VIDEO ENCODING QUALITY AND BIT RATE PREDICTION, AND ITS APPLICATION IN RESOLUTION, AND FRAME-RATE ADAPTIVE ENCODING

VIDEO ENCODING QUALITY AND BIT RATE PREDICTION, AND ITS APPLICATION IN RESOLUTION, AND FRAME-RATE ADAPTIVE ENCODING

BY

MARYAM JENAB, M.Sc., B.Sc.

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Doctor of Philosophy (2023) McMaster University

(Electrical & Computer Engineering) Hamilton, Ontario, Canada

TITLE:	Video Encoding Quality and Bit Rate Prediction, and Its Application in Resolution, and Frame-Rate Adaptive Encoding
AUTHOR:	Maryam Jenab M.Sc. (Electrical Engineering), B.Sc. (Electrical Engineering), Isfahan University of Technology, Isfahan, Iran
SUPERVISOR:	Prof. Shahram Shirani
NUMBER OF PAGES:	xvi, 130

To my beloved husband Sahand and my dearest parents Mitra and Hossein

Lay Abstract

The quality of images and videos is crucial for a satisfying visual experience. However, during compression and transmission, images often undergo degradation. This research focuses on predicting image distortion and bit rates without actually compressing the video. Rate control (RC) is essential in compression algorithms to minimize distortion while adhering to bit rate constraints. However, conventional RC models struggle with assigning appropriate bit rates to frames with fast motion and scene changes. To overcome this limitation, we propose a novel convolutional neural network (CNN)-based method for bit rate prediction. Our approach includes patch-level and frame-level predictions, utilizing spatial and temporal features extracted by the CNN network. Furthermore, we explore the impact of reducing spatial and temporal redundancy before encoding to improve compression efficiency. We propose two adaptive encoding methods: a machine learning approach and a CNN-based spatio-temporally adaptive encoder. These methods predict the optimal encoding parameters, such as the minimum quantization parameter (QP), for downscaled video frames. Our research introduces innovative CNN-based methods for predicting distortion, bit rates, and optimizing compression efficiency. These methods contribute to improving the quality of video communication and have the potential to enhance the viewing experience for users.

Abstract

The prediction of perceived quality of images by the human visual system (HVS) has gained considerable interest. The HVS serves as the ultimate destination for most videos. However, prior to reaching its destination, an image undergoes degradation through compression and transmission. Given the bitrate limitations associated with video transmission and storage, video compression plays a crucial role in image communication. In this thesis, we present a novel convolution neural network (CNN)-based method for predicting the distortion of encoded video without performing compression. We have employed strategies to overcome the limitation of the dataset size. First, instead of employing scored samples based on mean opinion scores (MOS), we utilize a closely related index, Video Multimethod Assessment Fusion (VMAF), which aligns with HVS perceived quality scores and is easier to generate compared to MOS. Second, we train our CNN at patch (square area in a frame) level to increase the number of training samples and enhance the prediction accuracy.

The patch-level quality predictor comprises a deep neural network (DNN) network consisting of a series of CNN and pooling layers, followed by a regressor with fully connected layers. This CNN network accepts patches of uncompressed video

frames and patches from motion estimation (ME) maps as input and generates quality scores (VMAF) for the patches. We have introduced and compared three patch-wise to frame-wise transformations for frame-level quality prediction. We proposed a method for predicting perceived quality, both for intra-frames and inter-frames. The results demonstrate the excellent performance of our innovative frame-level compression quality prediction method.

Rate control (RC) plays a crucial role in compression algorithms by minimizing distortion while adhering to bit rate constraints. RC operates by bit allocation at two levels: block or frame. Despite advancements in compression algorithms, conventional RC models still struggle with assigning bit rates to frames with fast motions and scene changes. To address this issue, we propose a novel CNN-based method for bit rate prediction, which overcomes the limitations of traditional RC models. Our approach consists of two phases: patch-level and frame-level bit rate prediction. The proposed CNN network includes CNN layers for extracting spatial and temporal features from video frames, and pooling layers to prevent overfitting. These CNN and pooling layers are followed by a regressor that predicts the bit rate of patches based on their extracted features. We utilize the trained patch-wise CNN bit rate predictor for frame-level bit rate prediction by feeding the extracted features of patches into the regressor. For intra-frame bit rate prediction, we employ frame patches to extract spatial features. For inter-frame bit rate prediction, in addition to spatial features, we incorporate motion estimation (ME) maps and extract temporal patch features. Notably, our proposed method is the first end-to-end CNN-based RC method that operates without relying on hand-crafted features. By not considering the previous frames' encoding information, such as their bit rates, our approach successfully

predicts the bit rate even during scene changes.

Previous research has demonstrated that reducing spatial and temporal redundancy before encoding can improve compression performance. Essentially, if a video frame is downscaled before encoding and upscaled after decoding, the resulting frame exhibits higher quality compared to a conventionally encoded frame at the same bit rate. In a temporally adaptive encoder, the video's frame rate is down-converted before encoding and up-converted after decoding. However, the impact of redundancy reduction on compression efficiency varies depending on the video content. Downscaling or down-converting can positively or negatively affect compression performance, contingent upon the characteristics of the video. Additionally, the bit rate used for video encoding is a critical factor in adaptive encoding. Empirical results indicate that downscaling or down-converting at a low bit rate enhances the quality of the compressed video while maintaining the same overall bit rate. Consequently, we propose two spatial/temporal adaptive encoding methods: a machine learning approach and a CNN-based spatio-temporally adaptive encoder. Our machine learning method predicts the minimum quantization parameter (QP) at the bit rate intersection where encoding with downscaled video outperforms conventional encoding, leveraging hand-crafted features of frames. Meanwhile, the CNN-based adaptive encoding method predicts the QP at the intersection based on spatial and temporal features extracted by the CNN network. Both of our proposed methods surpass the performance of state-of-the-art adaptive encoding techniques. The CNN-based adaptive encoder particularly excels in intra-frame encoding compared to using hand-crafted features and machine learning.

Acknowledgements

I would like to express my gratitude toward my supervisor, Dr. Shahram Shirani, for his invaluable guidance, unwavering support, and encouragement throughout my Ph.D. studies. My appreciation to my committee members Dr. Tim Davidson and Dr. Jim Reilly for their constructive feedback and insightful suggestions. Finally, I would like to give a very special thanks to my family whose boundless love and unwavering support have made it possible for me to excel in my studies.

Notation and Abbreviations

AVC	Advanced Video Encoding
CNN	Convolutional Neural Network
DNN	Deep Neural Network
GLCM	Gray Level Co-occurrence Matrices
GOP	Group of pictures
HEVC	High Efficiency Video Encoding
HVS	Human Visual System
IQA	Image Quality Assessment
MAE	Mean Absolute Error
ME	Motion Estimation
ML	Machine Learning
MSE	Mean Square Error
NCC	Normalized Cross-Correlation

PSNR	Peak Signal-to-Noise Ratio
RC	Rate Control
QP	Quantization Parameter
Qs	Quantization Step
SAE	Spatially Adaptive Encoding
SVM	Support Vector Machine
SSIM	Structural Similarity Index Measure
UHD	Ultra High Definition
VMAF	Video Multimethod Assessment Fusion

Contents

List of Figures

Chapter 1

Introduction

With the increasing popularity of digital videos, there has been a significant emphasis on processing videos with high resolution, frame rate, and dynamic color range, all while considering the constraints of bandwidth and storage. To address these challenges, compression algorithms have emerged as a means to optimize the trade-off between bit rate and distortion in the resulting videos. The primary objective of this thesis is to tackle a fascinating problem within compression algorithms: optimizing the encoder's performance based on the characteristics of the video content. One crucial aspect in improving the performance of a compression algorithm is the ability to accurately predict the bit rate and quality of the compressed video. Such predictions serve as valuable tools in enhancing the overall performance and efficiency of the compression process.

1.1 Quality prediction

Digital videos have become an integral part of modern life, finding applications in communication, entertainment, and video streaming. However, during the transmission and compression, videos can suffer from various types of degradation such as noise and blockiness. These degraded images can lead to annoyance and dissatisfaction when viewed by humans, who are the ultimate destination for most transmitted or stored videos. Therefore, it is essential to assess the quality of the compressed video and provide feedback in order to fine-tune video pre/post-processing and compression algorithms.

Assessing video quality by relying on human evaluation alone is often expensive and impractical. Consequently, different approaches have been developed to assess video quality. Image quality assessment (IQA) aims to predict the quality of images or videos. Depending on the available reference information, IQA methods can be categorized as full-reference, reduced-reference, or no-reference assessments. Full-reference methods compare the received image with the original reference image, while reduced-reference methods utilize both the degraded image and some features extracted from the reference image. No-reference methods play a crucial role in predicting video quality as they do not rely on any reference information. In contrast, predicting video quality before video compression requires only reference video information, without accessing the degraded video at the destination.

1.2 Rate Control

Advanced video coding (AVC) and high efficiency video coding (HEVC) are widely adopted video compression standards. Rate control, which aims to maintain the bit rate within a predefined range to prevent overflow and underflow, is a critical aspect of all compression algorithms. Rate control is implemented at various levels, including the group of pictures (GOP) level, frame level, and block level. Encoders utilize the empirical bit rates of blocks and frames already encoded to perform subsequent rate control. While this technique proves beneficial for smooth videos with slow movement and gradual changes, encoders face challenges when confronted with abrupt changes in frame sequences or scene transitions.

The primary objective of an encoder is to optimize rate-distortion, ensuring that the bit rate remains at an accessible level with minimal distortion. It is crucial for the encoder to maintain consistent frame distortion to minimize perceived video distortion. However, the encoder typically struggles to track fast motions, occlusions, or scene changes, leading to inadequate rate control. Unequal allocation of bit rates, such as assigning a higher bit rate to specific frames, reduces the bit rate available for subsequent frames. This inconsistency in perceived distortion across the frame sequence ultimately degrades video quality.

To address these limitations, some research endeavors have focused on modeling bit rates by leveraging extracted frame features [32]. Additionally, some studies have employed rate-quantization models instead of rate-distortion models [31]. However, these RC methods still rely on the correlation between blocks/frames, facing challenges with unrelated successive frames. Moreover, utilizing engineering-crafted features poses the risk of inaccuracies when dealing with more complex frames.

1.3 Spatially and temporally adaptive encoding

The ever-growing demand for a more realistic visual experience has led to the emergence of immersive images, high-resolution frames, extended color dynamic range, and videos with higher frame rates. While these advancements contribute to higher perceived video quality, the limitations of bandwidth and storage impose restrictions. Video encoding algorithms, such as HEVC and AVC, employ lossy compression techniques to minimize the video bit rate while attempting to minimize video degradation.

To adhere to bit rate constraints, encoders often perform low bit rate compressions, which can result in severely distorted images. Previous research has demonstrated that encoding videos at low bit rates can result in highly degraded videos [6]. Resampling videos before encoding can help reduce redundancy and improve rate-distortion performance. However, spatial or temporal downscaling introduces scaling distortions to the restored video [22], making it crucial to select an optimized scaling ratio that minimizes overall degradation. Another important factor in adaptive encoding is determining the bit rate at which scaling distortion exceeds the encoding distortion.

Additionally, downscaling images and then upscaling them after decoding can introduce blockiness, which can be influenced by the type of upscaling filter used. In videos with smooth changes, downsampling the frame rate before encoding and performing frame interpolation afterwards can enhance perceived quality at low bit rates. Hence, to decide whether spatial or temporal resampling is necessary, it is essential to extract features related to the textural content and motion of frames and utilize them to develop a resolution and frame rate parameter optimization model.

1.4 Contributions

This thesis showcases the outcome of our investigation into optimizing compression parameters. In Chapter 3, we propose a machine learning-based spatially adaptive encoding technique to enhance the encoding performance of H.264/AVC and H.265/HEVC. We presented the results of our spatially adaptive encoding method for AVC in a published paper at ICME [36]. In Chapter 4, our focus shifts to patch-based perceived quality prediction of compression algorithms using deep CNN. The results of Chapter 4 are also included in an accepted paper [35]. Chapter 5 introduces an innovative patch-based pre-encoded bit rate predictor and control method utilizing deep CNN. The findings from Chapter 4 and 5 will be submitted as a journal paper. Finally, in Chapter 6, we propose a CNN-based end-to-end temporally and spatially adaptive encoding method that surpasses previous adaptive encoding approaches relying on hand-crafted features.

1.5 Thesis Outline

This dissertation is structured as follows. In the Introduction, we have provided an overview of video distortion assessment, bit rate control for compression algorithms, and the importance of video compression parameter optimization and adaptive encoding. Chapter 2 offers a comprehensive review of previous research on distortion assessment, bit rate prediction, rate control, and spatially/temporally adaptive encoding. Chapter 3 is dedicated to our proposed machine learning-based spatially adaptive encoding method for H.264/AVC and H.265/HEVC encoders. Chapters 4

and 5 focus on presenting an end-to-end deep CNN-based approach for encoding distortion and bit rate prediction, eliminating the need for engineering-crafted features. In Chapter 6, we delve into the details of our CNN-based spatially and temporally adaptive encoding method specifically designed for high-definition frames using the HEVC compression algorithm.

Chapter 2

Literature Review

The literature review in this thesis is structured into four parts: the overall view of video encoding, distortion prediction/assessment, rate prediction/control, and adaptive video encoding. The first part provides an overview of the fundamentals of video encoding. The second part focuses on the metrics developed for measuring the distortion caused by encoding. Various types of distortion assessment and prediction techniques are surveyed, with an emphasis on those that take into account human visual perception. The review encompasses the theoretical foundations of human visual perception-based distortion metrics. In the third part, a review is presented on optimizing rate-distortion and controlling the rate within the encoder. This includes a discussion of rate control methods developed across different generations of encoders, highlighting their evolution and advancements. Finally, the studies related to the effects of spatial and temporal resampling of videos before and after encoding and decoding are reviewed. Prior research has shown that resampling frame sequences spatially or temporally can improve encoder performance at low bit rates. The last part of this chapter is dedicated to reviewing adaptive spatiotemporal encoding methods

and their impact on distortion and rate control. By organizing the literature review into these four parts, a comprehensive understanding of the research landscape in video encoding, distortion prediction/assessment, rate prediction/control, and adaptive video encoding is provided.

2.1 An Introduction to Video Encoding

The emergence of internet-based communications and mobile technology has made compression standards indispensable for storing, transmitting, and receiving visual data. These standards have been developed by two bodies, namely ITU-T and ISO/IEC. ITU-T has introduced compression standards such as H.261 [1] and H.263 [72] encoders. On the other hand, ISO/IEC has contributed to the development of MPEG-1 [2] and MPEG-4 Visual [3]. Additionally, ITU-T and ISO/IEC collaboration has led to the creation of H.262/MPEG-2 Video [59], H.264/MPEG-4 Advanced Video Coding (AVC) [57], and H.265/HEVC [80] encoding standards. Among these standards, AVC and HEVC are the most recent and have significantly improved encoding performance. They have pushed the boundaries of encoding capabilities. Encoder users have access to several parameters for controlling the encoding process, which will be further explained in the subsequent sections.

A video consists of a sequence of two-dimensional images/frames in the time domain. To achieve compression, the video is divided into groups of pictures (GOP). Each GOP begins with a key frame. The key frame is encoded independently, utilizing intra-frame coding (I-frame). Following the key frame, a series of frames are encoded using inter-frame prediction. These frames include predicted frames (P-frames) and bidirectional frames (B-frames). A P-frame is predicted based on the

spatial information of the preceding I-frame or P-frame, along with motion estimation (ME) between the P-frame and the reference frame. The ME estimates the motion vectors to account for the movement between frames, allowing for more efficient encoding. Similarly, a B-frame is encoded using information from the current GOP's I-frame, the next GOP's I-frame, and motion estimation in both forward and backward directions. This enables the B-frame to take advantage of temporal and spatial correlations between frames. In summary, the coding process involves encoding the I-frame based on spatial information, predicting P-frames using motion estimation and spatial information, and encoding B-frames by considering spatial and temporal information from neighboring frames. This hierarchical structure of frame types and their interdependencies contributes to efficient video compression.

