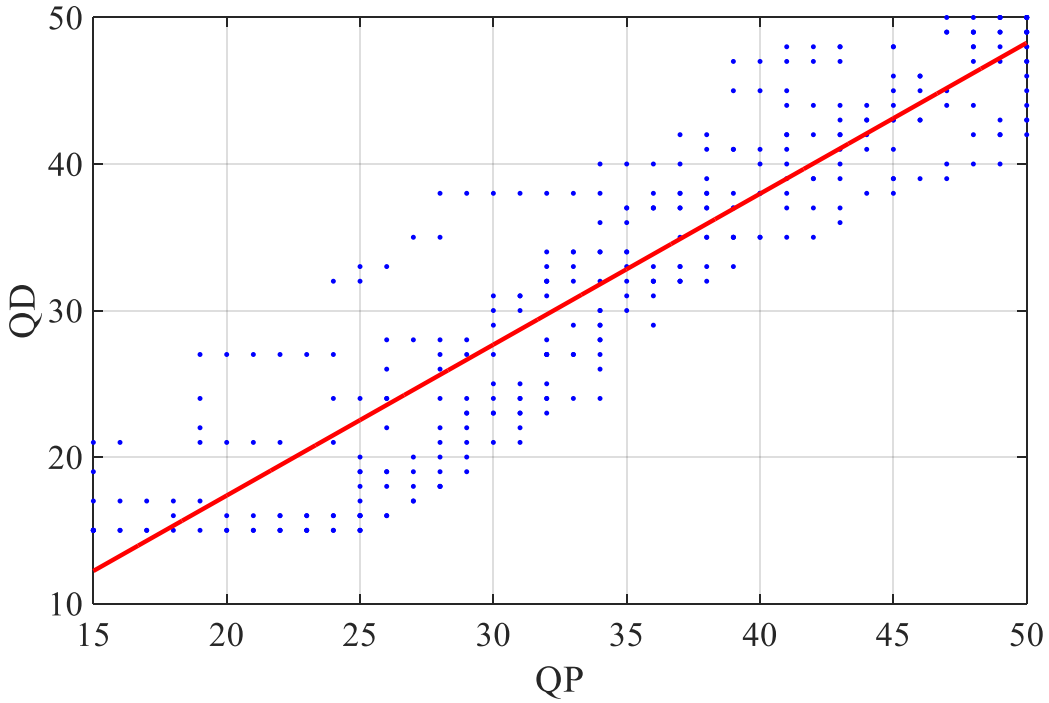


Fig. G.25. The optimum pairs of *QP*-*QD* for HEVC coding for the training sequences.

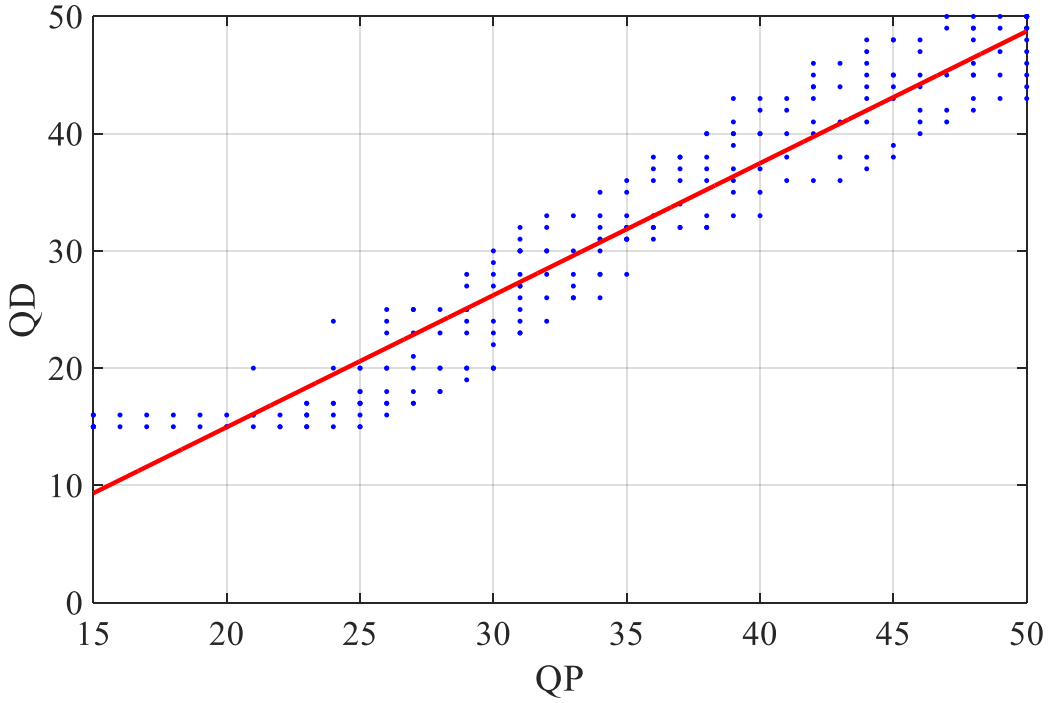


Fig. G.26. The optimum pairs of *QP*-*QD* for VVC coding for the training sequences.

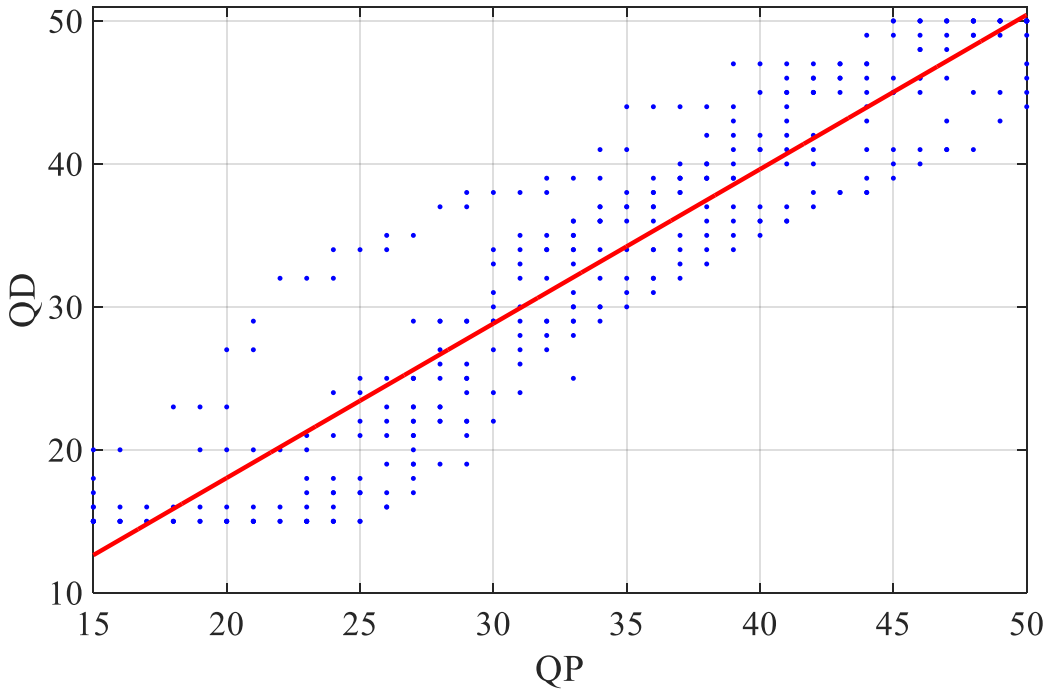


Fig. G.27. The optimum pairs of *QP*-*QD* for MV-HEVC coding for the training sequences.

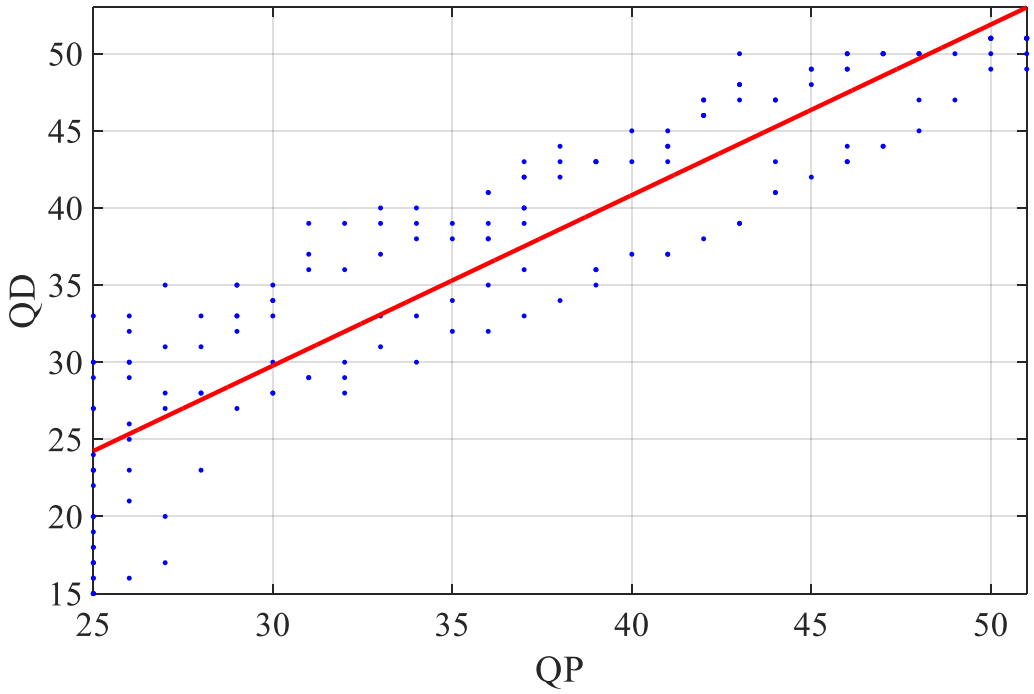


Fig. G.28. The optimum pairs of *QP*-*QD* for 3D-HEVC coding for the training sequences.

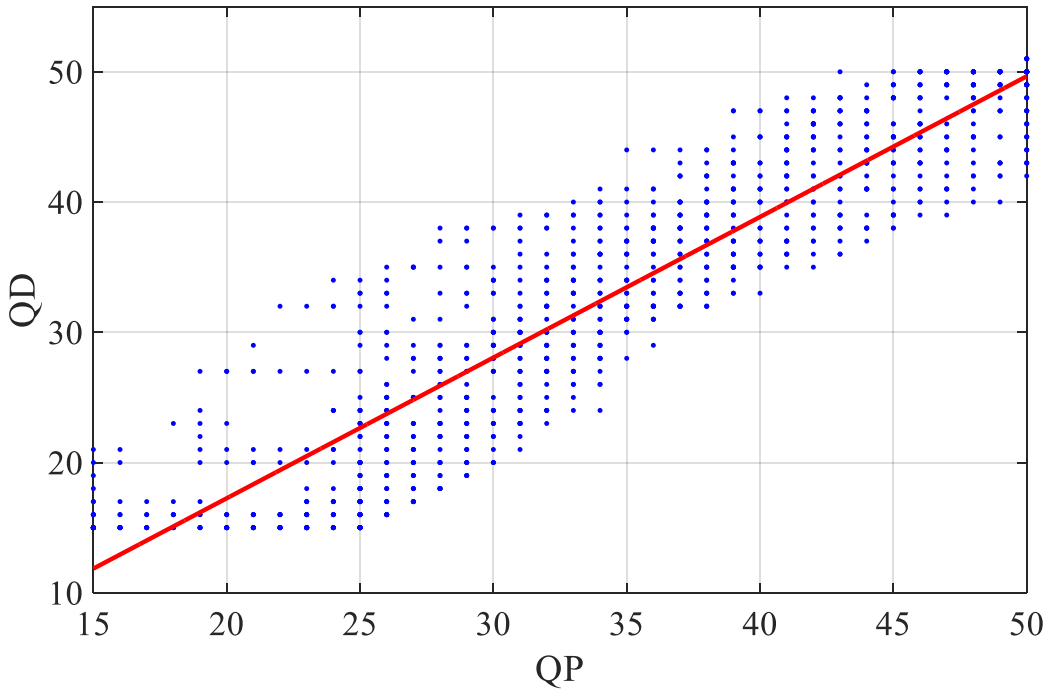


Fig. G.29. The optimum pairs of *QP*-*QD* for all considered codecs for the training sequences.

Appendix H

Related to Chapter Five

Appendix H presents an R-D curves comparison between the proposed models in Chapter 5 (Eq 5.2 with parameter values shown in Tables 5.2 or 5.3), reference approach, and optimum (*QP*, *QD*) pairs for a given codec (shown in Table 3.3) for verification sequence shown in Table 3.2. The reference approach for independent codecs is $QP = QD$, while the reference approach for joint coding is CTC (common test conditions) [Mull_14].

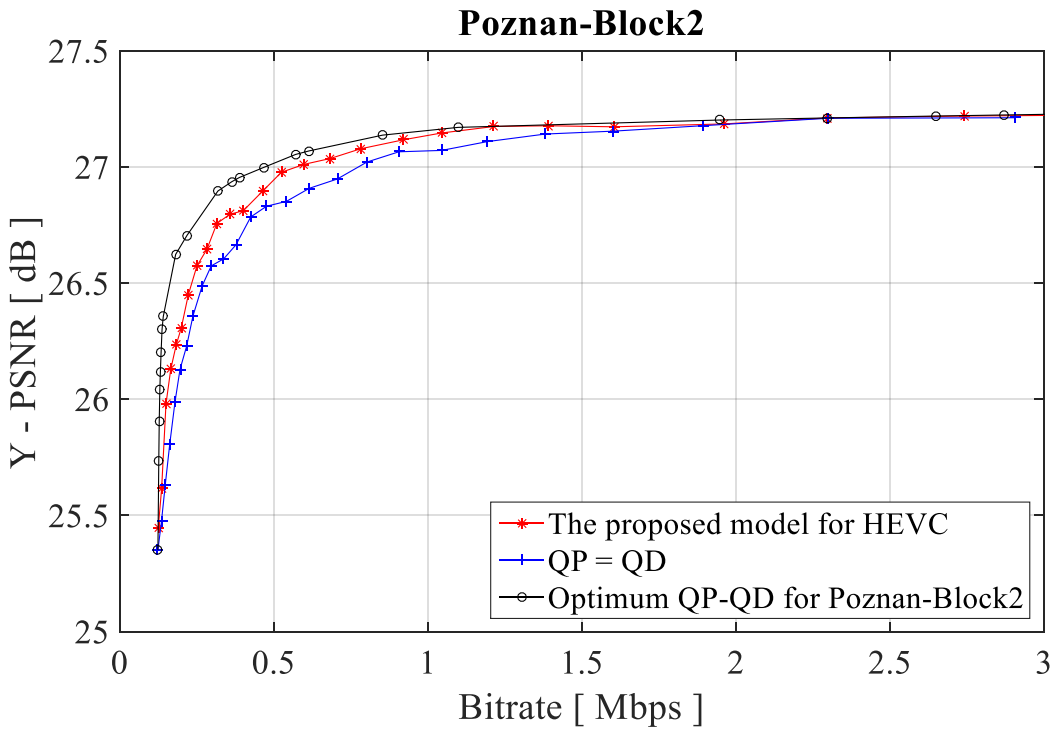


Fig. H.1. R-D curves comparison between the proposed model for HEVC, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for HEVC codec for the *Poznan_Block2* sequence.

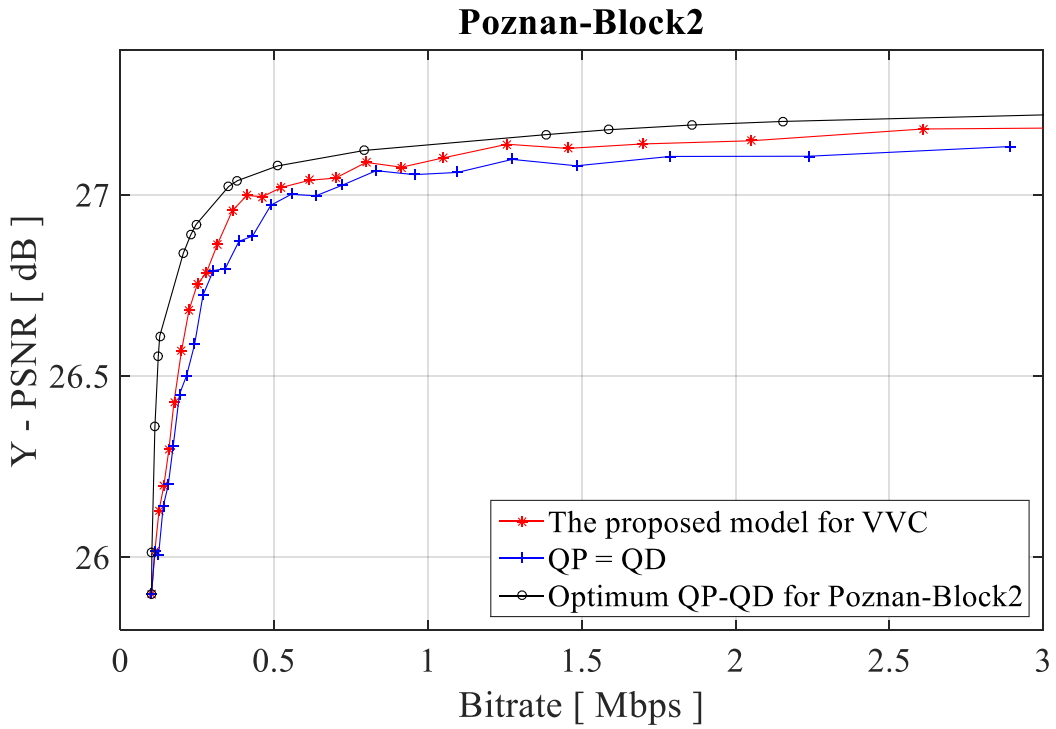


Fig. H.2. R-D curves comparison between the proposed model for VVC, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for VVC codec for the *Poznan_Block2* sequence.

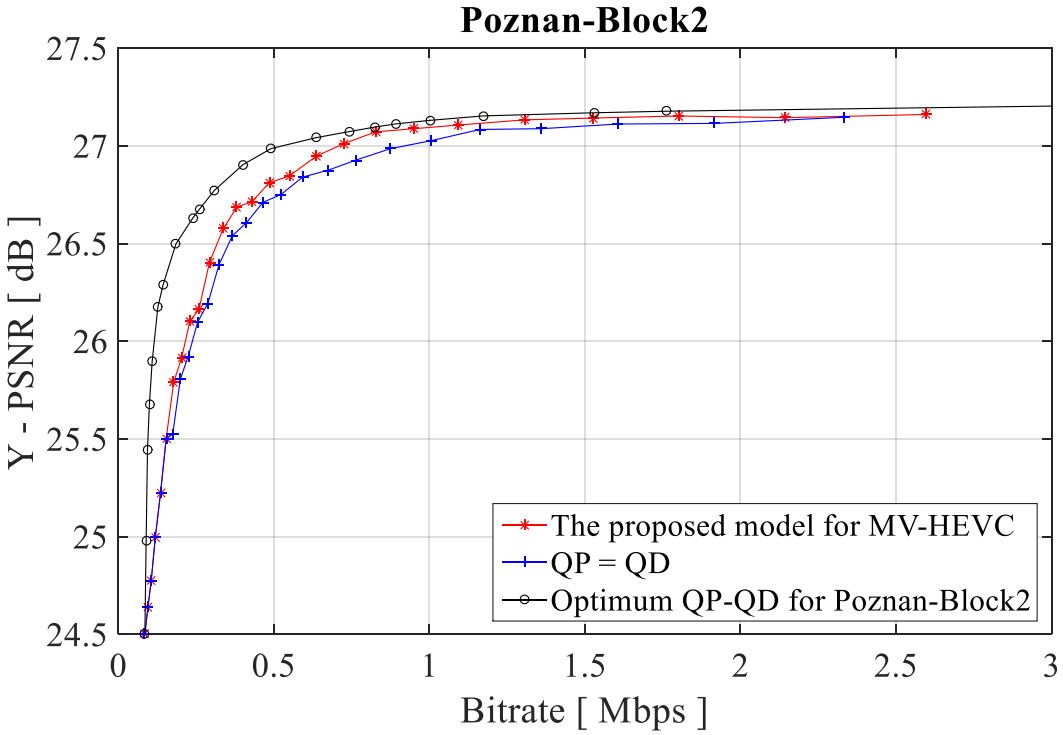


Fig. H.3. R-D curves comparison between the proposed model for MV-HEVC, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for MV-HEVC codec for the *Poznan_Block2* sequence.

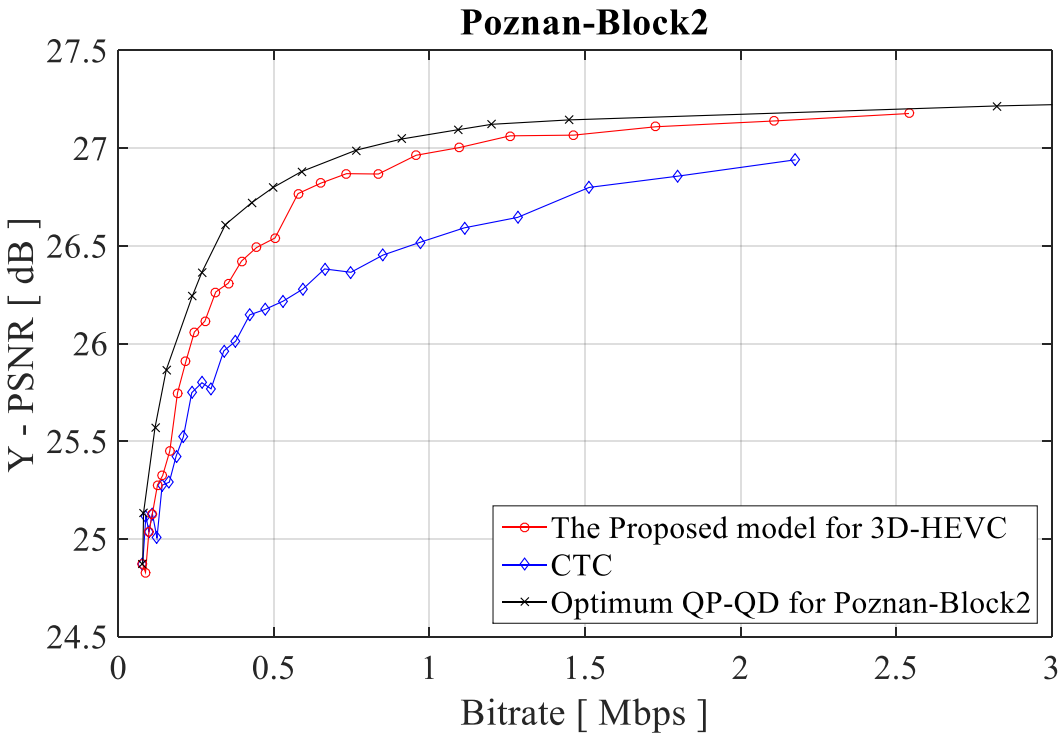


Fig. H.4. R-D curves comparison between the proposed model for 3D-HEVC, CTC approach, and optimum (*QP*, *QD*) pairs for 3D-HEVC codec for the *Poznan_Block2* sequence.

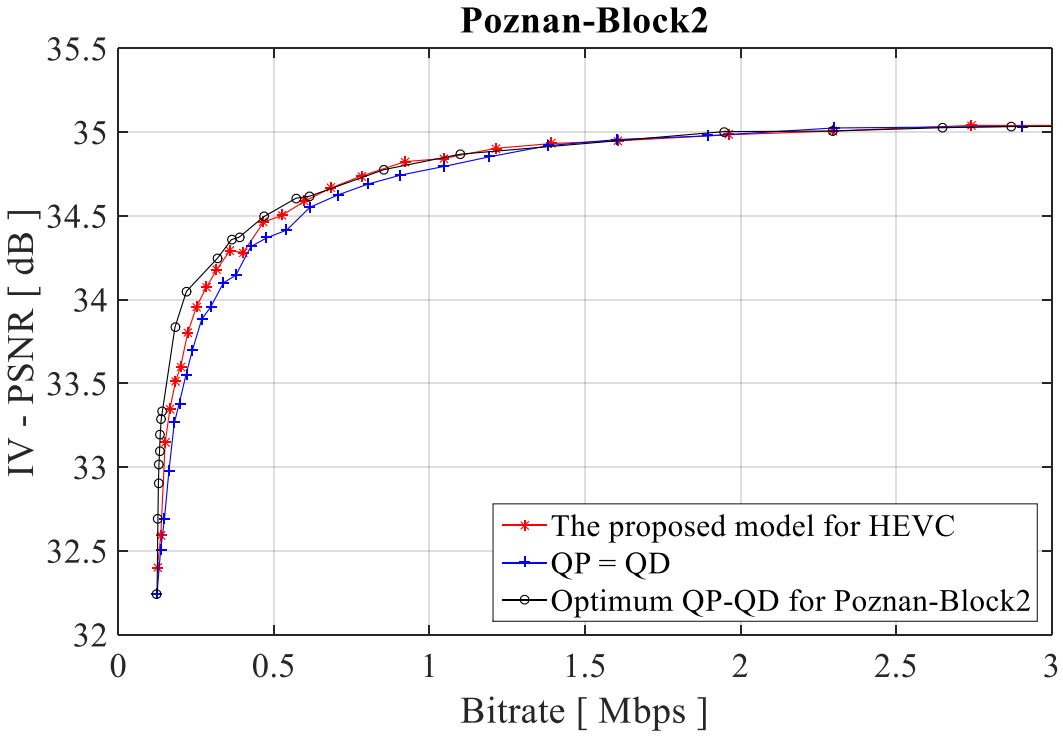


Fig. H.5. R-D curves comparison between the proposed model for HEVC, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for HEVC codec for the *Poznan_Block2* sequence.

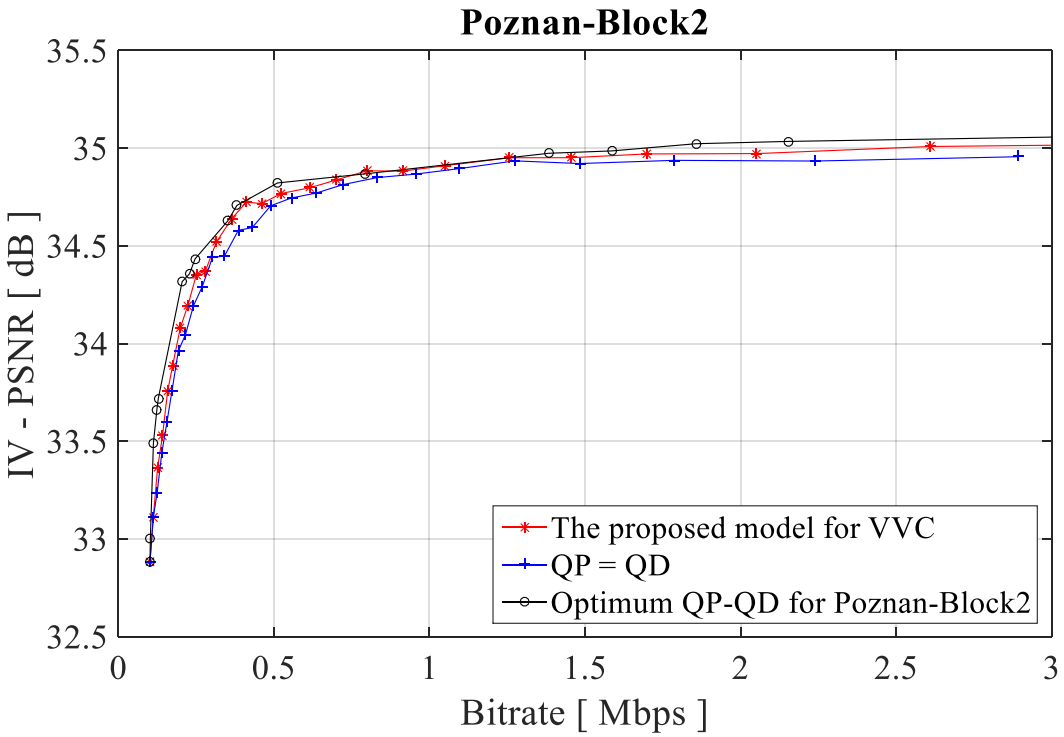


Fig. H.6. R-D curves comparison between the proposed model for VVC, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for VVC codec for the *Poznan_Block2* sequence.