The human visual system perceives images based on their brightness and color, with a higher sensitivity to brightness. To exploit this characteristic, conventional coding standards such as H.264/AVC and HEVC use the YCbCr image format. It includes a luminance component (Y) representing brightness and two color difference components (Cb and Cr) representing color information. The chroma components have a quarter size compared to the luma component. In the H.264/AVC coding standard, frames are divided into macroblocks of size 16×16 pixels. These macroblocks can further be divided into smaller blocks if they contain significant details. For smooth areas, the 16×16 block size may be maintained. The HEVC coding standard introduces a more advanced block structure. In HEVC, the I-frame is divided into coding tree units (CTUs), each containing three types of blocks: coding blocks (CUs), predicted blocks (PUs), and transform blocks (TUs). A coding block (CU) represents a portion of the image and carries spatial information. The maximum size

of a CU in HEVC is 64×64 pixels. A predicted block (PU) contains motion estimation/motion compensation (ME/MC) information for a CTU, which enables efficient prediction of motion between frames. The transform block (TU) carries quantized coefficient information after transformation. Overall, the use of macroblocks and coding tree blocks allows for efficient representation and compression of video frames. The hierarchical structure of these blocks enables adaptive coding and prediction based on the complexity of different regions within the frame.

The key parameter for controlling AVC and HEVC encoders is the average quantization parameter (QP) applied to macroblocks or coding tree units (CTUs), also known as the frame QP. The frame QP is selected from a range of values between 1 and 51. QPs play a crucial role in balancing the trade-off between encoded frame quality and bit rate. A higher QP value leads to lower bit rate and lower quality of the encoded frame. The QP value is closely related to another important encoder parameter called the quantization step (Qs). When the encoder predicts a block or frame, it computes the difference between the original data and the predicted data, which is known as the residue block or residue frame. The Qs value is used for quantizing the discrete cosine transform (DCT) coefficients during the encoding process. In a study by Søgaard et al. [78], a mapping function between QP and Qs was estimated using the Cauchy distribution, $\text{Qs} = 0.6249 \exp(0.1156(\text{QP}))$. This mapping function allows for the determination of appropriate Qs values based on the selected QP, enabling effective control of the quantization process and optimizing the trade-off between bit rate and quality in the encoded video.

2.2 Image Distortion Prediction

The human visual system (HVS) plays a crucial role in perceiving visual content, including videos. When watching videos on home TVs, various distortions can occur throughout the transmission and display process, such as compression distortion, transmission artifacts, decoding artifacts, and display imperfections. Therefore, it is essential to measure the quality of videos to provide feedback on visual perception at the destination. Quality assessment helps predict the perceived quality of videos before they are deployed, enabling optimization of encoding parameters for improved visual experience.

In the context described above, video quality measurement can be classified into two main groups: video quality assessment at the receiver and prediction of processed video quality before applying any specific procedure. Video quality assessment can be further categorized into three types based on the amount of available information: full reference (FR), reduced reference (RR), and no reference (NR) quality assessment.

Full reference (FR) image quality assessment requires both the distorted images and the corresponding original images for comparison. Reduced reference (RR) image quality assessment utilizes the distorted image along with selected features from the original image for quality evaluation. On the other hand, no reference (NR) image quality assessment solely relies on the distorted image itself to assess its quality without any reference information. By employing these different types of image quality assessment methods, researchers can effectively measure and evaluate the quality of videos, taking into account the specific requirements and constraints of the application scenario.

2.2.1 Full-Reference IQA

The field of full reference image quality assessment (FR IQA) encompasses three main types of methods: bottom-up, top-down, and machine learning approaches. Bottom-up methods aim to mimic the different layers of the human visual system (HVS) in order to analyze and score perceived images. These methods simulate the processing stages of the HVS and examine various image characteristics to assess quality. On the other hand, top-down techniques are more commonly used and practical. They treat the HVS as a black box and extract statistical information from the distorted image to determine the type and level of impairments. By analyzing the statistical properties of the distorted image, these methods can estimate the extent of distortion without explicitly modeling the underlying mechanisms of the HVS. In recent years, machine learning methods have experienced significant advancements in FR IQA. These approaches leverage the power of machine learning networks, which can automatically extract relevant features from images without relying on handcrafted engineering methods. The layers of these networks establish connections similar to the hierarchical structure found in the HVS, enabling them to learn complex relationships and patterns. By employing classifiers or regressors, machine learning methods can infer the quality score of an image based on its extracted features. The emergence of machine learning techniques in FR IQA has led to significant progress in accurately predicting image quality, driven by their ability to leverage large datasets and learn intricate representations from the data. These methods have demonstrated promising results and opened new possibilities for advancing the field of image quality assessment.

The simplest top-down method for measuring image and encoded video quality is by calculating the mean square error (MSE) between the distorted and original images. MSE is commonly used in encoders due to its computational efficiency. However, it has been observed that MSE does not always correlate well with perceived video quality [24]. This observation has driven the development of various methods that aim to capture the correlation between image artifacts and the characteristics of the human visual system (HVS). One of the most well-known approaches in this regard is the Structural Similarity Index (SSIM) [90]. SSIM takes into account the distortion of luminance and considers similarity between images. Many other quality metrics have been derived from SSIM, including MS-SSIM [89], FSIM [97], and SR-SIM [96]. While these metrics exhibit a meaningful correlation with perceived visual quality, they do not perfectly align with Mean Opinion Scores (MOS) that reflect human judgments. Furthermore, experimental results [24] have shown that video quality assessment approaches may not provide consistent quality scores for videos with similar MOS. To address these limitations, the authors in [24] introduced a quality assessment approach called Video Multi-method Assessment Fusion (VMAF). VMAF combines three quality metrics: Visual Information Fidelity (VIF) [76], Detail Loss Metric (DLM) [51], and motion. The output of these metrics is fused using a Support Vector Machine (SVM) [70]. Experimental results have demonstrated a strong correlation between VMAF and MOS, as well as consistent behavior across various Netflix videos. VMAF has emerged as a promising alternative to MOS for predicting video quality based on the HVS. Considering the effectiveness and robustness of VMAF, we have chosen it as our preferred metric for predicting video quality based on the characteristics of the human visual system.

2.2.2 No-Reference IQA

The NR IQA is the most practical distortion assessment among the other IQA types since the original image information usually is not accessible at the destination. The NR IQA methods typically model the natural images and find a distorted deviation from the natural image model. DIVINE method, presented in [63], classifies the type of image distortion and employs a specific regressor to score distortion with every artifact. BLINDS [73], NIQE [60] and BRISQUE [61] employ Gaussian distribution to model spatial features of images and measure distortion. FRIQUE proposed in [25] employs hand-crafted features as input of a deep belief network and a support vector machine as the regressor to predict the quality. CORNIA [94] method employ a constructed codebook developed by performing k-means clustering of natural images' patches illumination and contrast. Then SVR takes the distance between codewords and distorted patches as input to regress and concludes the perceived quality. SOM employs the CORNIA method but detects objects and takes object patches as its input.

Since the CNN methods solved machine vision recognition problems with high accuracy, they found their way to IQA as a reliable solution. Reference [41] proposed a shallow CNN network with one convolutional layer and two fully connected layers to regress and predict normalized input patch quality. Reference [41] used averaging as its pooling method to find image quality. Reference [44] employed a two-layer CNN network and simulated FR IQA scores of normalized patches to augment database size. It regressed the extracted features of patches with one layer perceptron to predict image-wise IQA. References [10], [11], [12],[45], and [81] used deeper CNN networks for IQA proposes.

2.3 Rate Prediction

Optimizing the performance of an encoder requires careful consideration of the trade-offs between distortion and bit rate. Reducing the bit rate generally leads to an increase in distortion, so it is important to establish models that characterize the relationship between bit rate and distortion for effective rate control. Several rate-distortion models have been developed to capture this relationship. These models include the quadratic model [18], linear model [21], exponential model [50], ρ model [87], and lambda model which is a general formulation that can adapt to various rate-distortion characteristics. Each of these models offers a different mathematical formulation to describe the rate-distortion trade-off. These models provide valuable insights and guidelines for rate control algorithms, allowing encoders to make informed decisions and optimize the compression process based on specific requirements and constraints.

Rate control techniques in video coding often rely on rate-quantization models that establish a relationship between the encoding rate and the quantization parameter. These models, such as those presented in [18][47], take into account the complexity of frames or blocks. Chiang et al. [18] proposed a quadratic rate-quantization model for H.264 rate control, utilizing the mean absolute difference (MAD) of the residual signal to estimate the quantization step. The quadratic model introduced in [18] has served as a basis for the development of several other models, as evidenced by works such as [54], [53], [55], and [17], which were inspired by and derived from it. In the pursuit of more efficient rate control, Liu et al. [53] proposed a simplified linear rate-quantization model as a verification of the quadratic model. This linear model exhibited a high-speed response to changes in frame complexity and demonstrated

accurate rate estimation. The quadratic rate-quantization model, based on MAD estimation mentioned in [19], was initially adopted in early versions of the HEVC encoder. Nevertheless, this model had limited accuracy and applicability in certain scenarios.

In the realm of rate control, various approaches have been proposed to estimate the encoding rate in video coding. He et al. and Milani [29] et al. [58] introduced the ρ-model, which utilizes the number of zero coefficients to estimate the rate. Similarly, Wang et al. [87] employed the ρ-domain model for HEVC rate control by estimating the number of zero DCT coefficients. While these models provide accurate rate estimation, they do not directly determine the quantization parameter. Dong et al. [21] presented a context-based adaptive linear rate control model that utilizes the estimated mean absolute difference (MAD). In contrast, Kwon et al. [47] used the sum of absolute transformed difference (SATD) as a measure of frame complexity in their rate control model. Content-dependent rate control methods were proposed by Karczewicz et al. [42] and Wang et al. [82]. The adaptive rate control model in [42] measures the intra-frame complexity using SATD as an index, while [82] incorporates the gradient index to measure frame complexity in its rate control model. Although these methods aim to enhance HEVC rate control, they operate at the GOP or frame level. Overall, these studies have contributed to advancing rate control techniques in video coding, but further improvements are still sought, especially in achieving more fine-grained control at the block level.

Jiang et al. [37, 38] introduced their rate control model, which utilizes the peak signal-to-noise ratio (PSNR) as the index of context complexity. Jing et al. [39] employed a gradient-based approach for frame complexity measurement and proposed

content-dependent rate control. Zhou et al. [102] utilized a histogram of differences (HOD) as a frame complexity index. Kamaci et al. [40] and Sanz et al. [74] proposed residual signal estimation methods using Cauchy probability density function (PDF) at the frame and block levels, respectively. The R-λ model, investigated in [50], has demonstrated high performance and low complexity for HEVC rate control. In fact, the R-λ model was adopted in the HEVC reference software version 10.0 to provide more accurate rate control. Additionally, an adaptive intra-frame rate control method was proposed in [50] to enhance the R-λ model. Lee et al. [48] employed the Laplacian probability distribution function (PDF) to model the residual signal, and they investigated textural and non-textural bits to develop their rate quantization model.

Li et al. [52] and Wang et al. [83] introduced a CTU-level rate control scheme. Wang et al. utilized a Lagrangian multiplier in their approach, but the overall performance was found to be unsatisfactory. To address this, Zhou et al. [100] presented a context complexity-based rate control model at the CTU level. They optimized their model based on the measured mean squared error (MSE), which exhibits a low correlation with perceptual quality. In coding optimization, peak signal-to-noise ratio (PSNR) and MSE are commonly used metrics; however, they do not necessarily align with human-perceived quality. To better assess visual quality as perceived by humans, the structural similarity index (SSIM) is a more suitable metric as it compares image structures to evaluate image distortion. Wang and Lit et al. in [85, 86, 88] developed an SSIM-based quantization and rate-distortion optimization technique with a two-pass encoding procedure. Aswathappa et al. [7] presented an inter-frame

SSIM-based rate control method. Furthermore, Zhou et al. [101] proposed a CTU-level SSIM-based rate control model specifically designed for HEVC. These studies have contributed to the advancement of HEVC rate control by exploring alternative metrics such as SSIM that better capture perceptual quality, ultimately leading to improved video coding performance.

2.3.1 Adaptive Spatio-Temporal Encoding

Previous research has extensively investigated the benefits of downsampling videos or images before encoding and upsampling them after decoding, which has been shown to improve encoding rate-distortion performance [6, 15, 22, 31, 32, 36, 66, 92]. Bruckstein et al. [15] examined the resampling of compressed images with JPEG and demonstrated that at low bit rates, resampling enhances JPEG encoding performance. Wu et al. [92] studied oversampled images and highlighted the negative impact of encoding such images at low bit rates. In the context of H.264/AVC encoding, Dong et al. [22] analyzed encoding and downsampling distortion separately. They aimed to minimize overall distortion by determining the optimal scaling ratio. Nguyen et al. [66] investigated the quantization parameter at a scaled resolution based on image content, exploring the potential benefits of adaptive quantization. Hosking et al. [31] focused on the resampling of intra-frames before encoding with HEVC, assuming that inter-frames already underwent efficient compression. They demonstrated that at low bit rates, resampling intra-frames reduces distortion. Furthermore, Hosking et al. [32] extended their adaptive encoder by introducing an adaptive algorithm that modifies coefficients based on the bit rates of previously encoded frames. Afonso et al.

[6] presented a linear content-based model for the adaptive spatiotemporal HEVC encoder, further enhancing the adaptiveness of the encoding process. Collectively, these studies highlight the potential benefits of downsampling and upsampling in improving encoding rate-distortion performance, particularly at low bit rates, and provide insights into adaptive encoding techniques for different video coding standards.

A low frame rate in videos can lead to various visual artifacts such as aliasing, blurring, and flickering. Increasing the frame rate can help reduce these artifacts and improve the overall visual quality. However, limited bandwidth at low bit rates can introduce additional artifacts and distortion. To mitigate these issues, it is important to consider the content of a video sequence and adjust the frame rate accordingly, aiming to minimize encoding distortion. One approach to achieve content-dependent temporal adaptation is by using a temporal index that predicts the perceived quality of frames at different frame rates. This enables the development of an encoder that dynamically adjusts the frame rate based on the content of the video. Researchers, such as Bull et al. [16] and Series et al. [75], have investigated the effects of increasing video parameter dimensions, such as wider color range, higher resolution, and frame rate. The impact of spatiotemporal features of videos on the human visual system has also been studied by Daly et al. [20], Noland et al. [67], and Mackin et al. [56]. Zhang et al. [95] proposed a quality assessment model based on temporal content to predict the perceived quality of videos at various frame rates, utilizing temporal wavelet transform in their assessment model. Several studies, including Sugawara et al. [79], Emoto et al. [23], Ou et al. [68, 69], Nasiri et al. [65], and Zhang et al. [95], have investigated the relationship between frame rate and perceived quality, shedding light on the perceptual impact of different frame rates on video viewing experience.

Chapter 3

ML based Spatially Adaptive Video Compression

With the growing interest in immersive visual experiences, the limitation of bandwidth becomes a crucial consideration. However, encoding videos at low bit rates often results in compression artifacts such as blockiness, and flickering. Prior research has demonstrated that applying spatial adaptive encoding (SAE), which involves downscaling the video before encoding and upscaling after decoding, can significantly enhance compression quality. By reducing spatial redundancy through downscaling, frame quality can be improved while maintaining the same bit rate. Prior studies such as [5] and [6] have explored the benefits of downscaling before encoding to improve rate-distortion performance. In these studies, a linear model was proposed to predict the quantization parameter (QP) for encoding downscaled videos.

The effectiveness of spatial scaling before encoding is heavily dependent on the content of the frames. Figure 3.1 illustrates two rate-distortion curves for a sample video at full resolution and downscaled resolution. As depicted in Figure 3.1, the

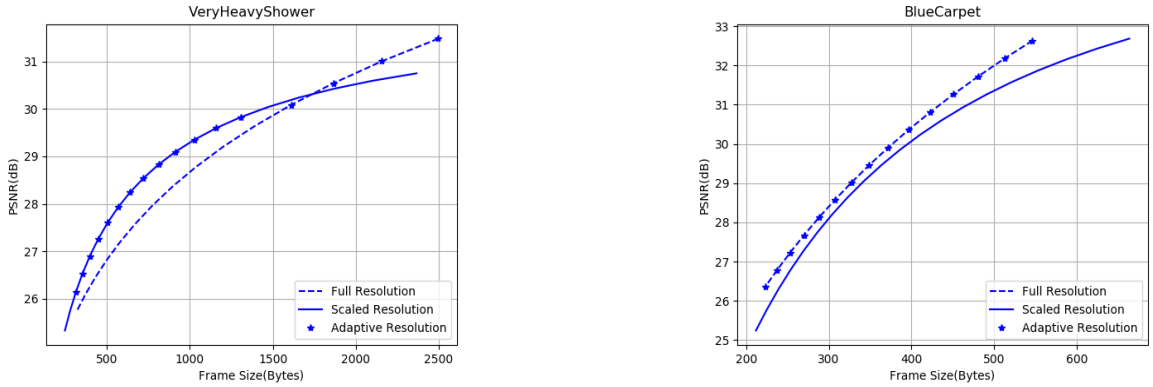

Figure 3.1: Performance comparison of a two samples rate-distortion curves at two resolutions.