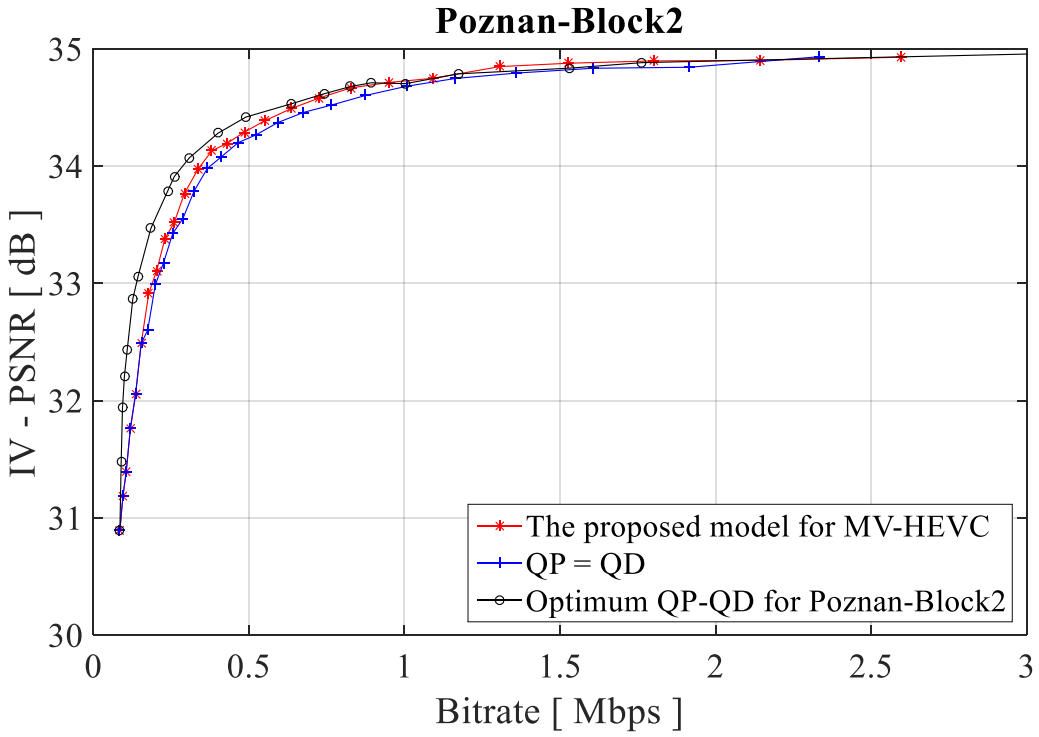


Fig. H.7. R-D curves comparison between the proposed model for MV-HEVC, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for MV-HEVC codec for the *Poznan_Block2* sequence.

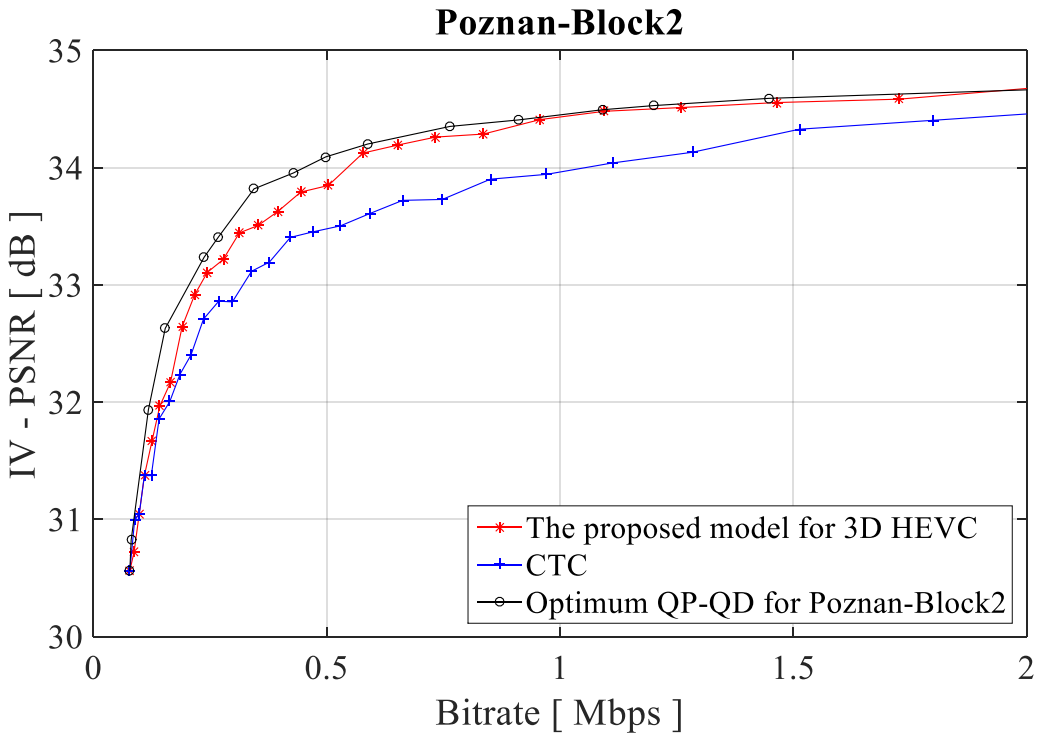


Fig. H.8. R-D curves comparison between the proposed model for 3D-HEVC, CTC approach, and optimum (*QP*, *QD*) pairs for 3D-HEVC codec for the *Poznan_Block2* sequence.

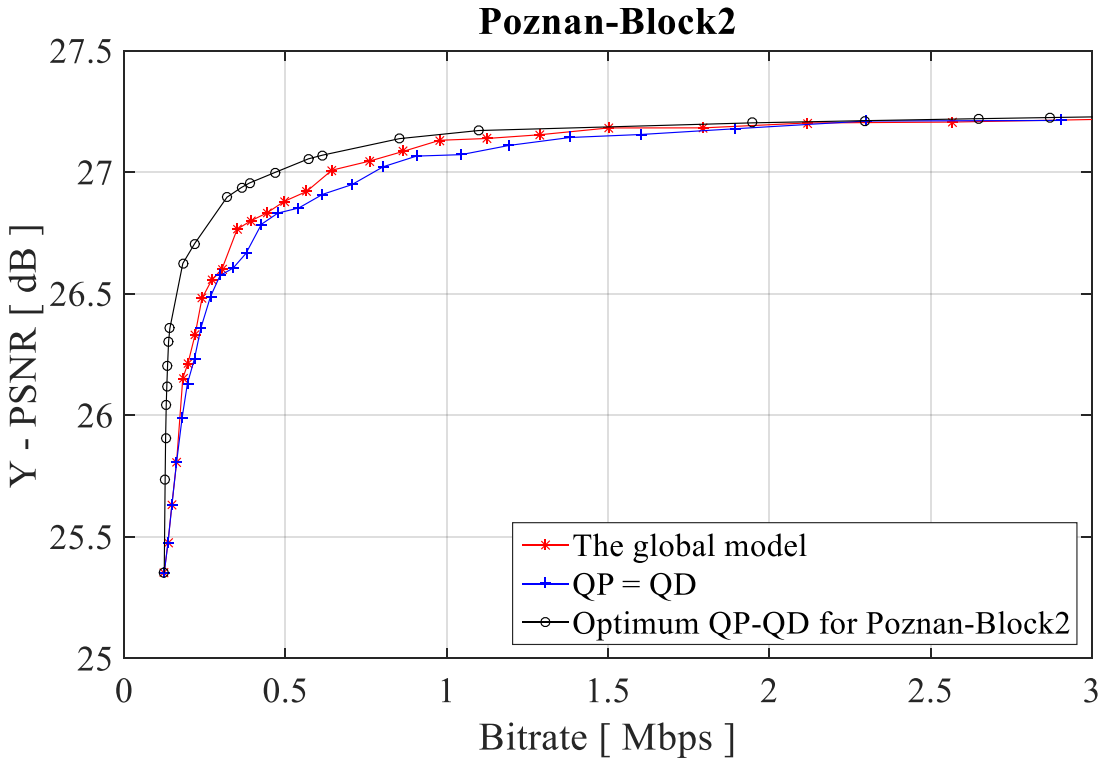


Fig. H.9. R-D curves comparison between the global model, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for HEVC codec for the *Poznan_Block2* sequence.

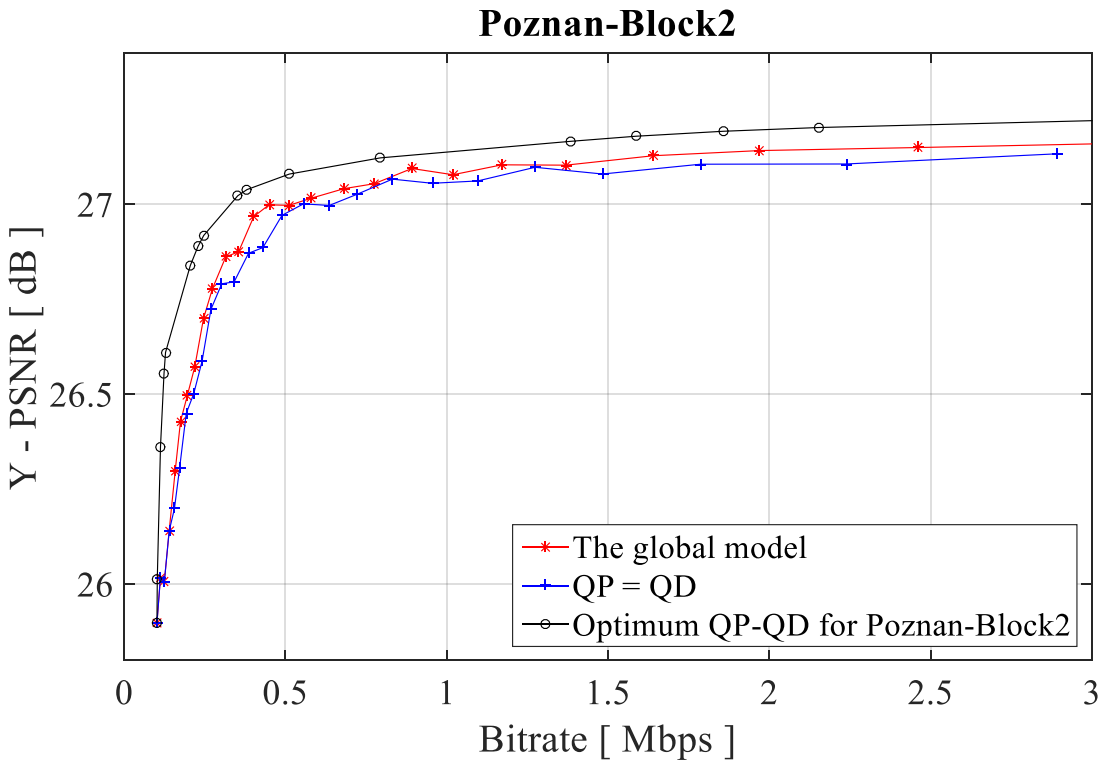


Fig. H.10. R-D curves comparison between the global model, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for VVC codec for the *Poznan_Block2* sequence.

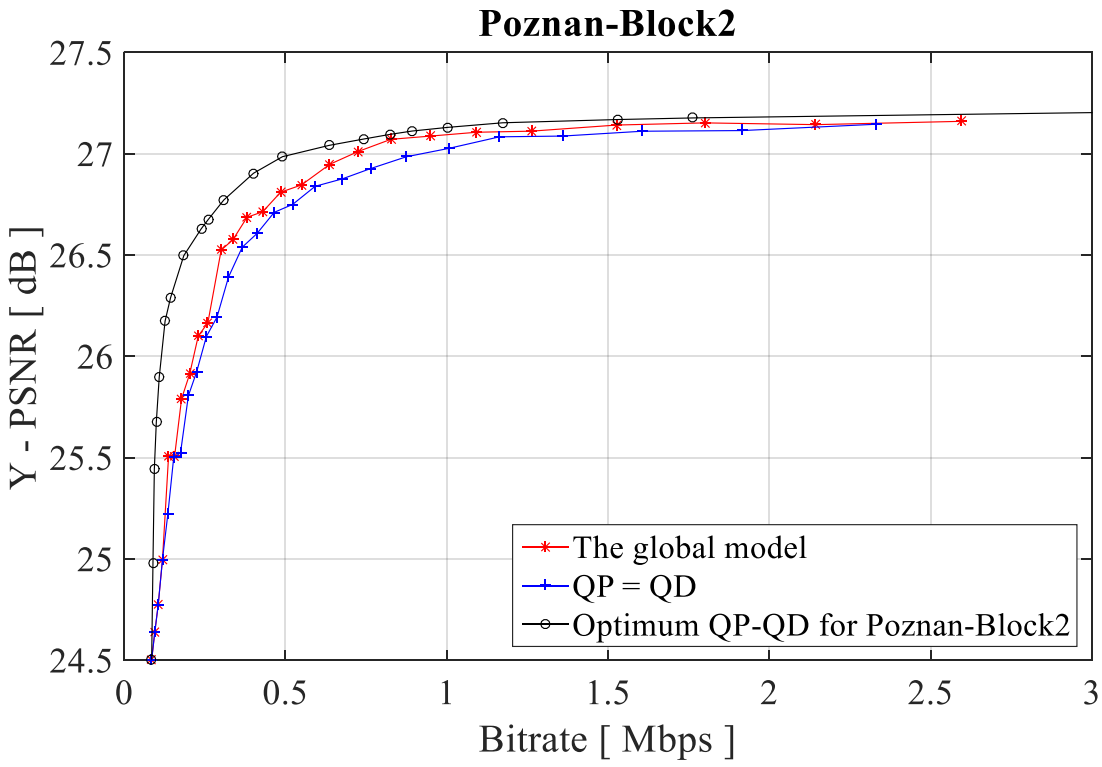


Fig. H.11. R-D curves comparison between the global model, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for MV-HEVC codec for the *Poznan_Block2* sequence.

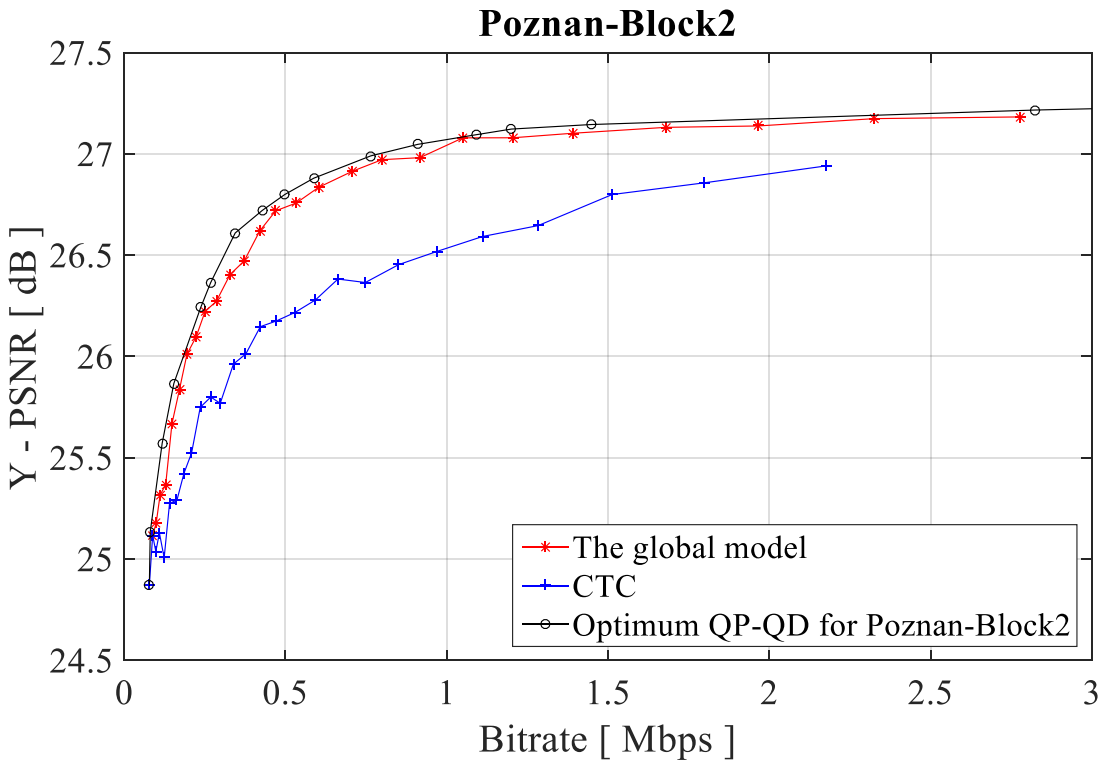


Fig. H.12. R-D curves comparison between the global model, CTC approach, and optimum (*QP*, *QD*) pairs for 3D-HEVC codec for the *Poznan_Block2* sequence.

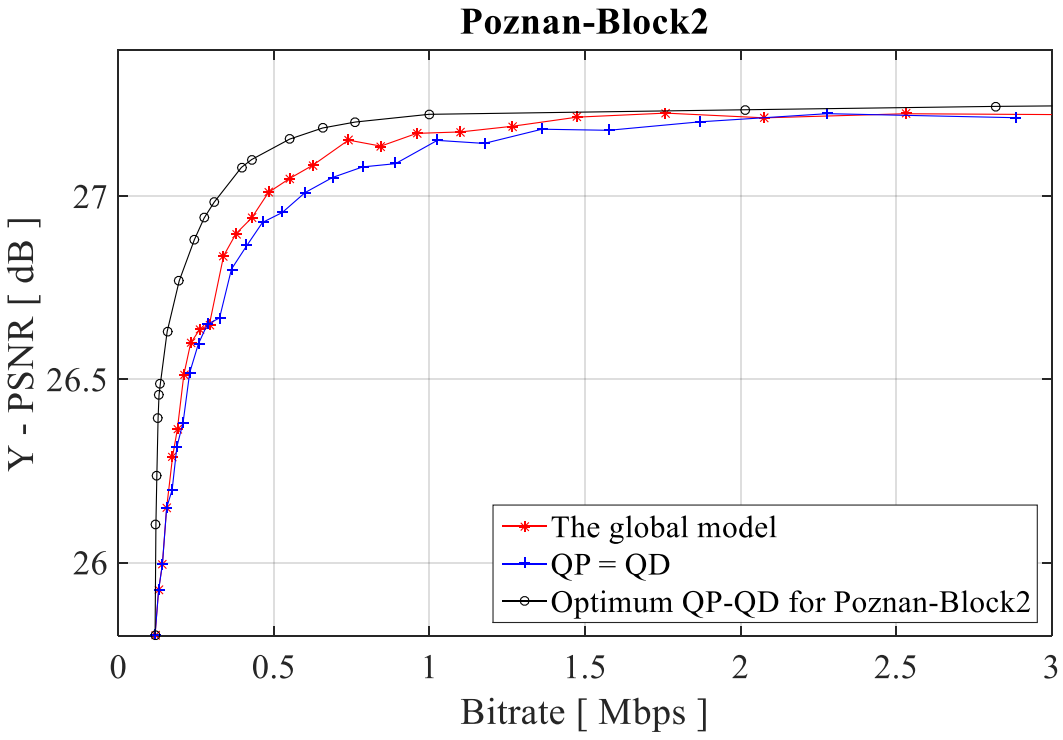


Fig. H.13. R-D curves comparison between the global model, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for VVC (VTM1.1) codec for the *Poznan_Block2* sequence.

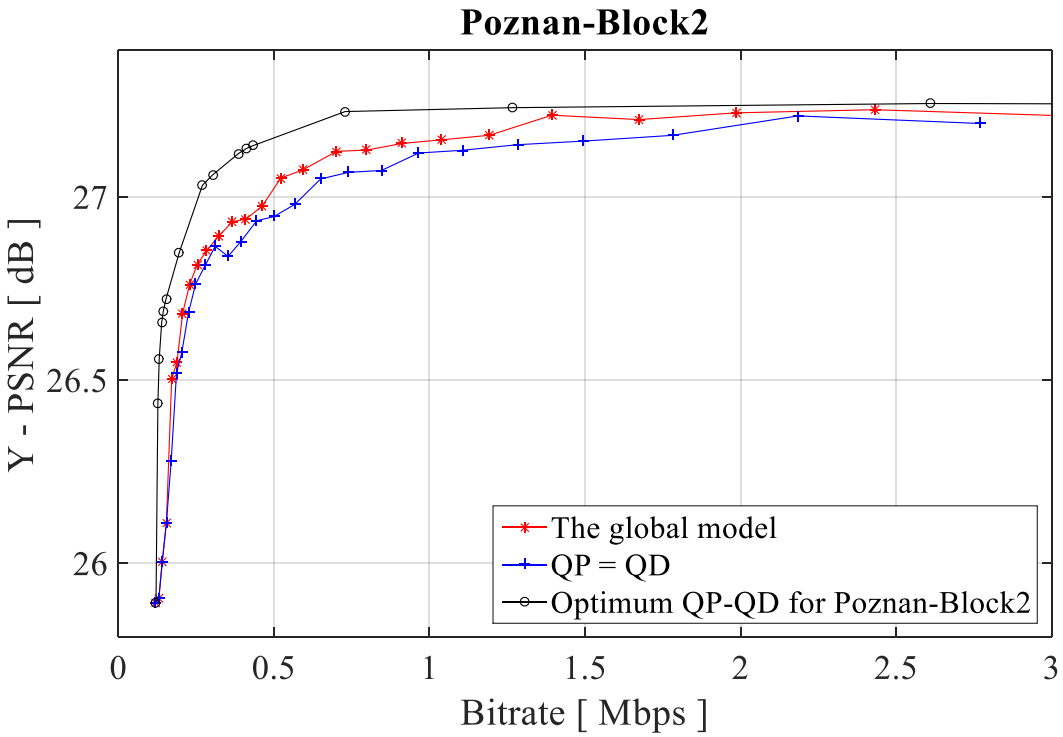


Fig. H.14. R-D curves comparison between the global model, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for VVC (HM-JEM-6.1) codec for the *Poznan_Block2* sequence.

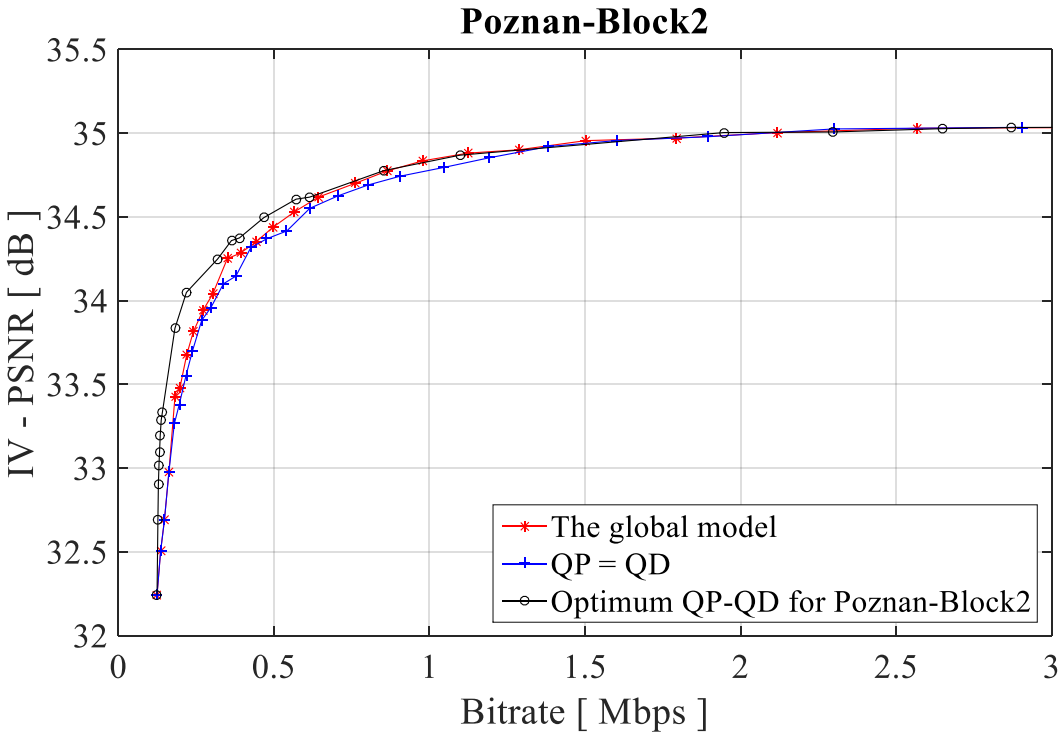


Fig. H.15. R-D curves comparison between the global model, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for HEVC codec for the *Poznan_Block2* sequence.

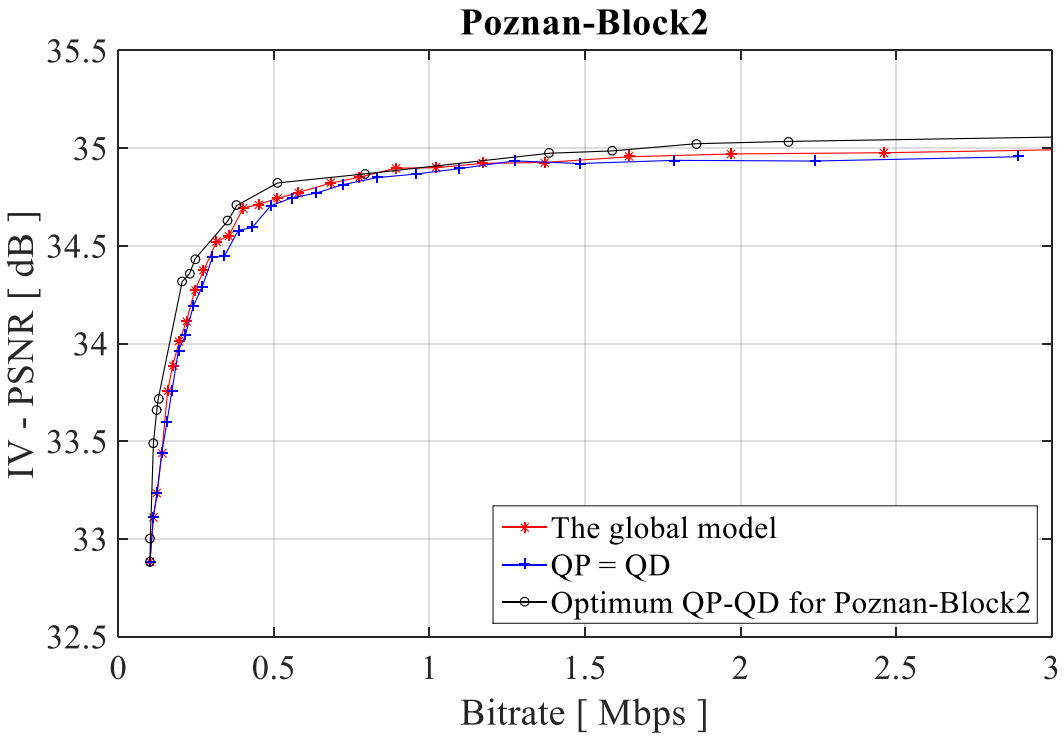


Fig. H.16. R-D curves comparison between the global model, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for VVC codec for the *Poznan_Block2* sequence.