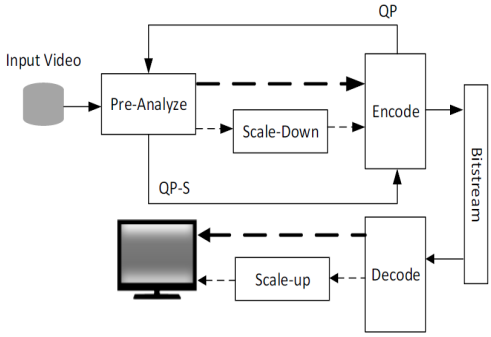

Figure 3.2: The top level pipeline of pre-encoding SAE module.

downscaled video exhibits higher quality at the same bit rate compared to the full resolution video, especially at low bit rates.

Figure 3.2 provides an overview of the proposed SAE method, showcasing a use-case scenario. The diagram presents the key components of the spatially adaptive encoder, including a pre-analyzer to find optimized resoluiton and QP with which to encode video, a downscaling process before encoding, and an upscaling process after decoding.

In the subsequent sections, two spatial adaptive encoding methods are introduced

for encoding I-frames and P-frames. Extensive research has been conducted on frame features to devise optimized and low-complexity SAE techniques. Various commonly employed frame and sequence features have been leveraged for machine learning-based training of the SAE approach, specifically targeting I-frames and P-frames.

3.1 Proposed Approach for I-frame Spatially Adaptive Compression

Traditionally, achieving optimal video quality necessitates an exhaustive search for quantization parameters and resolutions. However, our proposed method offers a more efficient approach by predicting video quality across different resolutions at the current bit rate, based on the complexity of I-frames. To accomplish this, we train three two-layer feed-forward neural networks (NNs). Each NN's hidden layer consists of 40 nodes, while the output layer contains a single node. The sigmoid function serves as the activation function.

Through experimentation, we found that employing 40 nodes in each NN yielded accurate predictions. The first NN is responsible for predicting the PSNR of the full-resolution video, while the second NN predicts the PSNR of the downscaled video. The third NN is trained to predict the quantization parameter (QP) for the downscaled video.

Fourteen video game sequences were employed to train and evaluate the performance of the NN models. To facilitate the training process, the sequences were divided into two subsets: 8 sequences for training and 6 sequences for testing. Within the training set, 80% of the frames were allocated as training data, while the remaining

frames were utilized as validation data.

3.1.1 Feature Extraction

Experimental findings indicate a significant correlation between I-frame complexity and the resulting bit rate, given a fixed quantization parameter. To quantify the complexity of an I-frame, we employed gradient. Previous experiments conducted in [46] have demonstrated that the average gradient of a frame serves as a reliable indicator of its complexity and, consequently, its bit rate. The gradient is defined in Equation 3.1:

$$Grad = \sum_{i=0}^{L} \sum_{j=0}^{W} \frac{|Y(i,j) - Y(i,j+1)| + |Y(i,j) - Y(i+1,j)|}{W \times L} \tag{3.1}$$

where Grad is the gradient of the frame; $Y(x, y)$ are the luminance values of pixel (x, y) in the related frame respectively; while $W \times L$ is the number of luminance pixels per frame.

3.1.2 NN Training Procedure

The training process involves training all three NN models using the designated training set. These models take the gradient and quantization parameter (QP) of I-frames as inputs and predict the peak signal-to-noise ratio (PSNR) at the native resolution, as well as the PSNR and QP at the scaled resolution.

To train the NN model for the native resolution, the I-frame's gradient and a set of QPs are provided as inputs. The NN model then outputs the PSNR of the encoded I-frames at those corresponding QPs.

For training the scaled resolution NN model, both the native and scaled I-frames from the training dataset are encoded using different QP values, and their respective sizes are recorded. For each reconstructed I-frame's size at the scaled resolution, a matching QP at the native resolution, which yields the same bit rate, is identified and stored. These stored QPs serve as the equivalent QPs at the native resolution.

During the training of the scaled resolution NN model, the inputs consist of the native I-frame's gradient and the QP at the native resolution, while the outputs are the PSNR and QP at the scaled resolution. This training setup enables the model to learn the relationship between the gradient, QP, and the resulting PSNR and QP at the scaled resolution.

3.1.3 NN Testing Procedure

Figure 3.3 illustrates the three trained NN models. For all models, the inputs consist of the I-frames' gradient and the corresponding QP values. The full resolution NN model is designed to predict the PSNR at the native resolution, while the scaled-resolution NN models are responsible for predicting the PSNR and QP at the scaled resolution.

To determine the optimal resolution, the predicted scaled resolution PSNR is compared with the native resolution PSNR. The resolution that yields the highest PSNR is adaptively selected. This approach allows for adaptive resolution selection based on the comparison of predicted PSNR values between the scaled resolution and the native resolution.

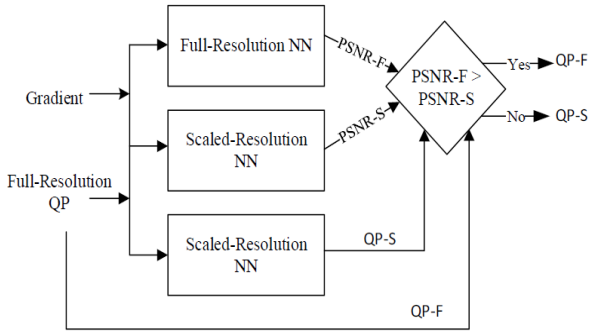

Figure 3.3: Overview of proposed adaptive compression method.

3.2 ML Based P-frame Spatially Adaptive Compression

Empirical findings indicate that video compression performance is heavily influenced by the content of the frames. Frames with significant texture require a higher bit rate for encoding compared to frames with more uniform areas. Besides the spatial characteristics that impact compression efficiency, fast motion within a video sequence can lead to higher bit rates for the encoded frames.

In this section, we investigate various spatial and temporal features of video datasets. These features are utilized to identify the QP of intersection (QP that native and scaled resolution R-D curve intersection) and equivalent QP (the QP to encode scaled resolution to address the assigned bit rate) values for P-frames and group of pictures (GOPs). To optimize the proposed spatially adaptive encoding (SAE) method, we train and test a K-nearest neighbors (KNN) network using different subsets of spatial and temporal features. The accuracy of QP prediction at the intersection is compared using these feature subsets, and the subset that yields the best prediction performance is selected. The P-frame SAE (PF-SAE) module predicts

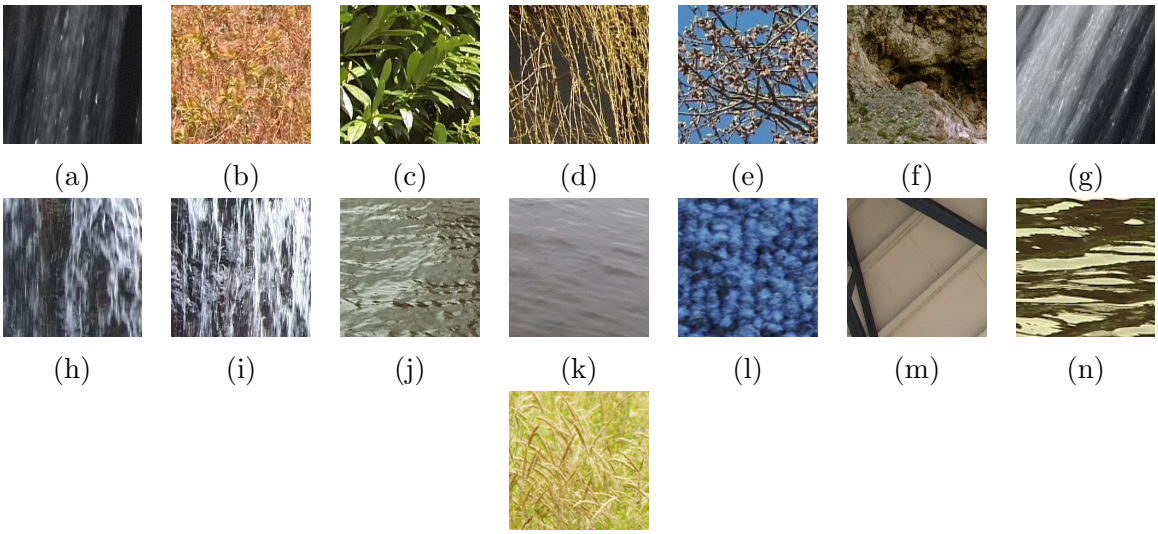

Figure 3.4: Frame samples from [43] dataset, denoted as (a)heavyshower, (b)movingfeild,(c)sunnybush, (d)thinbranches, (e)treeflower, (f)treetrunk, (g)veryheavyshower, (h)waterfall, (i)waterfall-homo2, (j)wavyshinnysea, (k)CalmSea, (l)bluecarpet(m)ceiling, (n)flowing-river, (o)grassfield

the optimal resolution for encoding, the QP value at the intersection of rate-distortion curves, and the equivalent QP for the scaled encoded videos at different resolutions.

3.2.1 Rate-distortion performance comparison

To investigate the impact of spatial and temporal features on adaptive resolution encoding, we utilized a dataset [4, 43] consisting of various videos. This dataset encompasses continuous dynamic videos, characterized by deformed surfaces like water or a flag; discrete dynamic videos, showcasing objects such as leaves in motion; and static videos featuring global movement. Figure 3.4 displays example frames extracted from a selection of videos within the dataset.

The expansion of video parameters, such as resolution, color dynamic range, and

frame rate, necessitates an increase in bandwidth or storage capacity. To mitigate the need for higher bandwidth, there are two potential approaches: employing more complex compression techniques and using coarser quantization steps, or downsizing the video resolution before encoding to reduce the bit rate.

However, it is important to note that both encoding and downscaling can introduce distortion and degrade video quality. The optimal trade-off between using a coarse quantization step during encoding and downsizing the video is highly dependent on the spatial and temporal features present in the video sequences. These features play a crucial role in determining the most suitable approach for achieving a balance between compression efficiency and preserving video quality.

Figure 3.5 compares the rate-distortion curves of selected sequences from the dataset at three different resolutions: full resolution, 85% of the original resolution, and 75% of the original resolution.

As depicted in Figure 3.5, downsizing the videos scaled to 85% of native resolution enhances the rate-distortion performance for certain videos, particularly at lower bit rates. However, downsizing the videos further to 75% of native resolution does not yield any noticeable improvement in the rate-distortion curve performance.

The filters are utilized during the downscaling and upscaling processes can effectively reduce aliasing effects and blurriness. We explored and compared three different filters: Nearest neighbor, Lanczos, and Bicubic.

The Nearest neighbor filter may result in stairway artifacts, while the Bicubic filter can introduce blurriness in the upscaled images. In contrast, the Lanczos filter exhibits superior performance compared to the other two filters. Figure 3.6 visually

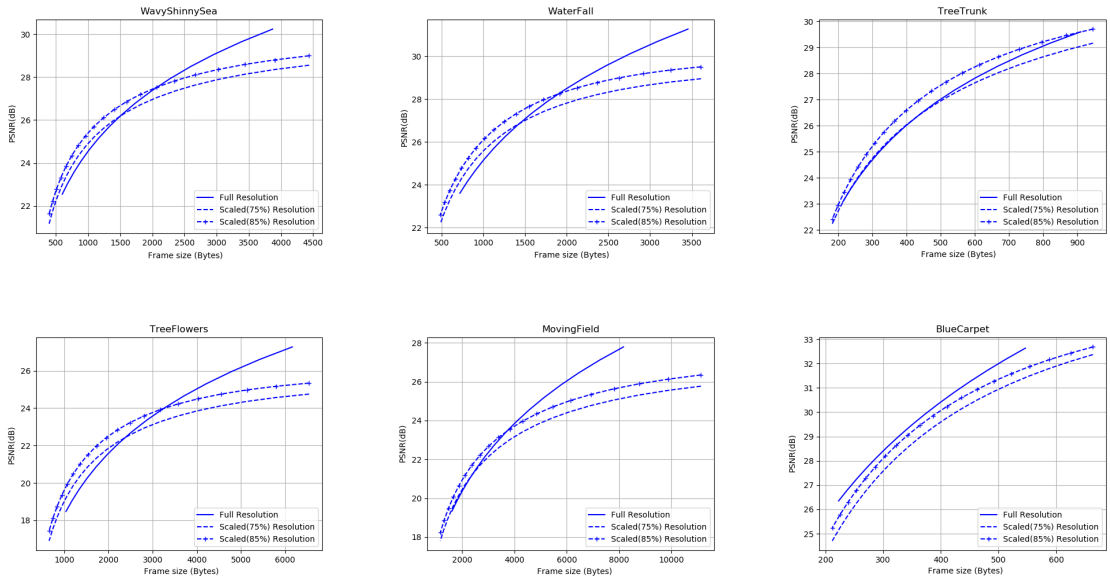

Figure 3.5: R-D performance of 6 tested sequences for P-frames encoded at three resolution. Each data point represents the average value of all frames for a given QP.

demonstrates the better performance of the Lanczos filter. Consequently, the Lanczos filter is employed for the training and testing phases of the proposed P-frame SAE (PF-SAE) method. By leveraging the Lanczos filter, we aim to enhance the rate-distortion performance while minimizing the negative visual artifacts caused by downscaling and upscaling operations.

3.2.2 Frame Features

In this section, we present a diverse range of spatial and temporal features that play a crucial role in the development of an efficient P-frame SAE (PF-SAE) method. These features encompass both spatial and temporal aspects, enabling the measurement of frame textural characteristics as well as the extraction of motion and dynamics between frames. By combining these spatial and temporal features, we create a

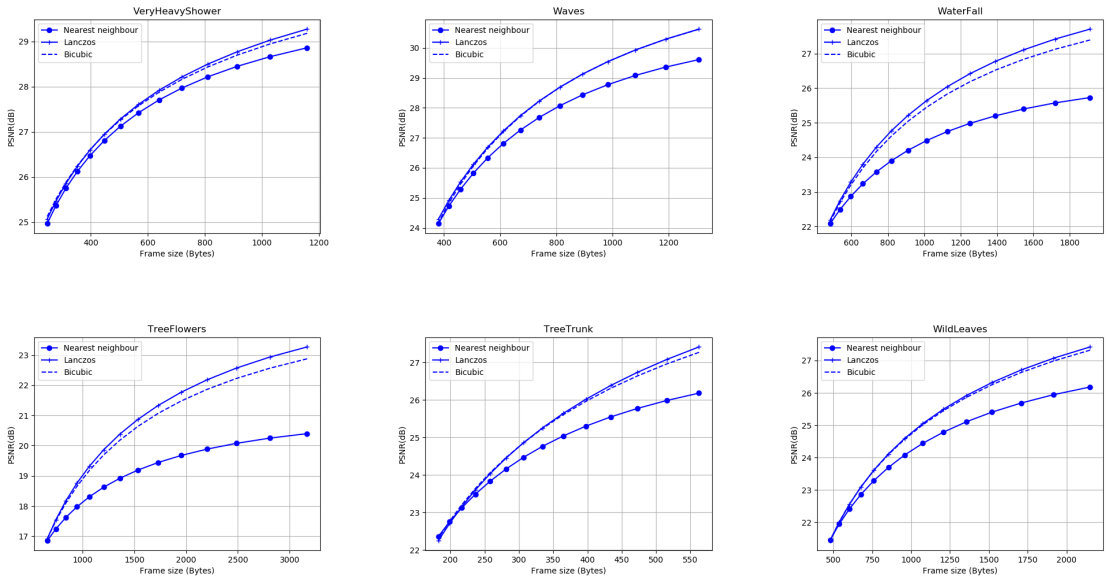

Figure 3.6: R-D performance of 6 downscaled tested sequences for P-frames upscaled with three filters: Bicubic, Nearest neighbour, Lanczos. Each data point represents the average value of all frames for a given QP.

comprehensive feature set that serves as the foundation for training and testing the ML-based PF-SAE model. The integration of these features empowers the PF-SAE to effectively adapt to the varying characteristics of different video sequences and optimize the encoding process accordingly.

Sobel Filter

The experimental results reveal a strong correlation between image encoding properties, such as quality and bit rate, and the texture characteristics of the image, specifically flatness and coarseness. The coarseness level of an image can be quantified by measuring its edge energy. To extract the horizontal and vertical edges of a grayscale image, the Sobel kernels are applied [26]. The resulting frames capture

either the horizontal edges or the vertical edges present in the image. Let s_v and s_h represent the pixels of the vertical and horizontal edge images, respectively. The pixel magnitude of the edges' image, denoted as $s_t = s_h + s_v$, represents the overall edge intensity. This edge intensity serves as a measure of the image's coarseness and is defined as follows:

$$Sob = \sum_{i=1}^{L} \sum_{j=1}^{W} \frac{s_{t(ij)}}{L \times W} \tag{3.2}$$

where L and W represent the length and width of the image, respectively. The aggregation of the edges' magnitude is divided by the product of the image's resolution. This normalization step, as shown in Equation 3.2, ensures that the Sobel indicator is independent of the frame resolution.

Gray Scale Co-Occurrence Matrix

The grayscale co-occurrence matrix (GLCM)[28] is a matrix that captures the co-occurrence of pixel intensities at a specified offset and direction within an image [28]. It serves as a powerful tool for texture analysis, providing insights into image directionality and texture coarseness [9] [43]. The GLCM matrix has dimensions of $N \times N$, where N corresponds to the intensity levels present in the image. The matrix element p_{ij} represents the normalized count of co-occurrences of intensities i and j within the chosen offset neighborhood and direction.

Several commonly used indicators can be derived from the GLCM matrix, including contrast, correlation, energy, homogeneity, entropy, and dissimilarity. These indicators offer valuable information about different aspects of the image's texture. Equations 3.3, 3.4, 3.5, 3.6, 3.7, and 3.8 demonstrate the formulas for calculating

these GLCM descriptors. By leveraging these descriptors, we can effectively analyze and quantify the texture properties of an image.

$$GLCM_{contrast} = \sum_{i=1}^{N}\sum_{j=1}^{N}(i-j)^2 p_{ij} \tag{3.3}$$

$$GLCM_{correlation} = \sum_{i=1}^{N}\sum_{j=1}^{N}\frac{(i-m_r)(j-m_c)p_{ij}}{\sigma_r \sigma_c} \tag{3.4}$$

$$GLCM_{energy} = \sum_{i=1}^{N}\sum_{j=1}^{N} p_{ij}^2 \tag{3.5}$$

$$GLCM_{homogeneity} = \frac{p_{ij}}{1+|i-j|} \tag{3.6}$$

$$GLCM_{entropy} = \sum_{i=1}^{N}\sum_{j=1}^{N} p_{ij} log_2 p_{ij} \tag{3.7}$$

$$GLCM_{dissimilarity} = \sum_{i=1}^{N}\sum_{j=1}^{N} p_{ij}|i-j| \tag{3.8}$$

where i and j represent pixel intensities, m_r and m_c denote the mean values of the GLCM matrix in the row and column directions, and σ_r and σ_c correspond to the standard deviation values of the GLCM matrix in the row and column directions,

respectively. These parameters are used in the calculations of various GLCM indicators.

Once the GLCM indicators are computed for each frame, the mean value indicator for the entire sequence is determined. This mean value serves as a representative measure of the GLCM properties across the sequence, capturing the overall texture characteristics present.

3.2.3 Temporal Characteristic Measures

Accurately capturing movement within a sequence of frames is crucial, as the displacement between consecutive frames significantly impacts encoding performance, bit rate, and the quality of encoded P-frames. In order to predict encoding properties such as QP, bit rate, and quality, it is important to assess the temporal characteristics of a Group of Pictures (GOP). Consequently, several widely recognized temporal indicators are presented in the subsequent sections to facilitate this analysis.

Normalized Cross Correlation

Cross-correlation is a valuable metric for assessing the similarity or displacement between two matrices or vectors [49]. In this section, we employ cross-correlation to capture the temporal features of a frame sequence. Normalized cross-correlation (NCC) is a variant of cross-correlation that is bounded between -1 and 1. Equation 3.9 represents the calculation of NCC:

$$NCC = \frac{\sum_{i=1}^{W}\sum_{j=1}^{L}(I_t(i,j) - \mu_{I_t})(I_{t+1}(i-u, j-v) - \mu_{I_{t+1}})}{\sigma_{I_t} \times \sigma_{I_{t+1}}} \tag{3.9}$$

where I_t and I_{t+1} are successive frames, μ is their mean and σ is their standard deviation value. In Equation 3.9 the sliding window is defined by u and v and NCC is calculated at the frame level. Mean, skewness, deviation, kurtosis, and entropy of NCC are calculated for a sequence.

Motion Vector And Dense Motion Estimation

In frame sequences, a sparse motion vector is often employed as a temporal feature [91]. However, dense motion estimation, which considers motion vectors for all pixels rather than a subset, can offer higher accuracy [71]. We utilized dense motion estimation to extract motion vectors and dense motion estimation maps. These are obtained by comparing the P-frame with the I-frame within each GOP, as P-frames are constructed based on their differences from the I-frame.

Downsclaing Quantization And QP Prediction

To predict the intersection of rate-distortion curves for full and scaled resolutions (as depicted in Figure 3.5), determine the QP at the intersection point, and identify the equivalent set of QP_S values required to encode the scaled resolution while maintaining the same bit rate as the encoded frames at a set of QP_F values for full resolution, three KNN networks shown in Figure 3.7 were trained.

The PF-SAE module utilizes spatial and temporal features extracted from frames, which are to be encoded as P-frames. It determines, at each encoding QP for full resolution (QP_F), whether encoding at full resolution or downscaled resolution would yield higher quality while maintaining the same bit rate. For a given set of QP_F at full resolution, the PF-SAE must predict a corresponding set of equivalent QPs at

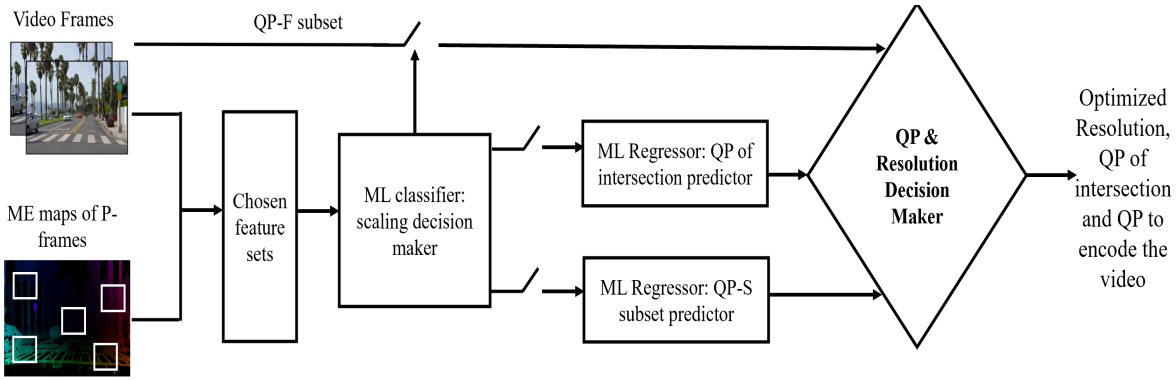

Figure 3.7: Overview of proposed P-frame adaptive compression method.

downscaled resolution (QP_S) to achieve the same bit rate. The proposed PF-SAE module consists of three ML models (KNN) [27] [99], trained to predict the optimized resolution, the QP value at the intersection of rate-distortion curves, and the set of equivalent QP_S values. Figure 3.7 illustrates the architecture of the PF-SAE module. It takes the features of frames as input and outputs the optimized encoding resolution, the QP at the intersection, and the set of QP_S values. To optimize the performance of the PF-SAE module, different combinations of features have been experimented with during training and evaluation, resulting in various feature sets denoted as F_1, F_2, ..., F_{17}. Each feature set includes a selection of spatial and temporal features described in Sections 3.2.2 and 3.2.3. In Table 3.1, G denotes GLCM, $skew$, and $kurt$, $frst$, $mean$, are skewness of the sequence, kurtosis of the GOP sequence, the mean value of the sequence, and feature value of the first frame, respectively.

3.3 Experiments

In this section, we present the results of the proposed ML-based adaptive scaled encodings. The method explores two aspects: I-frame low complexity adaptive scaling

Table 3.1: PLL and SROCC of predicted frames' bit rate from predicted bit rate of patches.

set/ Features	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}	F_{17}
$Sobel$																	✓
$PSNR$	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
$G_{hom_{frst}}$							✓	✓	✓	✓	✓		✓	✓			
$G_{hom_{skew}}$														✓	✓	✓	
$G_{hom_{kurt}}$															✓		
$G_{corr_{frst}}$							✓	✓	✓	✓	✓	✓		✓			
$G_{corr_{skew}}$														✓	✓	✓	
$G_{corr_{kurt}}$															✓		
$G_{Cntr_{frst}}$							✓		✓	✓	✓	✓	✓				
$G_{eng_{frst}}$							✓	✓	✓	✓		✓	✓				
$G_{ent_{frst}}$							✓	✓	✓		✓	✓	✓				
$G_{diss_{frst}}$							✓	✓		✓	✓	✓	✓				
NCC_{mean}		✓															
$MV_{dense_{mean}}$			✓														
$MV_{dense_{skew}}$					✓												
$MV_{arrow_{mean}}$				✓													
$MV_{arrow_{skew}}$						✓											

encoding and P-frame adaptive scaled encoding.

3.3.1 Dataset

Dataset of Proposed IF-SAE

For this study, the native resolution is set to 1920×1080, and the scaled resolution is chosen as 1280×720. To ensure efficient training and testing, a hardware-accelerated H.264 encoder is employed. This choice leverages the prevalence of H.264 accelerators in commercial systems. It is important to note that the proposed ideas are not limited to a specific codec or scaling scheme and can be applied to other codecs and scaling methods as well. In our experiments, Bicubic interpolation is used for video rescaling. The dataset used in this study [34] encompasses a diverse range of gaming content, including videos with intricate details and pronounced edges (such as Rocket,

Bioshock, Border, and Skyrim) as well as those with simpler and smoother graphics (such as Shantae and Hollow).

Dataset Of Proposed PF-SAE

The dataset used to develop the proposed GOP level adaptive scaling encoding scheme consists of 120 videos with a resolution of 256×256. Each video contains 250 frames. For encoding, a GOP size of 16 is chosen, and the IPPPP... picture structure is employed. The H.265 encoder is utilized for the encoding process. Each GOP is encoded at five different quantization parameters (QPs) selected from the set $28, 32, 36, 40, 44$. The scaled GOPs are resized to a resolution of 216×216 using the Lanczos filter for both downscaling and upscaling interpolation.

3.3.2 Proposed Adaptive Resolution Compression Method For I-frames Encoding

To evaluate the accuracy of the neural networks (NNs), the prediction error is calculated as the average absolute error across all the tested data points. The average absolute error of low-resolution QP prediction is given by Equation 3.10, where QP_{pred} represents the predicted low-resolution QP that matches the high-resolution bit rate, QPs denotes the experimentally-found low-resolution QP, and n is the number of I-frames selected for downscaling.

$$QP_{Error} = \sum_{i=1}^{n} \frac{|QP_s(i) - QP_{pred}(i)|}{n} \tag{3.10}$$

Table 3.2 presents the QP_{Error} values of I-frames for all six tested sequences. It

can be observed that the average absolute error remains below 1, which is relatively low considering the QP values ranging from 0 to 51. This indicates that our method accurately predicts the low-resolution QP while maintaining the bit rate unchanged.

Table 3.2: QP_{Error} for 6 tested sequences

Skyrim	Border	Hollow	Shantae	Bioshock	Rocket
0.39	0.36	0.4	0.45	0.39	0.87

Figure 3.8 illustrates the improved rate-distortion (R-D) performance of the proposed adaptive method compared to fixed native and downscaled resolution encoding for 6 tested sequences. To quantify the performance gain, the Bjontegaard delta PSNR (BD-PSNR) metric [8] is employed. Table 3.3 presents the average BD-PSNR gain achieved by the proposed adaptive resolution approach compared to the fixed resolution methods.

Table 3.3: BD-PSNR gain of the proposed method (dB)
(top) w.r.t. 1080p fixed (low bit rate data-points)
(bottom) w.r.t. 720p fixed (high bit rate data-points)

Skyrim	Border	Hollow	Shantae	Bioshock	Rocket
0.21	0.33	0.48	0.58	0.64	0.41
0.94	2.84	1.50	2.93	1.69	2.72

Proposed P-frame Resolution Decision Model

To select the optimized encoding resolution, K-nearest neighbors (KNN) classifier has been trained. The objective of the classifier is to predict whether scaling a video improves its rate-distortion curve or not. The KNN classifier takes F_1, F_2, ..., F_{17} as input features and produces a binary output (0 or 1) for each GOP, indicating whether scaling improves the rate-distortion performance (1) or not (0). The KNN

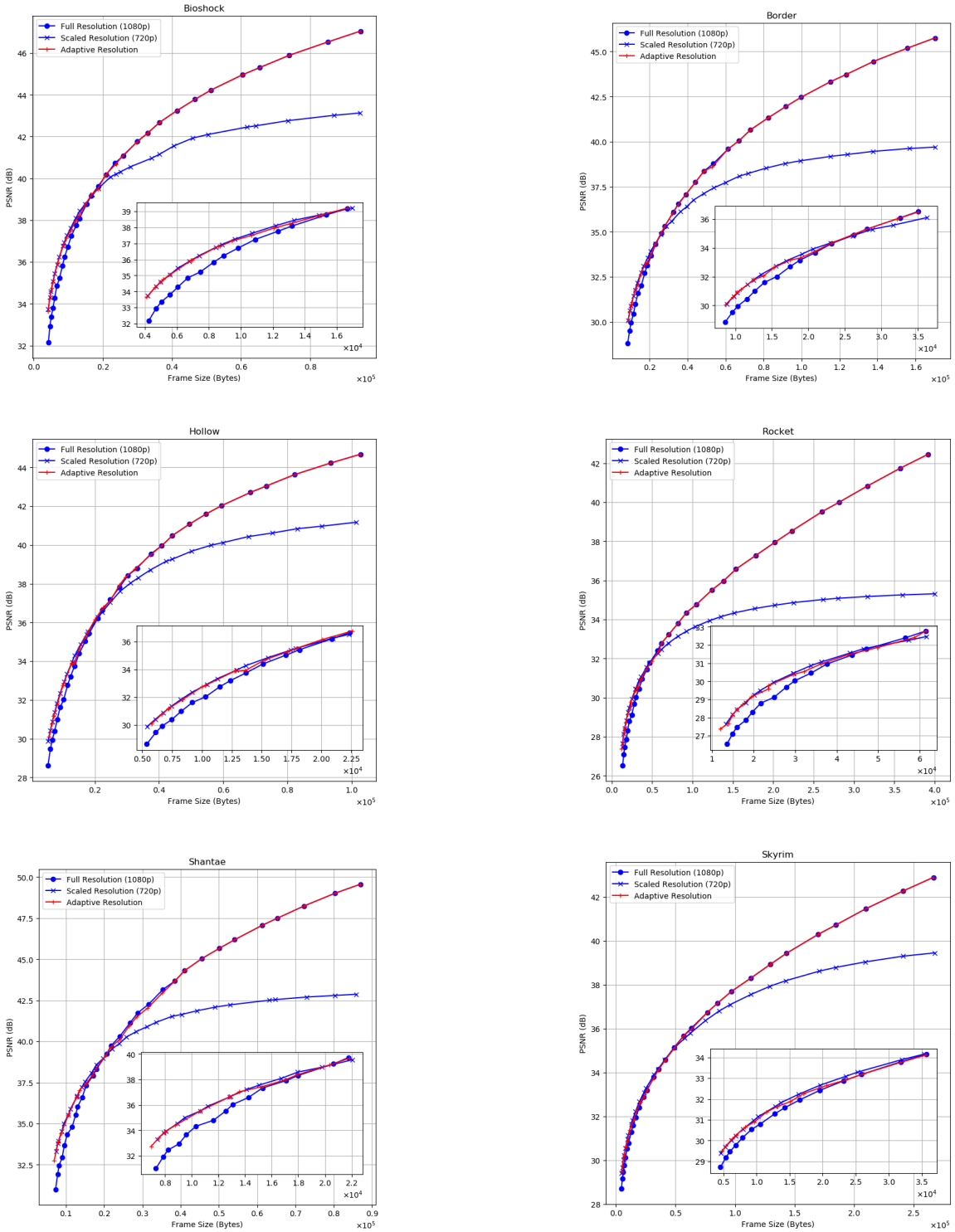

Figure 3.8: R-D performance of 6 tested sequences for I-frames. Each data point represents the average value of all frames for a given QP. The small figures expand the curves at low bit rates

classifiers were trained and then evaluated by comparing the predicted outputs with the target outputs.