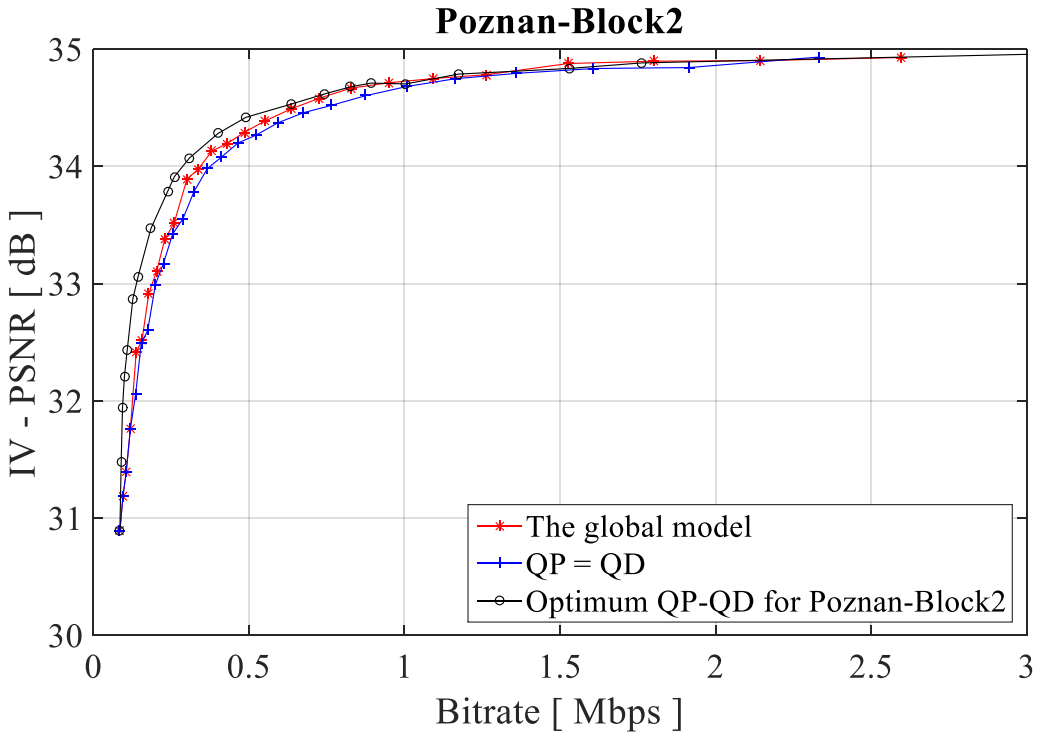


Fig. H.17. R-D curves comparison between the global model, ($QP = QD$) approach, and optimum (*QP*, *QD*) pairs for MV-HEVC codec for the *Poznan_Block2* sequence.

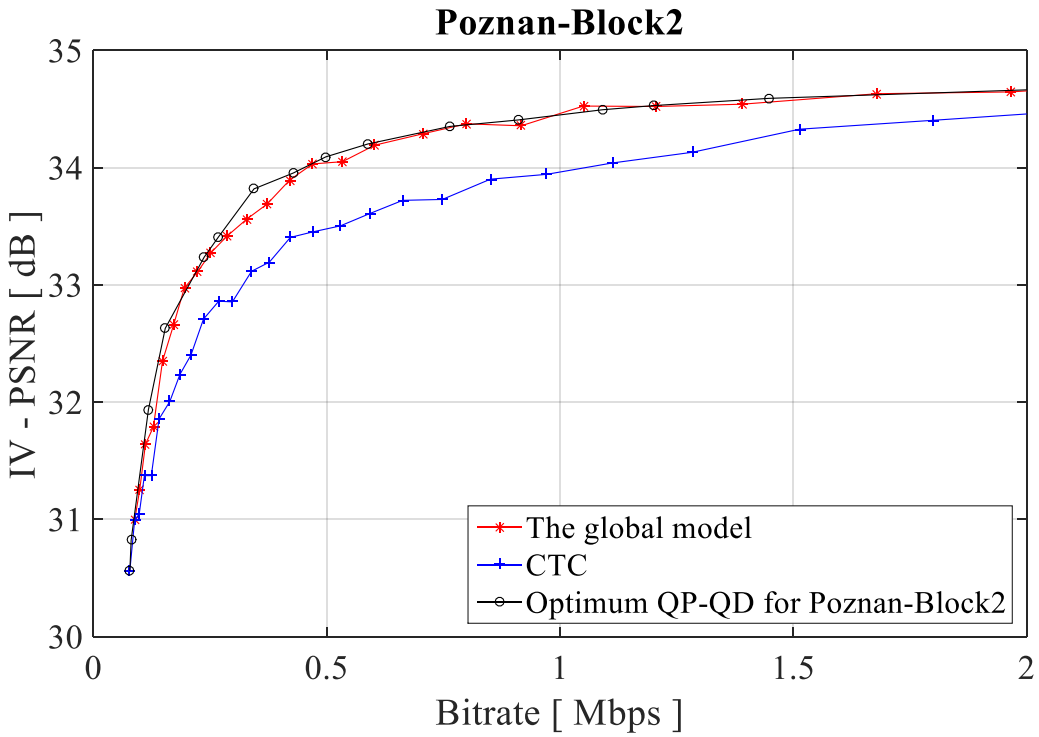


Fig. H.18. R-D curves comparison between the global model, CTC approach, and optimum (*QP*, *QD*) pairs for 3D-HEVC codec for the *Poznan_Block2* sequence.

Appendix I

Related to Chapter Five

Appendix I shows the relationship between the bitrate ratio for videos in total bitrate of stereoscopic video and *QP* for the optimum pairs for training sequences with the use of third-order polynomial regression. The black dots represent optimum (*QP*, *QD*) pairs for training sequences for the HEVC coding, while the red line represents the line obtained from a third-order polynomial regression.

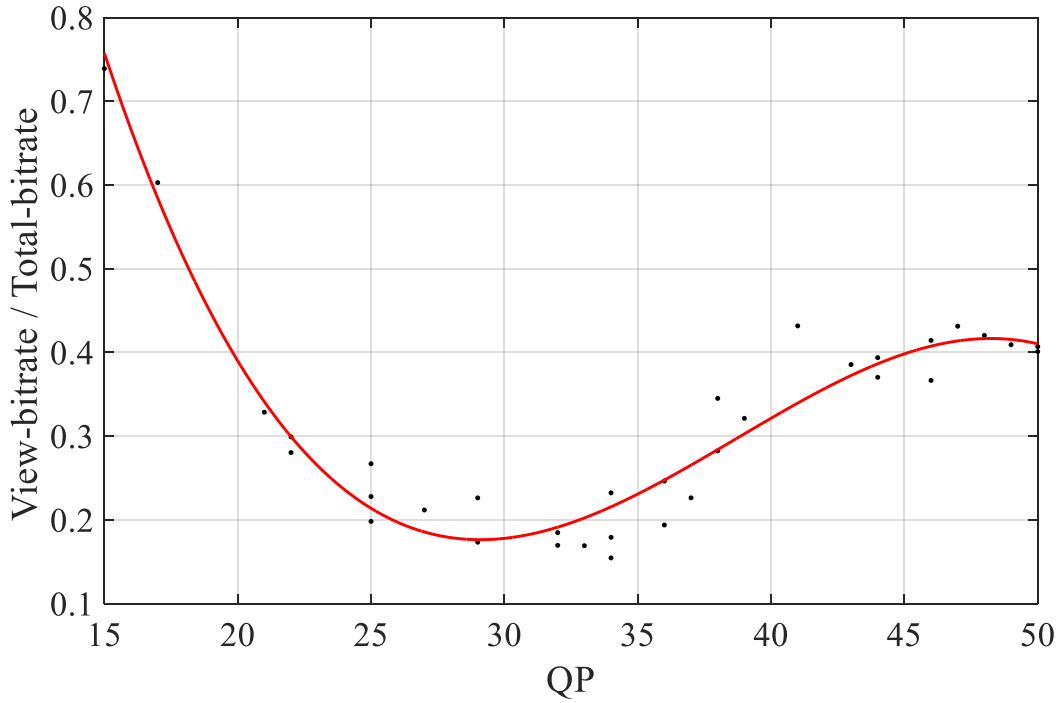


Fig. I.1. The approximate relationship $\frac{View_{Bitrate}}{Total_{Bitrate}} = f(QP)$ for the optimum pairs for *Ballet* sequence with the use of third-order polynomial regression.

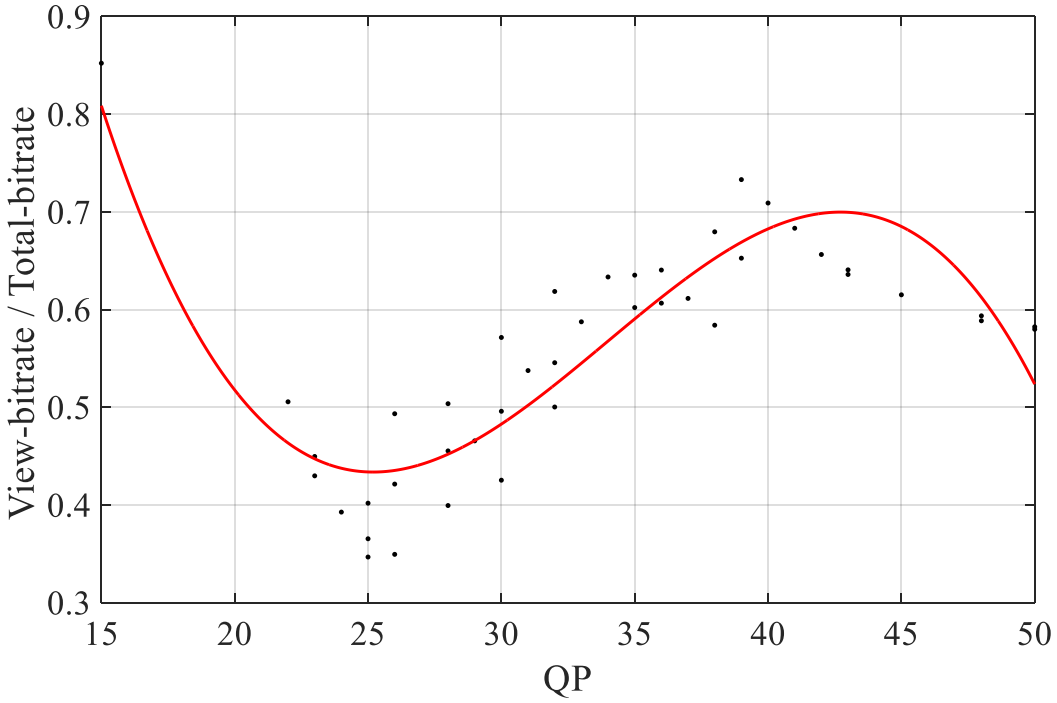


Fig. I.2. The approximate relationship $\frac{View_{Bitrate}}{Total_{Bitrate}} = f(QP)$ for the optimum pairs for *Breakdancers* sequence with the use of third-order polynomial regression.

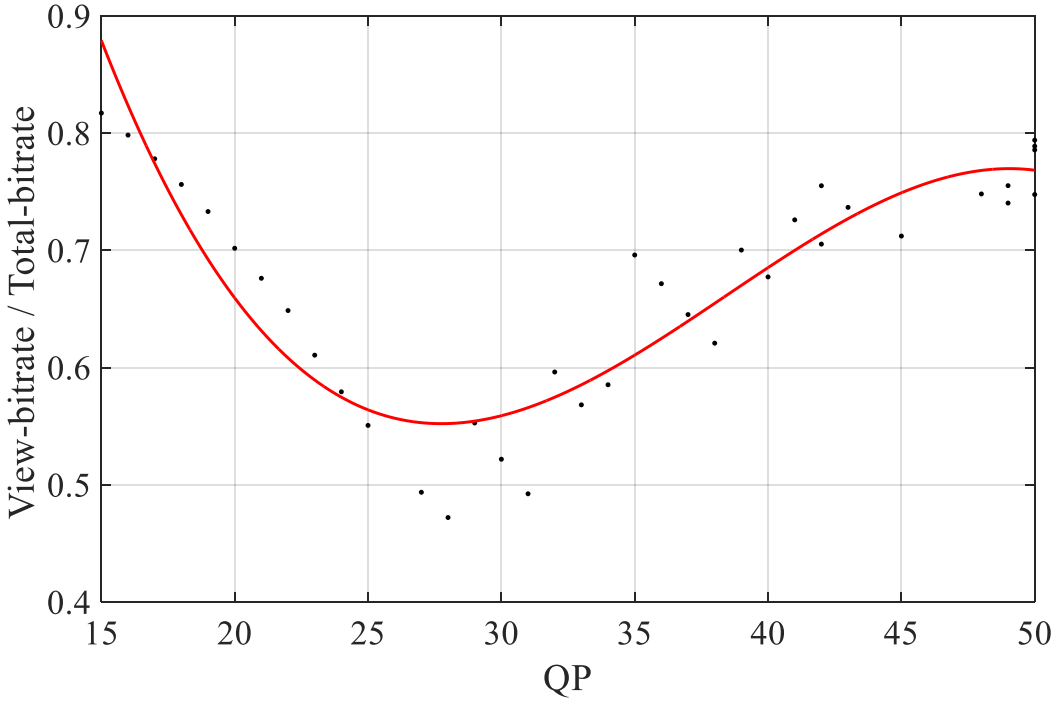


Fig. I.3. The approximate relationship $\frac{View_{Bitrate}}{Total_{Bitrate}} = f(QP)$ for the optimum pairs for *BBB.Butterfly* sequence with the use of third-order polynomial regression.

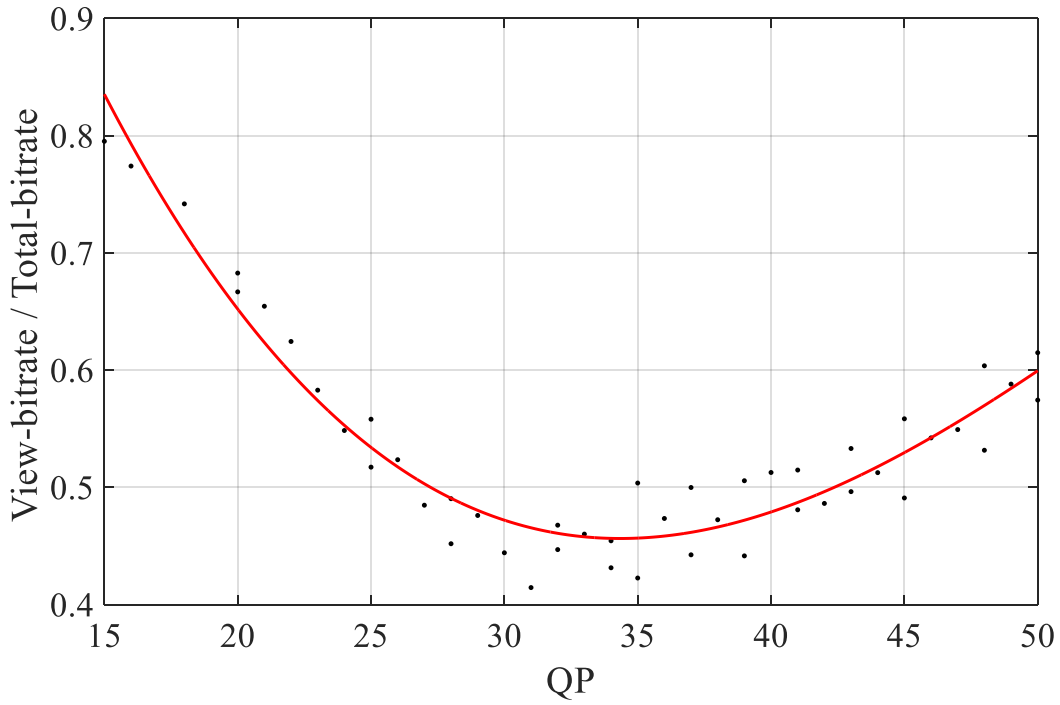


Fig. I.4. The approximate relationship $\frac{View_{Bitrate}}{Total_{Bitrate}} = f(QP)$ for the optimum pairs for *BBB.Flowers* sequence with the use of third-order polynomial regression.

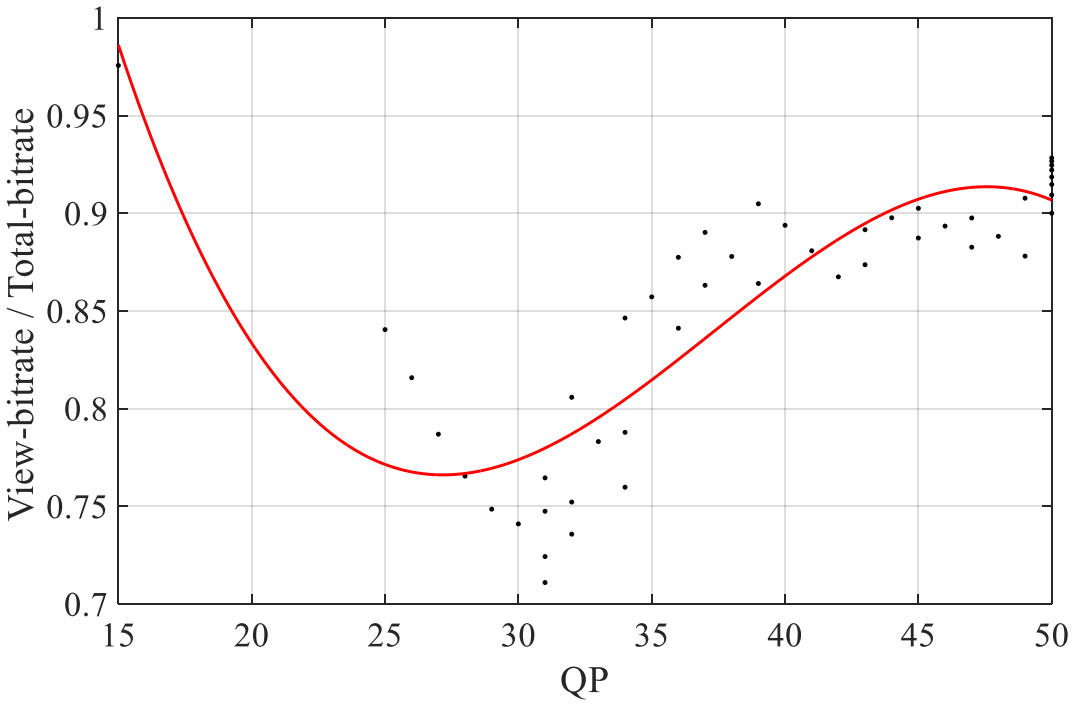


Fig. I.5. The approximate relationship $\frac{View_{Bitrate}}{Total_{Bitrate}} = f(QP)$ for the optimum pairs for *Kermit* sequence with the use of third-order polynomial regression.

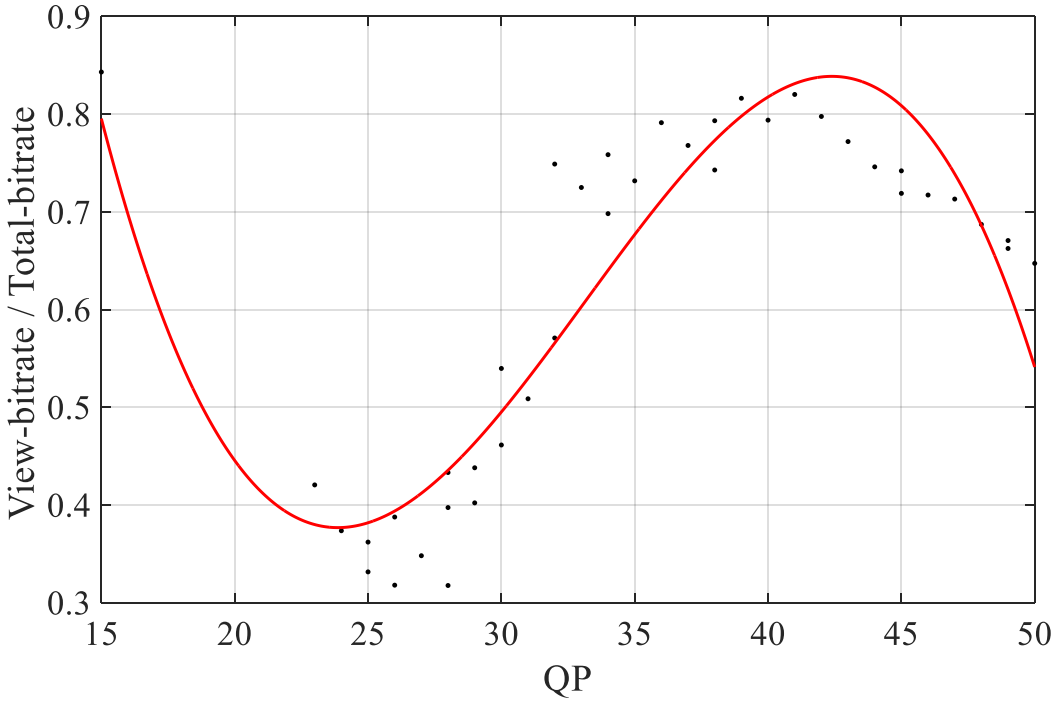


Fig. I.6. The approximate relationship $\frac{View_{Bitrate}}{Total_{Bitrate}} = f(QP)$ for the optimum pairs for *Poznan_CarPark* sequence with the use of third-order polynomial regression.

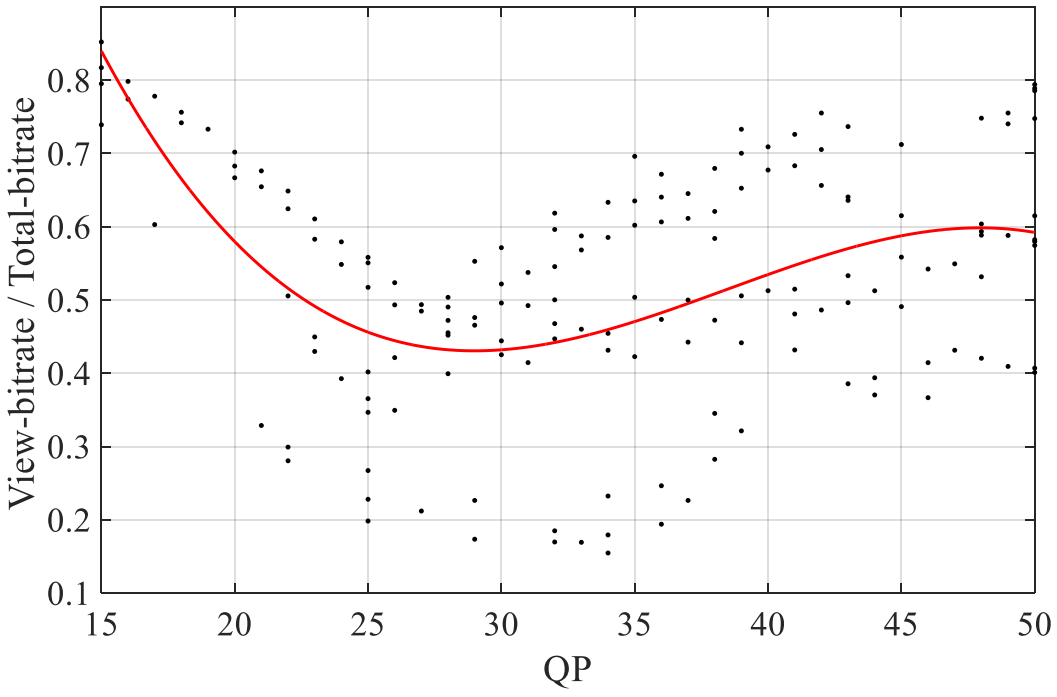


Fig. I.7. The approximate relationship $\frac{View_{Bitrate}}{Total_{Bitrate}} = f(QP)$ for the optimum pairs for training sequences with the use of third-order polynomial regression.

Appendix J

Related to Chapter Seven

Appendix J shows the values of parameters *a*, *b*, and *c* of Eq. 7.1 that were estimated directly from experimental data of the training set (shown in Table 3.2) for many codecs (shown in Table 3.3) by minimization of the square error according to Eq. 7.2. The mean relative approximation errors for total bitrate are also presented in Appendix J.

Table J.1: The model parameters of Eq.7.1 and the average relative approximation error for bitrate for the HEVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	41500	1.07	-0.92	6.11	5.11
Breakdancers	45000	1.08	-3.70	10.94	7.03
BBB.Butterfly	8775	0.84	-3.71	8.49	5.40
BBB.Flowers	51500	1.00	-0.92	4.84	3.30
Kermit	76750	0.95	-6.21	2.81	2.01
Poznan_CarPark	117500	1.18	-7.27	3.65	2.60
Poznan_Block2	13000	0.89	-4.98	2.09	1.36
Poznan_Fencing	90000	1.07	-3.00	8.13	5.11

Table J.2: The model parameters of Eq.7.1 and the average relative approximation error for bitrate for the VVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev.
Ballet	48000	1.17	1.00	3.34	3.02
Breakdancers	63000	1.20	-3.99	2.88	2.19
BBB.Butterfly	10500	0.93	-4.00	4.70	3.57
BBB.Flowers	53000	1.06	-1.00	4.73	3.37
Kermit	98000	1.04	-7.25	1.69	1.67
Poznan_CarPark	140000	1.25	-6.99	1.99	1.46
Poznan_Block2	12902	0.92	-5.43	0.93	0.91
Poznan_Fencing	160000	1.25	1.01	4.57	4.17

Table J.3: The model parameters of Eq.7.1 and the average relative approximation error for bitrate for the MV-HEVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	50000	1.23	-0.01	3.73	2.77
Breakdancers	62000	1.25	-3.22	2.44	1.89
BBB.Butterfly	11000	1.01	-4.16	2.55	2.31
BBB.Flowers	55000	1.11	-0.92	6.40	4.21
Kermit	110000	1.12	-8.49	1.92	2.29
Poznan_CarPark	151000	1.34	-4.00	1.83	1.19
Poznan_Block2	19600	1.03	-4.53	1.17	0.85
Poznan_Fencing	117000	1.22	-2.99	5.87	3.58

Table J.4: The model parameters of Eq.7.1 and the average relative approximation error for bitrate for the 3D-HEVC coding.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev.
Ballet	21500	1.10	-3.00	3.03	3.19
Breakdancers	32500	1.14	-2.00	3.18	2.99
BBB.Butterfly	8050	0.95	-2.33	1.29	1.32
BBB.Flowers	29000	1.02	-2.27	1.88	2.00
Kermit	215000	1.27	-3.00	4.87	2.83
Poznan_CarPark	90000	1.26	-3.00	2.26	1.61
Poznan_Block2	25000	1.07	-3.00	2.58	1.65
Poznan_Fencing	75000	1.17	-3.00	2.63	1.41

Appendix K

Related to Chapter Seven

Appendix K shows the values of parameters *a* and *b* of Eq. 7.3 that were estimated directly from experimental data of the training set (shown in Table 3.2) for many codecs (shown in Table 3.3) by minimization of the square error according to Eq. 7.2. The mean relative approximation errors for total bitrate are also presented in Appendix K.