Table 3.4 presents the prediction accuracy of the KNN classifiers. The accuracy is measured using the Pearson correlation coefficient, which indicates the correlation between the predicted outputs and the actual targets. The monotony of the predicted outputs is measured using the Spearman metric, and the mean absolute error (MAE) is calculated for the predicted binary outputs. According to Table 3.4, the feature set F_{14}, which includes features such as PSNR, $G_{hom_{frst}}$, $G_{corr_{frst}}$, $G_{hom_{skew}}$, and $G_{corr_{skew}}$, achieves the highest prediction accuracy for scaling decisions.

Table 3.4: PLCC and SROCC of encoding resolution quantization of test sequence.

	PLCC	SROCC	MAE
F_1	0.573	0.573	0.16
F_2	0.619	0.619	0.14
F_3	0.465	0.465	0.19
F_4	0.428	0.428	0.21
F_5	0.638	0.638	0.13
F_6	0.607	0.607	0.15
F_7	0.258	0.258	0.24
F_8	0.457	0.457	0.19
F_9	0.223	0.223	0.24
F_{10}	0.258	0.258	0.24
F_{11}	0.258	0.258	0.24
F_{12}	0.258	0.258	0.24
F_{13}	0.258	0.258	0.24
F_{14}	0.684	0.684	0.12
F_{15}	0.67	0.67	0.13
F_{16}	0.706	0.706	0.11
F_{17}	0.473	0.473	0.18

Proposed QP of Intersection And Low Resolution Model for P-frames

To predict the QP of rate-distortion curve intersection, a K-nearest neighbors (KNN) regressor, as shown in Figure 3.7, has been trained. Similar to the training and feature set selection process for the scaling decision predictor, the KNN regressor is trained

with input features F_1, F_2, ..., F_{17} and the intersection QP as the output. Table 3.5 presents the prediction accuracy of the intersection QP, measured using Pearson correlation coefficient, Spearman metric, and mean absolute error (MAE).

According to Table 3.5, the feature set F_4, which includes features such as PSNR and $MV_{arrow_{mean}}$, achieves the highest accuracy in predicting the QP of the rate-distortion curve intersection. It is worth noting that the proposed PF-SAE's intersection prediction accuracy surpasses the linear model QP prediction based on hand-crafted features, as shown in Table 3.5.

Table 3.5: PLL and SROCC of predicted intersection QP of test sequence

	PLCC	SROCC	MAE
F_1	0.804	0.722	1.44
F_2	0.81	0.736	1.38
F_3	0.759	0.702	1.62
F_4	0.818	0.756	1.42
F_5	0.774	0.675	1.58
F_6	0.775	0.715	1.56
F_7	0.718	0.638	1.66
F_8	0.734	0.653	1.68
F_9	0.718	0.638	1.66
F_{10}	0.718	0.638	1.66
F_{11}	0.718	0.638	1.66
F_{12}	0.718	0.638	1.66
F_{13}	0.718	0.638	1.66
F_{14}	0.784	0.705	1.48
F_{15}	0.798	0.746	1.47
F_{16}	0.787	0.709	1.5
F_{17}	0.775	0.709	1.49
Linear model[6]	0.708	0.642	1.71

A K-nearest neighbors (KNN) regressor has been trained to predict the equivalent QPs (QP_S) for the scaled resolution. The input features for the KNN regressor are F_1, F_2, ..., F_{17} feature sets, while the target output is the set of QP_S values obtained for each GOP, ensuring that the encoded downscaled GOP maintains the same bit rate as the encoded full resolution GOP at QPs ranging from 48 to 32. The array of equivalent QPs serves as the target output for the KNN regressor.

Tables 3.6, 3.7, and 3.8 present the accuracy of QP_S prediction, measured using Pearson correlation coefficient, Spearman metric, and mean absolute error (MAE), respectively. These tables show the prediction accuracy for equivalent QPs at QPs 36, 40, 44, and 48.

Based on the results shown in Tables 3.6, 3.7, and 3.8, the feature set F_{14} containing features such as PSNR, $G_{hom_{frst}}$, $G_{corr_{frst}}$, $G_{hom_{skew}}$, and $G_{corr_{skew}}$ achieves the highest accuracy in predicting the QP_S values. Notably, the downscaled QP_S prediction accuracy of the proposed PF-SAE outperforms the linear model QP prediction based on hand-crafted features, as indicated in Tables 3.6, 3.7, and 3.8. Figure 3.9 compares the R-D performance of our proposed adaptive resolution with full resolution and scaled resolution. As can be seen our method outperform coding at full or scaled resolution.

Table 3.6: PLCC of equivalent QP prediction of test sequence.

	36	40	44	48
F_1	0.48	0.661	0.714	0.681
F_2	0.525	0.667	0.713	0.687
F_3	0.466	0.628	0.745	0.737
F_4	0.421	0.654	0.733	0.694
F_5	0.435	0.667	0.717	0.671
F_6	0.427	0.651	0.735	0.701
F_7	0.475	0.612	0.677	0.658
F_8	0.507	0.659	0.771	0.764
F_9	0.479	0.611	0.672	0.652
F_{10}	0.475	0.612	0.677	0.658
F_{11}	0.475	0.612	0.677	0.658
F_{12}	0.475	0.612	0.677	0.658
F_{13}	0.475	0.612	0.677	0.658
F_{14}	0.596	0.723	0.804	0.791
F_{15}	0.578	0.665	0.709	0.689
F_{16}	0.597	0.716	0.79	0.775
F_{17}	0.495	0.626	0.682	0.66
Linear model[6]	0.174	0.585	0.693	0.642

Table 3.9 shows BD-PSNR gain of the proposed PF-SAE adaptive resolution compression approach with respect to fixed resolution ones.

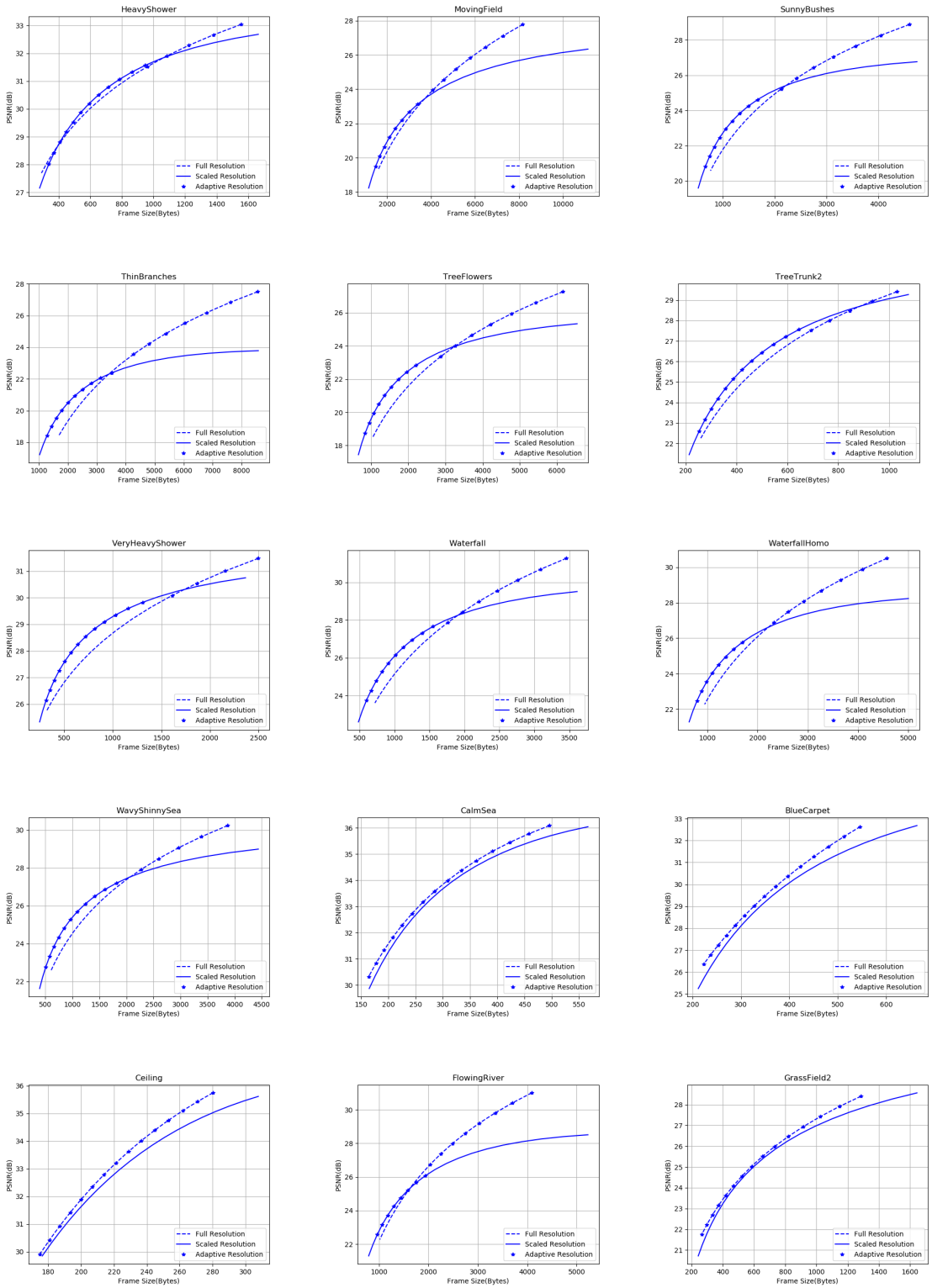

Figure 3.9: R-D performance of 15 tested sequences for P-frames. Each data point represents the average value of all frames for a given QP.

Table 3.7: SROCC of equivalent QP prediction of test sequence.

	36	40	44	48
F_1	0.493	0.682	0.777	0.849
F_2	0.533	0.681	0.782	0.86
F_3	0.4	0.622	0.794	0.852
F_4	0.422	0.652	0.795	0.85
F_5	0.433	0.645	0.774	0.842
F_6	0.445	0.666	0.782	0.859
F_7	0.426	0.48	0.644	0.695
F_8	0.458	0.628	0.774	0.837
F_9	0.427	0.478	0.644	0.695
F_{10}	0.426	0.48	0.644	0.695
F_{11}	0.426	0.48	0.644	0.695
F_{12}	0.426	0.48	0.644	0.695
F_{13}	0.426	0.48	0.644	0.695
F_{14}	0.583	0.663	0.801	0.862
F_{15}	0.547	0.645	0.785	0.861
F_{16}	0.564	0.664	0.779	0.849
F_{17}	0.521	0.591	0.691	0.734
Linear model[6]	0.251	0.572	0.748	0.832

Table 3.8: MAE of equivalent QP prediction of test sequence.

	36	40	44	48
F_1	0.97	0.69	0.81	1.04
F_2	0.94	0.69	0.8	1.02
F_3	1.02	0.77	0.81	0.96
F_4	0.99	0.68	0.74	1.0
F_5	1.01	0.72	0.8	1.04
F_6	0.99	0.72	0.76	0.93
F_7	0.98	0.77	0.91	1.17
F_8	0.97	0.7	0.73	0.91
F_9	0.98	0.78	0.91	1.18
F_{10}	0.98	0.77	0.91	1.17
F_{11}	0.98	0.77	0.91	1.17
F_{12}	0.98	0.77	0.91	1.17
F_{13}	0.98	0.77	0.91	1.17
F_{14}	0.88	0.63	0.66	0.87
F_{15}	0.92	0.68	0.73	0.95
F_{16}	0.87	0.64	0.7	0.91
F_{17}	0.92	0.7	0.83	1.13
Linear model[6]	1.32	1.26	1.64	2.2

Table 3.9: BD-PSNR gain of the proposed PF-SAE method (dB) for 15 test sequences
(top) w.r.t. full resolution (low bit rate data-points)
(bottom) w.r.t. scaled resolution (high bit rate data-points)

Heavy Shower	Moving Field	Sunny Bushes	thin Branches	Tree Flower	Tree Trunk2	Very Heavy Shower	Water Fall	water Fall -Homo	Wavy Shiny sea	Calm Sea	Blue Carpet	Ceiling	Flowing River	Grass Field
0.07	0.20	0.37	0.30	0.40	0.37	0.45	0.40	0.36	0.43	0.00	0	0	0.07	0
0.07	0.8	0.45	0.81	0.33	0.00	0.05	0.35	0.62	0.34	0.00	0	0	1.12	0

Chapter 4

Deep-learning based VMAF prediction of I-frames

The increasing prevalence of ultra HD TVs, video chat, video streaming, and surveillance cameras in our daily lives, work, and entertainment has made the use of lossy compression methods essential to meet bandwidth and storage requirements. Advanced compression techniques aim to introduce minimal distortion while encoding videos at specific bit rates. Perceptual quality assessment methods are employed to measure video quality after decoding. Additionally, predicting video perceptual quality prior to compression can aid in optimizing compression parameters to achieve the desired encoding quality and prevent flickering caused by unbalanced perceptual quality among frames in a video.

Recently, there has been growing interest in deep convolutional neural network (CNN)-based video quality assessment. In [81], M. Utke et al. assessed the quality of random patches[1] and computed their average to estimate the quality of frames.

[1] A patch is a square region of a video frame.

S. Boss et al. presented a CNN-based weighted average method for frames quality assessment in [12]. Y. Zhang et al. proposed a full-reference assessment method using deep CNN in [98]. However, none of these methods specifically address the task of predicting perceptual quality to optimize encoding parameter QP. In [93], B. Xu et al. predicted the structural similarity index (SSIM) of patches. They trained a deep CNN to predict the SSIM, but their method is limited to predicting the SSIM metric for patches of size 128×128.

This chapter introduces deep CNN-based methods for predicting the quality of encoded video frames. Given that the quality of the I-frame in a GOP with an IPP...P structure has a significant impact on the quality of subsequent P-frames, encoding the I-frame optimally plays a crucial role in determining overall video quality. Therefore, the proposed methods focus on achieving accurate quality prediction for I-frames.

Additionally, considering that each I-frame or keyframe is followed by a substantial number of P-frames, we have explored the characteristics of GOPs to develop innovative methods for predicting the perceptual quality of P-frames. The proposed methods are trained in two stages: patch-level and frame-level, which not only augment the training dataset but also enhance prediction accuracy. The patch-based approach reduces computational complexity by processing only selected patches of a frame instead of the entire image, making the method applicable to videos of various resolutions.

As a quality metric and target for the proposed methods, we employ the multi-method assessment fusion (VMAF) [24]. VMAF serves as a reliable indicator of perceptual video quality, guiding the training and evaluation of our prediction methods.