Table K.1: The model parameters of Eq.7.3 and the average relative approximation error for bitrate for the HEVC coding.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	29995	1.02	6.75	4.24
Breakdancers	47203	1.09	11.02	7.10
BBB.Butterfly	8700	0.84	8.90	5.67
BBB.Flowers	22000	0.83	8.47	5.33
Kermit	210080	1.16	6.31	5.81
Poznan_CarPark	151909	1.23	5.21	3.83
Poznan_Block2	26191	1.03	4.20	3.72
Poznan_Fencing	88300	1.07	8.17	4.78

Table K.2: The model parameters of Eq.7.3 and the average relative approximation error for bitrate for the VVC coding.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev.
Ballet	31000	1.08	5.49	4.58
Breakdancers	66000	1.21	3.07	2.23
BBB.Butterfly	13200	0.98	4.75	3.91
BBB.Flowers	26520	0.92	8.67	5.02
Kermit	290000	1.28	6.42	4.64
Poznan_CarPark	200000	1.32	2.39	1.88
Poznan_Block2	33000	1.12	5.66	4.65
Poznan_Fencing	113000	1.18	6.87	5.46

Table K.3: The model parameters of Eq.7.3 and the average relative approximation error for bitrate for the MV-HEVC coding.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	32001	1.13	6.24	4.41
Breakdancers	64500	1.26	2.44	2.04
BBB.Butterfly	12000	1.03	2.95	2.41
BBB.Flowers	34500	1.02	9.33	5.75
Kermit	322003	1.35	6.82	3.54
Poznan_CarPark	155000	1.35	1.92	1.24
Poznan_Block2	24200	1.07	1.74	1.39
Poznan_Fencing	113000	1.21	6.09	3.86

Table K.4: The model parameters of Eq.7.3 and the average relative approximation error for bitrate for the 3D-HEVC coding.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev.
Ballet	22000	1.11	3.44	3.09
Breakdancers	33000	1.14	4.27	2.60
BBB.Butterfly	5500	0.87	1.35	1.07
BBB.Flowers	23721	0.98	3.17	2.28
Kermit	230000	1.28	5.92	3.22
Poznan_CarPark	85000	1.25	2.29	1.75
Poznan_Block2	23513	1.06	2.55	1.88
Poznan_Fencing	72200	1.16	2.65	1.73

Appendix L

Related to Chapter Seven

Appendix L shows the values of parameter *a* of Eq. 7.4 that were estimated directly from experimental data of the training set (shown in Table 3.2) for many codecs (shown in Table 3.3) by minimization of the square error according to Eq. 7.2. The mean relative approximation errors for total bitrate are also presented in Appendix L.

Table L.1: The model parameter of Eq.7.4 and the average relative approximation error for bitrate for the HEVC coding.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	43821.33	7.60	7.48
Breakdancers	51715.10	10.65	7.59
BBB.Butterfly	27515.55	9.08	12.00
BBB.Flowers	79981.54	11.55	14.17
Kermit	174290.98	6.68	4.96
Poznan_CarPark	90342.20	10.75	6.76
Poznan_Block2	35193.57	5.52	5.91
Poznan_Fencing	103999.16	7.98	5.61

Table L.2: The model parameter of Eq.7.4 and the average relative approximation error for bitrate for the VVC coding.

Sequences	*a*	Relative error [%]	
		mean	std. dev.
Ballet	35446.81	5.90	5.31
Breakdancers	42552.59	6.31	4.91
BBB.Butterfly	22091.97	7.76	8.01
BBB.Flowers	62914.34	7.98	11.96
Kermit	142906.58	6.44	8.52
Poznan_CarPark	83983.58	12.02	8.24
Poznan_Block2	31108.78	5.18	4.88
Poznan_Fencing	83261.60	7.76	5.67

Table L.3: The model parameter of Eq.7.4 and the average relative approximation error for bitrate for the MV-HEVC coding.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	29501.39	6.89	4.49
Breakdancers	34575.44	8.52	5.14
BBB.Butterfly	16287.39	4.57	3.86
BBB.Flowers	50048.73	7.86	5.69
Kermit	114805.57	10.07	10.14
Poznan_CarPark	55746.97	16.07	9.07
Poznan_Block2	28512.29	2.97	1.92
Poznan_Fencing	74460.17	10.26	5.76

Table L.4: The model parameter of Eq.7.4 and the average relative approximation error for bitrate for the 3D-HEVC coding.

Sequences	*a*	Relative error [%]	
		mean	std. dev.
Ballet	21642.35	3.28	3.10
Breakdancers	28380.83	2.74	2.12
BBB.Butterfly	14345.61	11.69	7.93
BBB.Flowers	40621.17	6.94	5.47
Kermit	110324.93	5.91	6.89
Poznan_CarPark	49093.76	8.06	6.41
Poznan_Block2	29031.81	5.28	2.63
Poznan_Fencing	57381.99	2.85	2.75

Appendix M

Related to Chapter Seven

Appendix M shows the values of parameters *a*, *b*, and *c* of Eq. 7.5 that were estimated directly from experimental data of the training set (shown in Table 3.2) for HEVC and VVC coding by minimization of the square error according to Eq. 7.6. The mean relative approximation errors for individual frames of various types are also presented in Appendix M.

Table M.1: The values of the model parameters in Eq.7.5 and the average relative approximation error for I-frames for HEVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	6000.00	1.01	3.00	3.45	1.99
Breakdancers	5000.00	1.03	-2.00	2.18	1.88
BBB.Butterfly	4940.00	0.99	1.87	3.37	2.68
BBB.Flowers	22000.00	1.06	5.27	3.74	2.52
Kermit	88000.00	1.17	3.93	1.90	1.27
Poznan_CarPark	61500.00	1.25	3.51	2.65	1.84
Poznan_Block2	8000.00	0.92	-2.91	1.34	1.98
Poznan_Fencing	37000.00	1.19	8.02	4.61	3.19

Table M.2: The values of the model parameters in Eq.7.5 and the average relative approximation error for P-frames for HEVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	5000.00	1.12	3.00	6.55	4.84
Breakdancers	9920.00	1.21	4.00	2.71	2.24
BBB.Butterfly	5600.00	1.20	4.00	4.77	3.94
BBB.Flowers	9300.00	1.05	3.00	7.26	4.85
Kermit	90100.00	1.38	2.00	6.37	3.53
Poznan_CarPark	32000.00	1.39	3.00	5.76	4.14
Poznan_Block2	1010.00	0.89	-7.00	8.35	5.41
Poznan_Fencing	19100.00	1.22	7.86	6.36	4.41

Table M.3: The values of the model parameters in Eq.7.5 and the average relative approximation error for B0-frames for HEVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	6589.99	1.20	10.32	9.02	7.57
Breakdancers	9700.00	1.23	3.00	5.68	5.50
BBB.Butterfly	4599.99	1.18	6.13	4.45	5.69
BBB.Flowers	8300.00	1.16	8.00	9.93	7.08
Kermit	41000.00	1.32	-14.00	6.68	5.23
Poznan_CarPark	28000.00	1.41	1.00	3.69	3.56
Poznan_Block2	7400.00	1.42	3.00	5.18	4.39
Poznan_Fencing	15000.00	1.22	2.00	11.64	8.49

Table M.4: The values of the model parameters in Eq.7.5 and the average relative approximation error for B1-frames for HEVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	6890.00	1.24	4.00	14.71	7.65
Breakdancers	13600.00	1.32	3.02	6.90	4.33
BBB.Butterfly	4000.00	1.23	3.00	4.96	5.70
BBB.Flowers	10000.00	1.22	10.00	13.65	6.98
Kermit	100000.00	1.59	20.00	15.11	9.41
Poznan_CarPark	46000.00	1.56	11.00	6.74	4.60
Poznan_Block2	5200.00	1.40	10.00	2.49	2.49
Poznan_Fencing	24000.00	1.32	10.00	13.33	9.27

Table M.5: The values of the model parameters in Eq.7.5 and the average relative approximation error for B2-frames for HEVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	7900.00	1.29	1.00	15.93	7.78
Breakdancers	14000.00	1.35	1.00	8.19	5.59
BBB.Butterfly	3550.00	1.26	1.00	5.20	5.41
BBB.Flowers	7000.00	1.21	1.00	17.31	8.68
Kermit	90000.00	1.68	20.00	12.86	8.71
Poznan_CarPark	70000.00	1.72	10.00	10.41	4.96
Poznan_Block2	3200.00	1.37	1.00	3.67	2.62
Poznan_Fencing	25000.00	1.36	10.00	15.85	10.85

Table M.6: The values of the model parameters in Eq.7.5 and the average relative approximation error for B3-frames for HEVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	9900.13	1.36	-5.71	17.42	8.77
Breakdancers	27000.00	1.51	10.00	7.03	3.79
BBB.Butterfly	4500.00	1.37	2.00	7.61	7.26
BBB.Flowers	12500.00	1.42	3.00	19.97	9.45
Kermit	60000.00	1.71	3.00	6.37	5.43
Poznan_CarPark	120000.00	1.90	3.00	10.88	6.02
Poznan_Block2	2000.00	1.34	3.00	4.27	4.00
Poznan_Fencing	33000.00	1.46	3.00	18.38	9.50

Table M.7: The values of the model parameters in Eq.7.5 and the average relative approximation error for I-frames for VVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	2134.00	0.81	-0.58	1.92	2.29
Breakdancers	2530.00	0.88	-3.48	1.35	1.03
BBB.Butterfly	3972.00	0.96	1.00	3.01	1.69
BBB.Flowers	12650.00	0.97	1.45	2.68	2.20
Kermit	77000.00	1.13	2.53	0.70	0.62
Poznan_CarPark	72500.00	1.28	6.57	1.92	1.98
Poznan_Block2	7800.00	0.96	-3.24	1.06	1.01
Poznan_Fencing	32580.00	1.19	8.00	5.63	4.73

Table M.8: The values of the model parameters in Eq.7.5 and the average relative approximation error for P-frames for VVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	3700.00	1.09	4.00	5.51	5.05
Breakdancers	9000.00	1.21	4.00	3.33	2.90
BBB.Butterfly	4980.00	1.21	4.00	5.74	4.92
BBB.Flowers	8355.90	1.07	4.00	5.63	3.31
Kermit	160100.00	1.56	3.50	10.30	6.86
Poznan_CarPark	13000.00	1.23	-8.99	4.00	3.91
Poznan_Block2	2040.00	1.05	-9.00	6.25	3.88
Poznan_Fencing	14893.00	1.20	5.00	7.00	4.10

Table M.9: The values of the model parameters in Eq.7.5 and the average relative approximation error for B0-frames for VVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	4000.00	1.13	7.00	7.87	5.06
Breakdancers	6871.00	1.20	1.00	3.69	3.52
BBB.Butterfly	3500.00	1.18	7.00	8.40	5.36
BBB.Flowers	9500.00	1.23	21.00	6.58	4.88
Kermit	12000.00	1.15	-13.96	6.53	9.00
Poznan_CarPark	23420.00	1.41	4.00	2.91	2.79
Poznan_Block2	1770.00	1.14	-10.00	3.68	3.68
Poznan_Fencing	18000.00	1.28	18.00	6.89	7.16

Table M.10: The values of the model parameters in Eq.7.5 and the average relative approximation error for B1-frames for VVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	5400.00	1.21	10.00	8.15	5.77
Breakdancers	8572.00	1.26	6.00	4.71	4.05
BBB.Butterfly	2000.00	1.12	1.00	9.58	7.07
BBB.Flowers	8050.00	1.22	10.00	8.53	8.33
Kermit	3700.00	1.03	-17.67	4.63	6.19
Poznan_CarPark	29940.00	1.51	4.00	3.18	2.77
Poznan_Block2	2120.00	1.21	2.00	5.55	4.14
Poznan_Fencing	18999.96	1.30	22.67	7.33	7.95

Table M.11: The values of the model parameters in Eq.7.5 and the average relative approximation error for B2-frames for VVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	5950.00	1.25	9.00	7.55	7.15
Breakdancers	8459.00	1.27	1.00	5.29	4.22
BBB.Butterfly	650.00	0.97	-8.00	9.60	8.82
BBB.Flowers	6900.00	1.25	10.00	9.81	9.16
Kermit	3299.99	1.11	-25.33	8.19	9.81
Poznan_CarPark	36800.00	1.61	3.00	4.71	3.63
Poznan_Block2	663.00	1.06	-10.00	5.92	5.59
Poznan_Fencing	18500.00	1.33	16.00	8.99	9.83

Table M.12: The values of the model parameters in Eq.7.5 and the average relative approximation error for B3-frames for VVC.