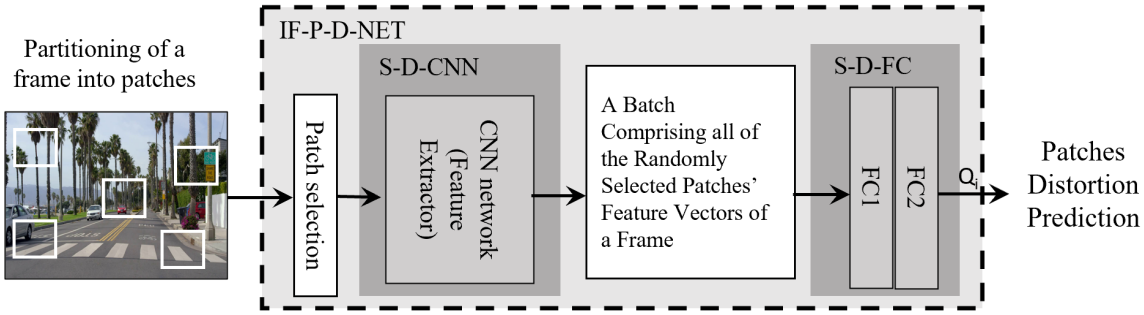

Figure 4.1: Patches' perceptual quality predictor architecture.

This chapter is organized as follows: Section 2 presents deep CNN patch perceptual quality prediction; Section 3 introduces frame-level quality prediction for I-frame. Finally, Section 4 the experimental results, and Section 5 is the conclusion.

4.1 Deep Learning CNN network architecture of patch-wise pre-encoding quality predictor

VGGNet [77] has demonstrated exceptional performance in various computer vision tasks, including classification, regression, and assessment [12]. Drawing inspiration from the success of VGGNet and previous work such as [12], we design the I-frame's patches distortion predictor network (IF-P-D-NET) shown in Figure 4.1. IF-P-D-NET adopts a similar architecture to VGGNet, consisting of CNN layers, max pool layers, and fully connected regression layers. The network architecture of IF-P-D-NET is straightforward, employing small kernel sizes of 3×3. To provide a visual representation, Figure 4.2 displays the CNN and pooling layers of IF-P-D-NET, which are denoted as S-D-CNN.

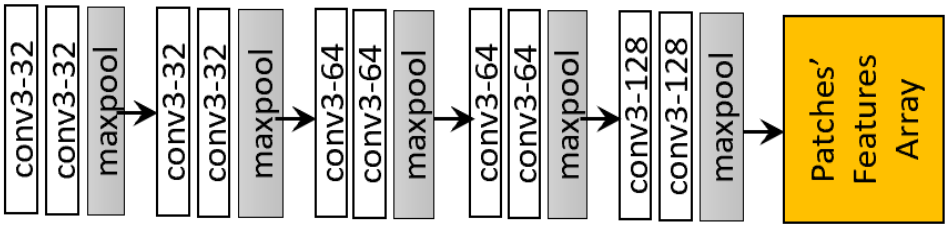

Figure 4.2: Deep CNN layers.

The S-D-CNN serves as the spatial feature extractor network for distortion prediction in our proposed method. It consists of the following layers: conv3-32, conv3-32, maxpool-2, conv3-32, conv32-32, maxpool-2, conv3-64, conv3-64, maxpool-2, conv3-64, conv3-64, maxpool-2, conv3-128, conv3-128, maxpool-2. In this notation, conv3-32 represents a convolutional layer with a 3×3 kernel size and 32 filters, while maxpool-2 denotes a max-pooling layer with a 2×2 kernel size. The regressor component comprises two fully connected layers: FC-240 and FC-5. For example, FC-240 indicates a fully connected layer with 240 output nodes. Following each convolutional layer, there is an activation layer, specifically a rectified linear layer (ReLU) [64], to introduce non-linearity into the network.

To train the IF-P-D-CNN, patches extracted from frames are used as input, while the output consists of an array representing the perceptual quality of each patch at five different QPs ($QPs \in 28, 32, 36, 40, 44$). The accuracy of frame quality prediction is influenced by two factors: the accuracy of patch quality prediction and the accuracy of transforming patch qualities into frame qualities. The size of patches can impact the transformation from patches to frame quality. Hence, we investigated frame quality prediction using three different patch sizes: 64×64, 128×128, and 256×256. Figure 4.3 illustrates the mean absolute error (MAE) of the patch-wise to frame-wise

Table 4.1: Layers of deep learning CNN network of pre-encoded I-frame's bit-rate predictor

#	Type	Kernel	Stride	Activation	Outputs
01	Conv.	3×3	1×1	ReLU	32
02					
03	Pool.	-	2×2	-	32
04	Conv.	3×3	1×1	ReLU	32
05					
06	Pool.	-	2×2	-	32
07	Conv.	3×3	1×1	ReLU	64
08					
09	Pool.	-	2×2	-	64
10	Conv.	3×3	1×1	ReLU	64
11					
12	Pool.	-	2×2	-	64
13	Conv.	3×3	1×1	ReLU	128
14					
15	Pool.	-	2×2	-	128
16					120
17	FC			ReLU	5
18					120
19	FC			ReLU	5

quality transformation for each patch size. To calculate the MAE, video samples were encoded at the five QPs, the actual VMAF values of the patches were computed and averaged to predict the frame quality. Subsequently, the predicted frame quality was compared to the actual encoded frame VMAF, and the MAE was calculated. As depicted in Figure 4.3, the transformation error is lowest for the 64×64 patch size compared to the other two patch sizes. Therefore, we utilized the 64×64 patch size for training and testing the proposed perceptual quality prediction methods.

For training IF-P-D-CNN, the patch-wise perceptual quality predictor, the cost function is stated in equation 4.1. ψ_I is the networks mapping function, P_i and ω_I in $\psi_I(P_i, \omega_I)$ are i'th patch and weights of ψ_I respectively. $\psi_I(\cdot)$ maps patch features to

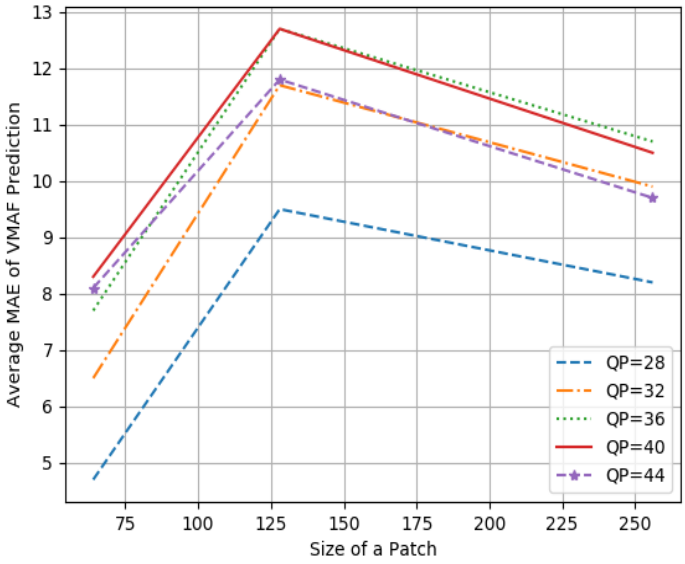

Figure 4.3: Average MAE of transforming patches' quality to a frame quality for different patch-sizes

the quality of each patch. Q_{I-Pi} is the VMAF of the i'th patch of an I-frame.

$$J = |Q_{I-Pi} - \psi_I(P_i, \omega_I)| \tag{4.1}$$

4.2 Patch-wise to Frame-wise Perceptual Quality Prediction Transformation

4.2.1 Homogeneous Average I-frame Quality Prediction Method

The Homogeneous Average I-frame's Distortion Predictor network (HA-IF-D-NET) leverages the S-D-CNN network to predict the quality of randomly selected patches

Table 4.2: MAE of Predicting frame quality (VMAF) by transforming patch-wise to frames-wise quality with different percent of randomly selected patches.

QP	28	32	36	40	44
100%	4.97	6.56	7.65	8.22	7.86
40%	4.92	6.52	7.61	8.23	7.92
30%	4.97	6.57	7.65	8.21	7.92
20%	4.98	6.57	7.68	8.25	7.98
15%	4.98	6.58	7.64	8.18	7.83
10%	4.97	6.57	7.59	8.15	7.9

and computes the average predicted quality for each I-frame. The structure of HA-IF-D-NET is depicted in Figure 4.4. In this network, I-frame patches are fed into the CNN for feature extraction, and a subset of these patches is randomly chosen. The feature vectors of these patches are then passed through a regressor, resulting in the VMAF score for each patch. To reduce computational costs in frame quality prediction, it is essential to determine the minimum number of randomly selected patches that can maintain prediction accuracy. To investigate this, the transformation error (MAE) was measured when using different percentages of frame patches to predict frame quality, and the results are presented in Table 4.2. As shown in Table 4.2, utilizing only 10% of the patches yields nearly the same frame quality prediction accuracy as using all patches.

4.2.2 Sorted Average I-frame Quality Prediction Method

The proposed Sorted Average I-frame Distortion Predictor network (SA-IF-D-NET) is based on an important observation: the average quality of patches is consistently lower than or equal to the quality of the entire frame. This observation is in line with the masking effect of the human visual system (HVS), which incorporates the HVS's

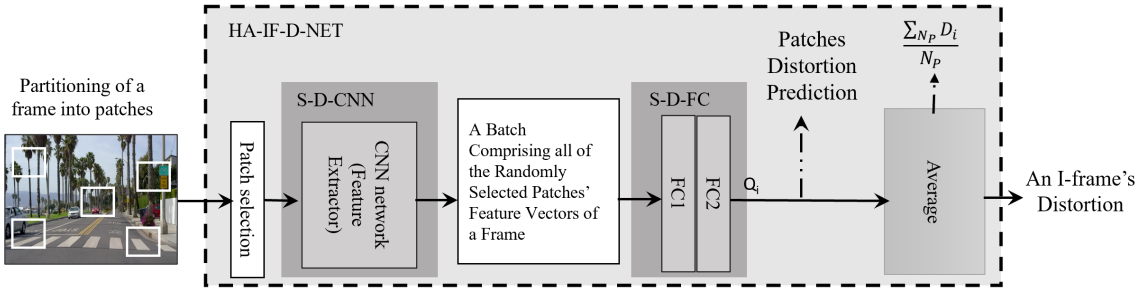

Figure 4.4: Architecture of homogeneous average method for I-frames' perceptual quality prediction.

sensitivity limitations in perceiving luminance, spatial frequency, and orientation. Consequently, the HVS perceives the overall picture and does not process an image based on the quality of individual patches. For example, the luminance masking effect reduces the sensitivity of the HVS to distortion in darker and brighter areas. The structure of SA-IF-D-NET is illustrated in Figure 4.5. As depicted in Figure 4.5, SA-IF-D-NET sorts the quality of patches within a frame and computes the average of a subset of predicted qualities with higher magnitudes. Different percentages of sorted patch qualities ($S_{ptch}\%$) are used to calculate the predicted frame quality. Table 4.5 displays the mean absolute error (MAE) of the sorted average transformation method for various $S_{ptch}\%$ values corresponding to five QPs. The table is based on the training dataset. In Table 4.5, the minimum MAE achieved for each QP is highlighted, and the corresponding $S_{ptch}\%$ value is utilized for predicting frame-wise quality at that QP.

4.2.3 Weighted Average I-frame Quality Prediction Method

The Weighted Average I-frame Distortion Predictor network (WA-IF-D-NET) is depicted in Figure 4.6. Unlike the sorted method (SA-IF-D-NET) where high-quality

Table 4.3: MAE of Predicting frame quality (VMAF) by using sorted averaging method with different percent of sorted patches' VMAF at different QPs.

QP	28	32	36	40	44
10%	0.85	1.6	3.21	7.15	14.88
20%	**0.69**	**1.06**	1.96	4.81	11.02
40%	1.26	1.55	**1.69**	**2.66**	6.29
60%	2.23	2.85	2.97	2.93	**4.31**
80%	3.36	4.47	4.98	4.83	5.23
100%	4.9	6.61	7.93	8.68	8.6

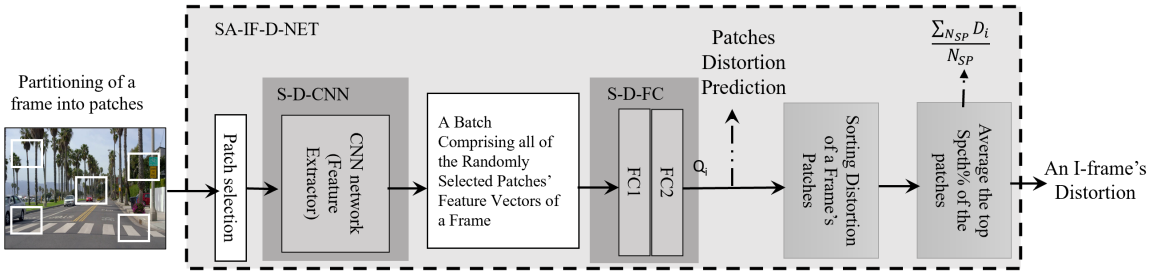

Figure 4.5: Architecture of sorted average method for I-frames' perceptual quality prediction.

patches are used for frame quality prediction, the WA-IF-D-NET method employs a fully connected network trained to predict the weight assigned to each patch in the frame quality prediction process, as illustrated in Equation 4.2. This weight assignment mechanism enables the network to dynamically adapt the contribution of each patch to the overall frame quality estimation.

$$Q_W = \frac{\sum_{i=0}^{N_P} Q_{Pi} \times W_i}{N_P} \tag{4.2}$$

Where Q_W is the frame quality predicted by the weighted average method, N_P is the number of random patches extracted from each frame, Q_{Pi} is the predicted quality of the ith patch, and W_i is the weight of the ith patch quality. For predicting W_i, two

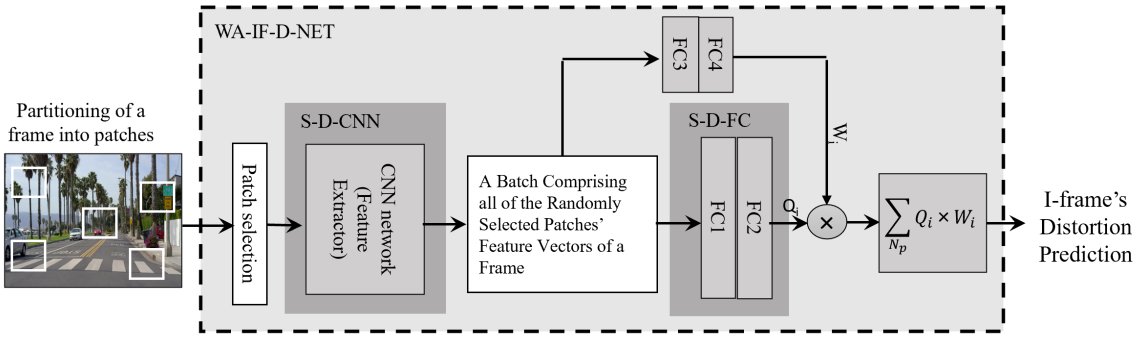

Figure 4.6: weighted average method for I-frames' perceptual quality prediction.

fully connected layers (FC-3 and FC-4) shown in figure 4.6 take CNN-based extracted spatial features as input, FC-3 and FC-4 are trained to minimize the cost function shown in equation 4.3.

$$J = |Q_{IF} - \sum_{i=0}^{N_P} \tau_I(\psi_{Feature}(P_i, \omega_I^*), \theta_I) \times \psi_I(P_i, \omega_I^*)| \tag{4.3}$$

Where τ_I maps a batch of patches' features to I-frame quality. Q_{IF} is the actual I-frame's VMAF, $\psi_{Feature}$ is part of ψ_I mapping function and maps patches to their extracted features, ω_I^* is optimized weights of ψ_I mapping function, θ is the weight of τ_I mapping function, and N_P is the number of random patches utilized for predicting I-frame perceptual quality.

4.3 Deep CNN Architecture for P-Frame Patch-wise Pre-Encoding Quality Predictor

P-frames are generated based on the residual information between an I-frame and the P-frame, and various factors such as the content complexity of the I-frame and

the temporal distance between each P-frame and the I-frame can affect the quality of decoded P-frames. To capture the motion displacement between frames, we utilize the dense motion estimation (ME) approach presented in [14] to generate a ME map between the I-frame and P-frames. By extracting CNN-based features from the I-frame and the ME map, we aim to model the impact of both spatial content features and temporal motion features on the quality of P-frames. The spatio-temporal feature-based P-frame patch distortion predictor network (ST-D-NET) is depicted in Figure 4.7, consisting of two CNN pipelines: S-D-CNN for spatial feature extraction and T-D-CNN for temporal feature extraction. The S-D-CNN, as discussed in Section 4.1, has already been trained using I-frame patches for content and quality. Figure 4.8 illustrates the CNN layers of the T-D-CNN network, which closely resembles the S-D-CNN pipeline but consists of six CNN layers. We conducted experiments to determine the optimal number of layers for T-D-CNN and observed that reducing the number of CNN layers from ten to six did not compromise the accuracy of P-frame quality prediction. Therefore, the T-D-CNN comprises six CNN layers followed by two fully connected layers, FC3 and FC4, as shown in Figure 4.7. Table 4.4 provides an overview of the CNN layer parameters, where layers 1 to 9 denote CNN layers followed by pooling layers, and layers 12 to 13 represent FC3 and FC4 (fully connected layers) depicted in Figure 4.7.