Sequences	*a*	*b*	*c*	Relative error [%]	
				mean	std. dev
Ballet	7300.00	1.32	1.00	8.73	6.80
Breakdancers	9280.00	1.31	2.00	5.05	3.49
BBB.Butterfly	59.00	0.63	-6.79	8.33	13.15
BBB.Flowers	6990.00	1.34	10.00	12.48	7.07
Kermit	20000.00	1.57	-70.00	17.73	11.34
Poznan_CarPark	36100.00	1.68	1.00	10.83	9.04
Poznan_Block2	69.99	0.70	-7.20	7.46	5.50
Poznan_Fencing	21200.00	1.41	1.00	10.05	10.29

Appendix N

Related to Chapter Seven

Appendix N shows the values of parameters *a* and *b* of Eq. 7.7 that were estimated directly from experimental data of the training set (shown in Table 3.2) for HEVC and VVC coding by minimization of the square error according to Eq. 7.6. The mean relative approximation errors for individual frames of various types are also presented in Appendix N.

Table N.1: The values of the model parameters in Eq.7.7 and the average relative approximation error for I-frames for HEVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	5970	1.01	3.40	2.02
Breakdancers	10460	1.20	6.48	4.02
BBB.Butterfly	5300	1.00	4.60	2.26
BBB.Flowers	18000	1.02	4.38	2.79
Kermit	82000	1.16	2.12	1.43
Poznan_CarPark	61000	1.24	2.95	1.64
Poznan_Block2	26500	1.20	6.69	3.93
Poznan_Fencing	25000	1.11	5.21	3.40

Table N.2: The values of the model parameters in Eq.7.7 and the average relative approximation error for P-frames for HEVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	5000	1.12	6.55	4.84
Breakdancers	10100	1.22	3.46	2.63
BBB.Butterfly	5300	1.19	4.78	4.16
BBB.Flowers	9300	1.05	7.26	4.85
Kermit	90000	1.38	6.54	3.85
Poznan_CarPark	32000	1.39	5.76	4.14
Poznan_Block2	11700	1.40	10.87	6.41
Poznan_Fencing	18500	1.22	8.29	5.90

Table N.3: The values of the model parameters in Eq.7.7 and the average relative approximation error for B0-frames for HEVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	5320	1.16	10.73	7.78
Breakdancers	9700	1.23	5.68	5.50
BBB.Butterfly	4300	1.17	4.78	6.14
BBB.Flowers	6500	1.11	11.40	7.27
Kermit	87000	1.47	11.52	6.86
Poznan_CarPark	28800	1.41	3.60	3.57
Poznan_Block2	7400	1.42	5.18	4.39
Poznan_Fencing	11000	1.14	11.20	10.40

Table N.4: The values of the model parameters in Eq.7.7 and the average relative approximation error for B1-frames for HEVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	7000	1.25	14.82	7.64
Breakdancers	13500	1.32	6.91	4.41
BBB.Butterfly	4000	1.23	4.96	5.70
BBB.Flowers	8000	1.18	14.66	7.43
Kermit	100000	1.60	16.37	11.03
Poznan_CarPark	45000	1.56	7.37	4.87
Poznan_Block2	4700	1.38	2.72	2.41
Poznan_Fencing	19000	1.27	14.30	9.38

Table N.5: The values of the model parameters in Eq.7.7 and the average relative approximation error for B2-frames for HEVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	8000	1.29	15.83	7.80
Breakdancers	14000	1.35	8.27	5.78
BBB.Butterfly	3800	1.27	5.28	5.20
BBB.Flowers	7300	1.22	17.07	8.93
Kermit	90000	1.68	13.79	9.69
Poznan_CarPark	70000	1.72	10.57	5.14
Poznan_Block2	3200	1.37	3.58	2.64
Poznan_Fencing	23000	1.35	16.40	10.36

Table N.6: The values of the model parameters in Eq.7.7 and the average relative approximation error for B3-frames for HEVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	9600	1.36	17.83	8.49
Breakdancers	27300	1.52	7.10	4.43
BBB.Butterfly	4700	1.38	7.82	7.33
BBB.Flowers	12500	1.42	19.97	9.45
Kermit	60000	1.71	6.37	5.43
Poznan_CarPark	120000	1.90	10.88	6.02
Poznan_Block2	2000	1.34	4.27	4.00
Poznan_Fencing	33000	1.46	18.38	9.50

Table N.7: The values of the model parameters in Eq.7.7 and the average relative approximation error for I-frames for VVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	4700	0.99	4.92	2.42
Breakdancers	9900	1.18	8.45	6.02
BBB.Butterfly	4600	0.98	3.80	2.93
BBB.Flowers	16300	1.02	2.80	1.62
Kermit	79400	1.14	0.84	0.57
Poznan_CarPark	60000	1.23	2.14	1.27
Poznan_Block2	26000	1.24	8.31	4.97
Poznan_Fencing	23020	1.11	5.69	4.97

Table N.8: The values of the model parameters in Eq.7.7 and the average relative approximation error for P-frames for VVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	3596	1.08	6.07	4.77
Breakdancers	8723	1.20	3.40	3.06
BBB.Butterfly	4710	1.20	5.75	5.11
BBB.Flowers	7280	1.04	5.70	4.42
Kermit	120000	1.49	10.36	6.97
Poznan_CarPark	31000	1.42	8.27	5.87
Poznan_Block2	9710	1.37	11.24	8.20
Poznan_Fencing	14210	1.19	7.26	4.52

Table N.9: The values of the model parameters in Eq.7.7 and the average relative approximation error for B0-frames for VVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	3210	1.09	9.10	5.26
Breakdancers	8220	1.23	3.71	3.61
BBB.Butterfly	3110	1.17	8.87	5.54
BBB.Flowers	4650	1.09	10.03	6.19
Kermit	75940	1.54	17.14	10.54
Poznan_CarPark	22450	1.40	2.92	2.69
Poznan_Block2	6200	1.39	9.34	6.99
Poznan_Fencing	12600	1.22	8.93	8.38

Table N.10: The values of the model parameters in Eq.7.7 and the average relative approximation error for B1-frames for VVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	4390	1.18	9.25	7.01
Breakdancers	7499	1.23	4.79	4.29
BBB.Butterfly	2080	1.13	9.66	7.40
BBB.Flowers	6300	1.18	9.73	9.13
Kermit	44000	1.50	15.87	10.74
Poznan_CarPark	30200	1.51	3.19	2.80
Poznan_Block2	2120	1.21	5.82	4.31
Poznan_Fencing	14710	1.26	9.21	10.11

Table N.11: The values of the model parameters in Eq.7.7 and the average relative approximation error for B2-frames for VVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	4942	1.21	8.06	7.84
Breakdancers	8992	1.28	5.31	4.50
BBB.Butterfly	1659	1.14	12.75	8.65
BBB.Flowers	5860	1.23	10.39	10.33
Kermit	40100	1.58	15.91	10.22
Poznan_CarPark	36800	1.61	4.71	3.63
Poznan_Block2	1231	1.16	8.78	6.60
Poznan_Fencing	15700	1.30	10.33	11.16

Table N.12: The values of the model parameters in Eq.7.7 and the average relative approximation error for B3-frames for VVC.

Sequences	*a*	*b*	Relative error [%]	
			mean	std. dev
Ballet	7200	1.32	8.80	6.89
Breakdancers	9350	1.31	5.09	3.54
BBB.Butterfly	790	1.06	18.15	12.84
BBB.Flowers	6101	1.32	12.91	7.25
Kermit	29000	1.63	19.72	11.79
Poznan_CarPark	33060	1.66	10.97	9.25
Poznan_Block2	402	0.99	12.67	11.05
Poznan_Fencing	21340	1.41	10.16	10.01

Appendix O

Related to Chapter Seven

Appendix O shows the values of parameter *a* of Eq. 7.8 that were estimated directly from experimental data of the training set (shown in Table 3.2) for HEVC and VVC coding by minimization of the square error according to Eq. 7.6. The mean relative approximation errors for individual frames of various types are also presented in Appendix O.

Table O.1: The values of the model parameter in Eq.7.8 and the average relative approximation error for I-frames for HEVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	11309.25	20.32	10.89
Breakdancers	11692.66	8.35	5.79
BBB.Butterfly	11512.61	16.96	12.42
BBB.Flowers	34384.75	21.34	10.14
Kermit	107243.00	8.65	4.32
Poznan_CarPark	59851.38	2.91	1.78
Poznan_Block2	28338.55	8.73	7.53
Poznan_Fencing	38095.70	14.41	6.01

Table O.2: The values of the model parameter in Eq.7.8 and the average relative approximation error for P-frames for HEVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	8153.56	13.59	9.12
Breakdancers	11115.36	4.68	3.68
BBB.Butterfly	6560.30	6.00	4.56
BBB.Flowers	19789.79	18.12	12.73
Kermit	48171.89	9.55	12.39
Poznan_CarPark	16526.42	8.38	8.71
Poznan_Block2	6124.33	4.31	9.16
Poznan_Fencing	20250.50	9.33	6.89

Table O.3: The values of the model parameter in Eq.7.8 and the average relative approximation error for B0-frames for HEVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	7307.67	13.66	9.42
Breakdancers	10078.41	5.48	5.19
BBB.Butterfly	5801.91	8.00	8.58
BBB.Flowers	11498.02	16.35	13.36
Kermit	30717.88	10.75	15.84
Poznan_CarPark	12654.59	15.46	10.76
Poznan_Block2	3228.80	10.99	10.39
Poznan_Fencing	16726.39	11.54	8.58

Table O.4: The values of the model parameter in Eq.7.8 and the average relative approximation error for B1-frames for HEVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	6785.96	14.84	7.79
Breakdancers	9078.77	10.20	6.48
BBB.Butterfly	4298.84	5.09	5.33
BBB.Flowers	10592.83	15.29	8.47
Kermit	18922.69	9.09	12.91
Poznan_CarPark	9496.71	23.34	13.54
Poznan_Block2	2400.36	9.46	6.51
Poznan_Fencing	15599.50	15.18	8.87

Table O.5: The values of the model parameter in Eq.7.8 and the average relative approximation error for B2-frames for HEVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	6718.83	16.33	11.23
Breakdancers	8216.88	13.26	8.04
BBB.Butterfly	3195.20	5.35	6.72
BBB.Flowers	7932.12	16.94	8.14
Kermit	10522.87	14.39	16.28
Poznan_CarPark	6359.38	31.86	16.82
Poznan_Block2	1736.35	9.41	6.71
Poznan_Fencing	13869.89	18.65	12.25

Table O.6: The values of the model parameter in Eq.7.8 and the average relative approximation error for B3-frames for HEVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	5910.70	21.54	16.86
Breakdancers	6955.89	17.24	10.56
BBB.Butterfly	2267.93	7.50	10.68
BBB.Flowers	5190.68	23.96	13.39
Kermit	4942.00	22.50	23.09
Poznan_CarPark	4090.75	30.48	19.53
Poznan_Block2	1269.58	6.18	5.22
Poznan_Fencing	11127.37	25.67	15.15

Table O.7: The values of the model parameter in Eq.7.8 and the average relative approximation error for I-frames for VVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	9377.27	22.59	12.70
Breakdancers	11386.34	10.12	7.71
BBB.Butterfly	10151.36	18.81	11.78
BBB.Flowers	30562.58	20.15	9.60
Kermit	108075.08	8.34	5.05
Poznan_CarPark	61090.25	2.22	1.32
Poznan_Block2	25680.22	8.25	5.90
Poznan_Fencing	33848.35	15.05	7.20

Table O.8: The values of the model parameter in Eq.7.8 and the average relative approximation error for P-frames for VVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	6366.24	16.19	8.23
Breakdancers	10008.09	4.92	4.20
BBB.Butterfly	5567.03	6.49	4.85
BBB.Flowers	16131.59	18.38	11.50
Kermit	37327.00	16.74	19.33
Poznan_CarPark	14877.14	8.88	11.14
Poznan_Block2	5580.48	5.72	12.12
Poznan_Fencing	17301.90	8.73	6.86

Table O.9: The values of the model parameter in Eq.7.8 and the average relative approximation error for B0-frames for VVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	6540.49	16.00	15.16
Breakdancers	8456.24	3.98	3.61
BBB.Butterfly	4240.55	12.01	8.09
BBB.Flowers	9262.90	16.36	17.57
Kermit	20561.99	14.92	20.54
Poznan_CarPark	11026.17	14.42	9.35
Poznan_Block2	3160.77	5.96	9.85
Poznan_Fencing	14013.56	9.83	9.06

Table O.10 The values of the model parameter in Eq.7.8 and the average relative approximation error for B1-frames for VVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	5915.77	10.60	8.31
Breakdancers	7900.67	4.82	4.15
BBB.Butterfly	3274.06	12.47	8.90
BBB.Flowers	8798.47	11.13	13.55
Kermit	13339.37	13.48	14.92
Poznan_CarPark	7757.64	20.98	13.68
Poznan_Block2	2433.49	5.58	4.88
Poznan_Fencing	13318.17	9.56	9.53

Table O.11: The values of the model parameter in Eq.7.8 and the average relative approximation error for B2-frames for VVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	5691.76	8.85	7.78
Breakdancers	7254.17	5.41	5.20
BBB.Butterfly	2481.74	14.50	11.31
BBB.Flowers	6312.09	10.49	10.44
Kermit	7128.76	18.62	18.43
Poznan_CarPark	5147.84	27.04	17.64
Poznan_Block2	1752.88	8.25	9.68
Poznan_Fencing	11292.51	11.06	10.68

Table O.12: The values of the model parameter in Eq.7.8 and the average relative approximation error for B3-frames for VVC.

Sequences	*a*	Relative error [%]	
		mean	std. dev
Ballet	4937.42	9.69	7.18
Breakdancers	6354.60	7.88	6.32
BBB.Butterfly	1931.62	21.01	10.82
BBB.Flowers	3861.36	15.52	10.16
Kermit	3416.23	26.78	22.79
Poznan_CarPark	3343.63	25.37	23.15
Poznan_Block2	1226.68	14.63	17.68
Poznan_Fencing	8617.46	17.82	12.98