To train the ST-D-NET, it is only necessary to train the T-D-CNN, as the S-D-CNN has already been trained. Hence, the T-D-CNN is designed to take patches of the ME map extracted from each P-frame as input and predict the temporal component of perceptual quality of a P-frame's patch at five QPs: $\{QPs \in 28, 32, 36, 40, 44\}$. On the other hand, the trained S-D-CNN takes patches of the primary I-frame as input

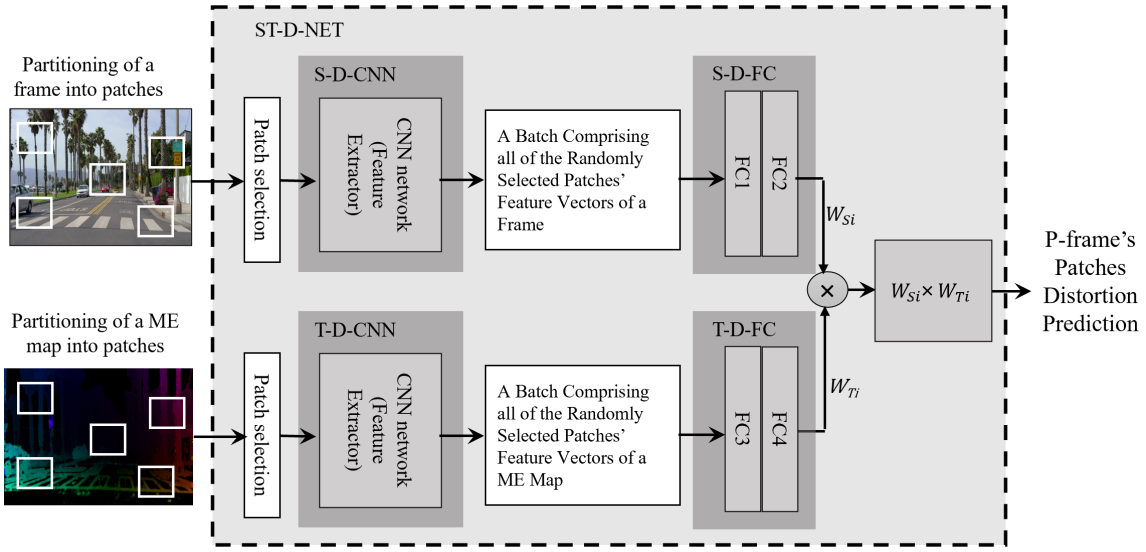

Figure 4.7: Frames' perceptual quality predictors architecture.

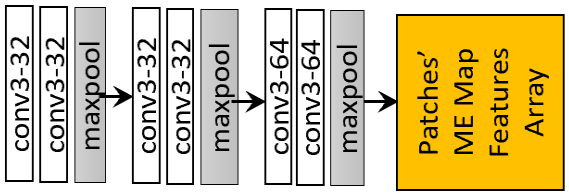

Figure 4.8: Deep CNN layers of ME map feature extractor.

and predicts an array comprising the perceptual quality of an I-frame's patch at the same five QPs: $\{QPs \in 28, 32, 36, 40, 44\}$, as its output. The output of the ST-D-NET, as shown in Equation 4.4, represents the predicted quality of the P-frame's patch.

$$Q_{Pi} = W_{Ti} \times W_{Si} \tag{4.4}$$

Where W_{Ti} is the predicted temporal weight of the i'th patch for P-frame quality prediction and W_{Si} is the predicted I-frame patch quality. Q_{Pi} is the quality of the P-frame i'th patch.

The cost function used for training T-D-CNN is stated in Equation 4.5, in which $\psi_I(\cdot)$ and $\psi_P(\cdot)$ are mapping functions that map I-frame and ME map patches to I-frames' patches' quality and ME map weights, respectively; P_{MEi} is the i'th ME patch and ω_P is the vector of weights of the mapping function ψ_P; ω_I^* is the vector of trained weights of the mapping function ψ_I and Q_{Pi} is the VMAF of a P-frame's i'th patch.

$$J = |Q_{Pi} - \psi_I(P_i, \omega_I^*) \times \psi_P(P_{MEi}, \omega_P)| \tag{4.5}$$

Table 4.4: Layers of deep CNN network for P-frame's bit rate predictor

#	Type	Kernel	Stride	Activation	Outputs
01	Conv.	3×3	1×1	ReLU	32
02					
03	Pool.	-	2×2	-	32
04	Conv.	3×3	1×1	ReLU	32
05					
06	Pool.	-	2×2	-	32
07	Conv.	3×3	1×1	ReLU	64
08					
09	Pool.	-	2×2	-	64
10					120
11	FC			ReLU	5
12					120
13	FC			ReLU	5

4.4 Patch-wise to Frame-wise Perceptual P-Frame Quality Prediction Transformation

In this section, we present three approaches aimed at predicting the P-frame quality with the utmost accuracy, leveraging the quality predictions of the patches.

4.4.1 Homogeneous Average Method for P-Frame Quality Prediction

The Homogeneous Average Spatio-Temporal Feature-based Distortion Prediction Predictor Network (HA-ST-D-NET) utilizes the S-D-NET network to predict the quality of randomly selected patches and calculates the average of the predicted patch qualities for each frame. The architecture of HA-ST-D-NET is illustrated in Figure 4.9. Similar to the prediction of I-frame quality, we aimed to determine the minimum number of patches required to reduce computational costs. The experimental results

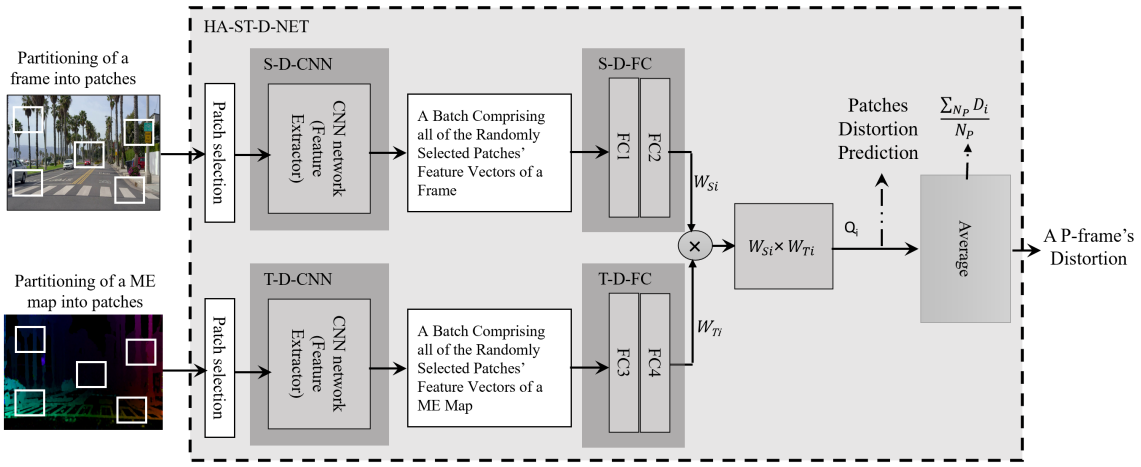

Figure 4.9: Architecture of homogeneous average method for P-frames' perceptual quality prediction.

demonstrate that using either 10% or 100% of the patches yields comparable frame quality prediction accuracy.

4.4.2 Sorted Average Method for P-Frame Quality Prediction

The proposed Sorted Average Quality Prediction method (SA-ST-D-NET) is based on our observation that the average quality of patches is consistently lower than or equal to the quality of the entire frame. This observation aligns with the visual masking phenomenon of the human visual system (HVS), where the brain tends to mask image artifacts. The structure of the Sorted Average P-Frame Quality Predictor (SA-ST-D-NET) is depicted in Figure 4.10. SA-ST-D-NET sorts the qualities of patches within a frame and calculates the average of a subset of predicted qualities with higher values. Different percentages of sorted patch qualities ($S_{ptch}\%$) are utilized to predict frame quality, while minimizing the prediction error of SA-ST-D-NET. Table 4.5 presents

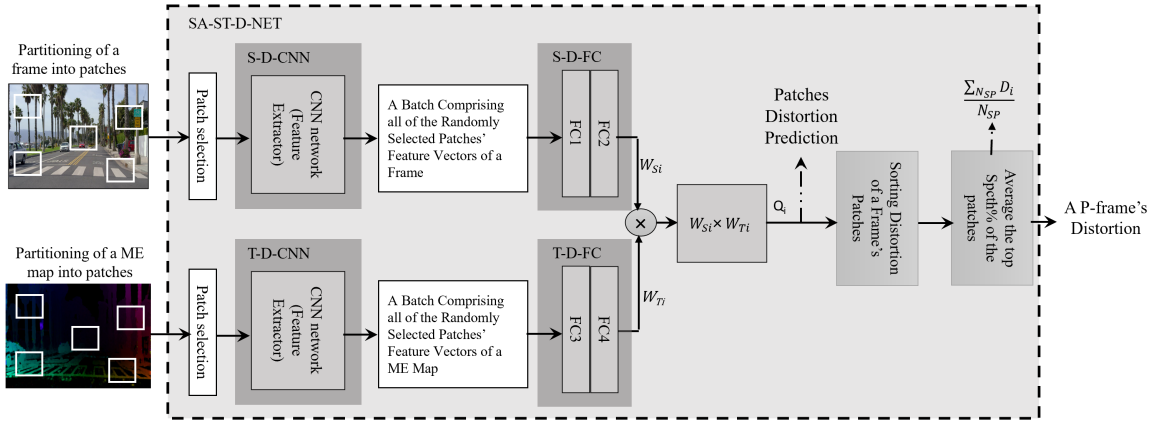

Figure 4.10: Architecture of sorted average method for P-frames' perceptual quality prediction.

the Mean Absolute Error (MAE) of the sorted average transformation method for various S_{ptch}% values at five Quantization Parameters (QPs). The training dataset is employed to generate Table 4.5. The minimum MAE values are highlighted in bold font, indicating the corresponding S_{ptch}% values to be used for predicting frame-wise quality at the designated QPs.

Table 4.5: Average MAE of Predicting P-frame quality (VMAF) by using sorted averaging method with different percent of sorted patches' VMAF at different QPs.

QP	28	32	36	40	44
10%	**2.97**	5.53	7.54	14.9	21.63
20%	2.99	4.75	6.54	12.25	17.51
40%	3.38	**4.65**	**6.23**	9.99	13.5
60%	4.12	4.87	6.47	**9.68**	**12.15**
80%	5.31	5.97	7.07	10.2	12.37
100%	6.86	8.1	8.74	11.68	13.08

4.4.3 Weighted Average P-Frame Quality Prediction Method

The Weighted Average Spatio-Temporal Feature-based P-Frame's Distortion Predictor Network (WA-ST-D-NET), illustrated in Figure 4.11, is introduced as an alternative approach to the sorted method (SA-ST-D-NET). While SA-ST-D-NET considers only high-quality patches in the average calculation for frame-wise quality prediction, WA-ST-D-NET employs a fully connected network to estimate the weight of each patch in the quality calculation, as indicated in Equation 4.6.

$$Q_{PF} = \frac{\sum_{i=0}^{N_P} |Q_i \times W_{Si} \times W_{Pi}|}{N_P}, \tag{4.6}$$

where Q_i is the predicted quality of the i'th patch of an I-frame, W_{Pi} is the weight to predict a P-frame's i'th patch quality, and W_{Si} is the i'th patch weight to provide weighted average that calculates a P-frame quality. N_P is the number of random patches extracted from each P-frame. For optimizing W_{Si}, two fully connected layers (FC-3 and FC-4) shown in Figure 4.11 take CNN-based extracted spatial features of S-D-CNN network as input, VMAF of a P-frame as the target and train FC-3 and FC-4 optimize the cost function defined in Equation 4.7.

$$J = |Q_{PF} - \sum_{i=0}^{N_P} (\tau_P(\psi_{Feature}(P_i, \omega_I^*), \theta_P) \times \psi_I(P_i, \omega_I^*) \times \psi_P(P_{MEi}, \omega_P^*))| \tag{4.7}$$

where τ_P is a mapping function that maps a batch of patches' features to a patch's weight for predicting P-frame quality. Q_{PF} is the P-frame's VMAF, $\psi_{Feature}$ is part of ψ_I mapping function and maps patches to their extracted feature map, ω_P^* is optimized weights of trained ψ_P mapping function, θ_P is the weight of τ_P mapping function and N_P is the number of random patches utilized for predicting P-frame

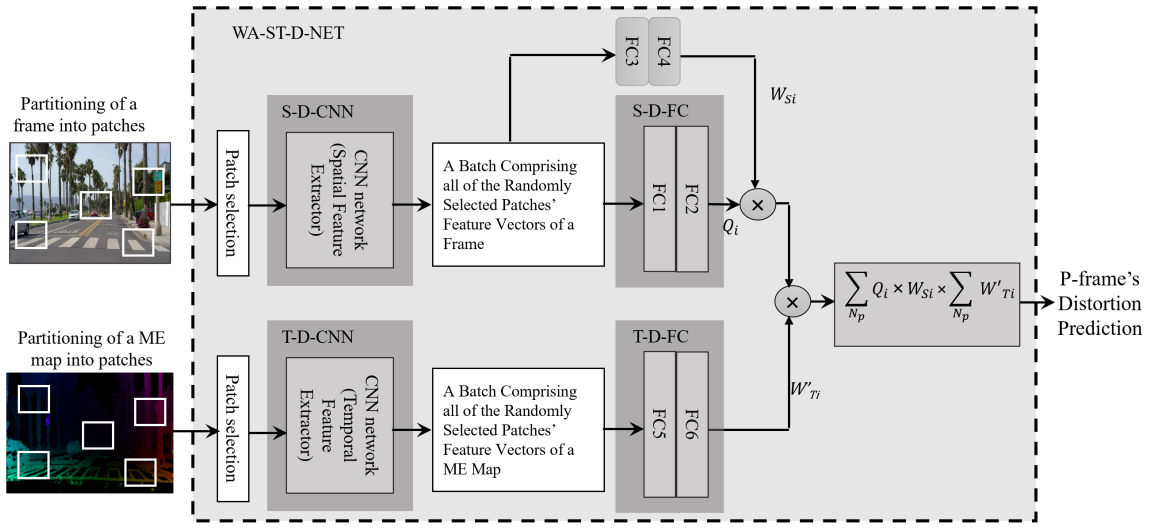

Figure 4.11: Architecture of weighted average method for P-Frames' perceptual quality prediction.

perceptual quality.

4.5 Experiments

The proposed perceptual quality prediction methods were trained and tested using a well-known public video dataset introduced in [33]. Q. Huang et al. [33] extensively studied the spatial and temporal features of this dataset and demonstrated its suitability for training and testing image processing systems due to its wide range of spatial and temporal characteristics. In this chapter, a subset of 300 tracks from [33] was selected for training and testing. The videos in the dataset were in HD resolution (1920×1080), and any videos with higher resolutions were downscaled to HD. The sample videos were split at key frames to form dataset tracks, each consisting of 16 frames in the $IPP...$ GOP format. Thus, a total of 300 tracks were utilized for training and testing the proposed deep CNN networks.

Since the proposed methods operate at the patch level, each frame was divided into blocks of size 64×64. Consequently, a dataset of $100,000$ patch samples was extracted from the tracks dataset. As discussed in Section 4.4.1, it was found that using only 10% of the patches yielded sufficient accuracy for frame-level quality prediction. Therefore, for each frame in a track, 10% of the patches were randomly selected multiple times (100 times). This resulted in a total of $30,000$ sample patches. To establish the training, testing, and validation sets, 75% of this data was randomly assigned for training, 15% for testing, and 10% for validation purposes. The tracks were encoded using H.265 at quantization parameters $QP \in 28, 32, 36, 40, 44$, and the VMAF [24] metric was employed to evaluate the perceptual quality of the frames.

4.5.1 Patch-Wise I-frame Quality Prediction Accuracy

To evaluate the accuracy of predicting quality for I-frame patches, the encoded I-frames were divided into patch sizes, and their corresponding VMAF scores were calculated. These VMAF scores were then compared to the predicted quality obtained using our proposed method. The perceptual quality predictor network for patches, as depicted in Figure 4.1, takes the patches from the frames as input and predicts their VMAF scores.

The prediction accuracy of the patch-wise quality was assessed using the Pearson metric (PLCC) [62], which measures the correlation between the predicted and actual VMAF scores. Additionally, the monotonicity of the predicted patch quality was evaluated using the Spearman metric (SROCC) [30]. Table 4.6 presents the results, demonstrating the high accuracy achieved in the prediction of patch-wise quality. The PLCC values indicate a strong correlation between the predicted and actual

VMAF scores, while the SROCC values indicate a monotonic relationship between the predicted and actual patch qualities.

Table 4.6: PLCC and SROCC of predicted I-frames' patches quality prediction.

QP	28		32		36		40		44	
	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC
	0.85	0.79	0.86	0.83	0.88	0.87	0.91	0.9	0.91	0.89

4.5.2 I-frame Perceptual Quality Prediction Accuracy

Table 4.7 provides the average absolute error of the predicted VMAF scores for the I-frames of 17 sample videos using the three proposed methods. The mean absolute errors (MAE) of the sorted, homogeneous, and weighted average quality prediction methods are denoted as S_{avg}, H_{avg}, and W_{avg} in the table, respectively.

As observed in Table 4.7, the sorted and weighted average methods outperform the homogeneous average quality prediction method significantly. At lower quantization parameters (QPs) such as 28, 32, and 36, the performance of the sorted and weighted average methods is comparable. However, at higher QPs such as 40 and 44, where the compression distortion becomes more prominent, the sorted average method exhibits lower MAE compared to the weighted average method.

This difference can be explained by considering that at higher QPs, the quality experiences substantial distortion, leading to a wider range of frame distortion and quality values. Consequently, training fully connected networks with such a broad range of outputs may require a larger number of image samples to prevent overfitting and ensure accurate prediction. Therefore, at higher QPs, the sorted average quality predictor demonstrates better performance than the weighted average method.

Table 4.7: Mean Absolute Error of predicting I-frames' VMAF with three presented pre-encoding perceptual quality prediction methods.

QP	28			32			36			40			44		
	S_{avg}	H_{avg}	W_{avg}	S_{avg}	H_{avg}	W_{avg}	S_{avg}	H_{avg}	W_{avg}	S_{avg}	H_{avg}	W_{avg}	S_{avg}	H_{avg}	W_{avg}
1.aspen	**0.55**	2.91	0.9	**0.69**	4.12	1.14	**0.49**	4.84	3.07	**2.27**	5.48	7.41	**3.54**	5.68	3.88
2.blue sky	**0.59**	3.01	1.98	**0.71**	3.42	1.18	**0.86**	2.87	5.9	4.13	**1.48**	7.72	8.81	**3.88**	6.34
3.controlled burn	1.3	5.94	**0.76**	1.74	8.25	**1.21**	3.35	10.1	**1.66**	**2.42**	11.95	5.83	**4.07**	12.89	5.76
4.ducks take off	1.01	3.19	**0.3**	1.39	4.71	**1.26**	2.12	5.67	**0.66**	**1.83**	6.79	2.18	4.17	8.87	**3.25**
5.in to the tree	**0.15**	6.11	1.09	**0.4**	9.66	3.14	4.32	12.72	**2.64**	**2.81**	15.71	13.2	**7.58**	16.96	9.54
6.old town cross	2.49	7.36	**2.43**	4.09	9.74	**2.71**	5.99	9.7	**0.46**	**2.28**	7.63	10.97	5.04	3.32	**0.94**
7.park joy	**0.25**	5.39	1.85	**0.16**	7.54	1.88	0.62	8.98	**0.37**	**0.24**	10.03	4.59	**1.76**	10.09	6.63
8.pedestrian area	0.79	5.25	**0.21**	**0.27**	6.53	0.63	**0.94**	7.21	3.16	2.95	7.26	**2.28**	**3.27**	6.52	3.69
9.red kayak	**1.0**	4.87	1.98	**1.06**	6.66	1.15	**1.34**	7.54	2.2	**1.24**	8.15	6.66	**1.22**	8.06	2.47
10.rush field cuts	**0.08**	3.69	1.42	**0.08**	5.77	0.79	**0.74**	7.47	2.32	**0.28**	9.15	5.67	**2.83**	10.48	7.22
11.tractor	**0.99**	4.32	1.06	**1.68**	6.45	1.81	4.16	8.34	**3.4**	**6.87**	10.62	11.57	9.9	12.98	**6.32**
12.ritual dance	**0.13**	5.44	0.36	**0.17**	7.34	0.88	**0.59**	8.5	2.7	**3.54**	9.01	3.39	5.25	8.7	**3.93**
13.touchdown-pass	1.35	6.8	**0.43**	**2.27**	9.6	2.67	5.18	11.18	**1.8**	**2.47**	11.46	10.38	**5.91**	9.91	8.66
14.Driving-POV	0.78	5.2	**0.28**	**0.12**	6.61	0.9	**0.91**	7.83	2.71	**1.28**	8.77	4.92	**0.63**	9.32	2.24
15.Pier Seaside	1.0	5.3	**0.25**	**0.07**	6.06	1.04	**1.08**	6.12	1.85	**1.79**	5.09	3.45	**1.97**	3.19	3.39
16.Cross Walk	**1.04**	6.74	1.81	3.16	9.12	**0.3**	4.04	9.92	**0.47**	**2.35**	10.93	12.81	**4.2**	11.14	4.97
17.Square and Timelapse	3.53	8.33	**0.39**	4.77	10.23	**1.25**	6.55	11.31	**2.36**	**3.39**	10.93	5.84	**1.65**	9.54	3.6
AVERAGE	**1.0**	5.2	**1.0**	**1.34**	7.16	1.41	2.54	8.25	**2.22**	**2.48**	8.85	6.99	**4.22**	8.91	4.87

4.5.3 Patch-Wise P-frame Quality Prediction Accuracy

To evaluate the accuracy of P-frame patch quality prediction, the encoded P-frames were segmented into patch sizes, and their corresponding VMAF scores were calculated. These VMAF scores were then compared with the predictions generated by our proposed method.

Table 4.8 presents the prediction accuracy at different quantization parameters

Table 4.8: PLCC and SROCC of predicted P-frames' patches quality prediction.

QP	28		32		36		40		44	
	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC
	0.72	0.69	0.71	0.68	0.71	0.68	0.70	0.67	0.70	0.66

(QP), specifically $\{QP \in 28, 32, 36, 40, 44\}$, as measured by the Pearson metric (PLCC) and the monotonicity of predicted patch quality assessed by the Spearman metric (SROCC).

When comparing the prediction results of I-frame patch quality (Table 4.6) with those of P-frame patch quality (Table 4.8), it is evident that I-frame patch quality prediction achieves higher accuracy and monotonicity compared to P-frame patch quality prediction. This difference can be attributed to the greater complexity involved in preparing P-frames and the addition of temporal features in the prediction process. The utilization of temporal features increases the complexity of the network, thereby affecting the prediction accuracy. Hence, it is reasonable to observe superior prediction performance for I-frame patch quality due to the absence of these additional complexities.

4.5.4 P-frame Perceptual Quality Prediction Accuracy

Table 4.9 presents the mean absolute error (MAE) of the predicted VMAF scores for P-frames in 17 sample videos using three proposed P-frame quality prediction methods. The MAE values for the sorted, homogeneous, and weighted average quality prediction methods are denoted as S_{avg}, H_{avg}, and W_{avg} in the table, respectively.

As observed in Table 4.9, the sorted method outperforms both the weighted and homogeneous average quality prediction methods significantly. The sorted average method consistently demonstrates superior performance compared to the other two

methods.

It is important to note that as the QP increases, indicating higher compression distortion, the prediction of frame quality becomes more challenging. However, even at higher QPs like 40 and 44, the sorted average method exhibits lower MAE values than the weighted and homogeneous average methods.

This suggests that the sorted average quality prediction method is effective in capturing and predicting the perceptual quality of P-frames, even under high compression distortion conditions.

Table 4.9: new: Absolute error of predicting P-frames' VMAF by three different method

QP	28			32			36			40			44		
	S_{avg}	H_{avg}	W_{avg}	S_{avg}	H_{avg}	W_{avg}	S_{avg}	H_{avg}	W_{avg}	S_{avg}	H_{avg}	W_{avg}	S_{avg}	H_{avg}	W_{avg}
1.Aspen	**2.49**	3.04	6.66	4.34	**3.94**	7.5	**3.5**	4.07	5.87	6.85	**4.97**	9.18	5.26	**4.63**	7.01
2.Blue Sky	**1.3**	4.89	27.93	**0.49**	2.83	39.64	2.95	**2.38**	28.2	6.83	**5.71**	25.26	11.42	**9.94**	21.84
3.Controlled Burn	**4.45**	9.73	10.85	**6.53**	11.01	14.19	**5.58**	10.29	13.18	**5.26**	10.56	10.25	**3.7**	7.52	9.62
4.Ducks Take off	**0.41**	3.9	3.35	**0.87**	5.25	3.71	3.93	8.53	**3.33**	6.97	11.55	**5.19**	11.39	15.35	**4.28**
5.In to The Tree	**1.88**	11.47	19.92	**1.41**	18.22	24.43	**10.41**	22.37	17.93	18.72	30.28	**18.14**	16.73	26.36	**14.8**
6.Old Town Cross	**1.17**	8.62	7.47	3.81	10.6	**1.05**	7.58	13.05	**0.64**	7.47	15.86	**1.68**	**6.3**	18.26	10.09
7.Park Joy	**2.01**	6.96	9.06	**1.39**	10.17	21.51	**4.84**	15.37	25.68	**12.25**	22.25	21.83	19.59	26.92	**13.04**
8.Pedestrian area	**2.75**	8.05	8.85	**5.21**	7.91	11.68	8.14	**6.74**	14.45	10.91	**9.86**	12.94	13.75	11.97	**6.66**
9.Red Kayak	**5.9**	10.1	11.39	**6.66**	12.18	7.46	**5.35**	11.42	5.48	**3.91**	11.48	5.57	**5.93**	7.43	8.52
10.Rush Field Cuts	**1.33**	5.27	2.08	**3.72**	5.39	5.13	**3.85**	5.92	7.4	**3.3**	5.93	3.95	**1.43**	5.72	2.91
11.Tractor	**5.09**	10.71	10.96	**7.3**	13.97	17.41	**6.68**	14.91	18.32	**10.05**	17.79	14.19	7.55	14.17	**6.73**
12.Ritual Dance	**1.57**	9.34	4.73	**3.79**	13.23	10.55	**6.22**	16.83	15.17	**10.32**	21.98	18.82	**12.37**	23.26	22.25
13.Touchdown-pass	**5.18**	14.79	8.02	8.61	19.74	**4.29**	**8.17**	19.81	9.66	14.91	23.75	**8.92**	10.28	16.16	**8.22**
14.Driving-POV	3.34	**2.21**	12.95	4.14	**2.1**	8.66	4.0	**2.68**	8.04	3.91	**3.2**	11.86	4.07	**3.34**	19.96
15.Pier Seaside	**1.17**	5.46	18.42	**1.47**	4.2	16.52	6.07	**1.67**	16.46	7.46	**2.01**	18.74	15.39	**8.07**	26.45
16.Cross Walk	**0.99**	5.66	10.76	**1.43**	5.57	14.14	4.1	**3.45**	13.65	**2.04**	2.68	9.61	5.95	**1.18**	1.25
17.Square and Timelapse	**1.78**	4.2	1.93	**1.33**	6.52	3.65	**2.47**	10.2	4.24	6.72	14.88	**3.8**	11.48	18.98	**6.89**
AVERAGE	**2.52**	7.32	10.31	**3.68**	8.99	12.44	**5.52**	9.98	12.22	**8.11**	12.63	11.76	**9.56**	12.9	11.21

Chapter 5

Deep-learning based bit-rate prediction of I-frames

The high-efficiency video coding (HEVC) is known for its improved rate-distortion performance compared to earlier encoding algorithms. Within the encoding architecture, the rate control (RC) module plays a crucial role. The RC module allocates bit rate to blocks, frames, and GOPs, to meet the target bit rate. While there have been numerous studies on video rate control, there is a need for further research on bit rate prediction prior to compression.

In this chapter, we introduce two novel CNN networks that predict bit rates for both I-frames and P-frames without the need for actual compression. Our approach enables the prediction of frame bit rates at various quantization parameters (QPs) before the encoding process. Consequently, the encoder can choose the optimal QP value to maintain the desired bit rate level. Unlike previous methods that rely on multi-pass encoding or exploit rate-distortion information from prior frames, our method provides accurate bit rate prediction without the need for actual compression.

5.1 Bit Rate Allocation

Rate control methods play a crucial role in minimizing distortion while adhering to bit rate constraints. To achieve this, rate control involves the characterization of the relationship between rate and distortion through the introduction of rate-distortion models and the definition of rate-distortion cost functions aimed at minimizing distortion. Several well-known rate control and rate-distortion models have been proposed, including the Q-R model [32], [39], exponential [50], linear [21], logarithmic [84][101], and ρ [87] models. An example of such a model is the Q-R model, as shown in Equation 5.1.

$$R = \alpha_{QR} \times Q_{step}^{(\beta_{QR})}, \tag{5.1}$$

where R is the bit rate, α_{QR} and β_{QR} are Q-R model content-dependent parameters, and Q_{step} is the quantization step size. Equation 5.2 shows the relation between Q_{step} and the quantization parameter (QP).

$$Q_{step} = 2^{\frac{QP-4}{6}} \tag{5.2}$$

which results in equation 5.3.

$$R = \alpha_{QR} \times 2^{(\beta_{QR} \times QP)} \tag{5.3}$$

The rate-distortion linear model is shown in Equation 5.4.

$$D = \frac{\beta}{R^{\alpha}} \tag{5.4}$$

where D and R are distortion and bit rate, respectively. α and β are model parameters and are content-dependent. The cost function for R-D performance optimization is formulated as the constrained optimization problem in Equation 5.5 and is reformulated as the unconstrained problem in Equation 5.6.

$$\begin{aligned} \min_{R_i} \quad & \sum_{i=1}^{N} D_i(R_i) \\ \text{s.t.} \quad & \sum_{i=1}^{N} R_i \leq R_{Fr} \end{aligned} \tag{5.5}$$

$$J = \sum_{i=1}^{N} D_i(R_i) + \lambda(R_{Fr} - \sum_{i=1}^{N} R_i) \tag{5.6}$$

where D_i, and R_i are the i'th patch distortion and bit rate, respectively. By considering Karush-Kuhn-Tucker (KKT) [13] conditions, the patch's optimal target bit rate allocation is achieved and shown in Equation 5.7.

$$R_j = \frac{\nu_j}{\sum_{i=1}^{N} \nu_i} R \tag{5.7}$$

where R_j is the j'th patch's allocated bit rate, R is the total frame bit rate, and ν is the model parameter. Zhou in [101] proved that the local optimum of Equation 5.7 is the global optimum bit rate. Considering Equation 5.7, frames' bit rate can be predicted at the local patch level, then expanded to the frame level.

Predicting the bit rate of a frame poses several challenges due to the diverse content and texture present in different blocks of the frame. Training CNN networks to accurately predict frame bit rates requires a vast number of samples to capture

the wide range of texture and content combinations. Additionally, the computational complexity involved in achieving accurate predictions, particularly for high-resolution videos, can be substantial. The inherent dynamic range of frame bit rates, resulting from variations in texture, further exacerbates the difficulty of accurate prediction.

This chapter introduces an innovative patch-wise bit rate prediction method based on CNNs, which addresses the aforementioned challenges and provides a practical approach for frame bit rate prediction. The proposed method is applicable to frames of different resolutions and various encoders, making it versatile in its application. Furthermore, it significantly reduces computational costs by processing only ten percent of a frame's patches to predict its bit rate.

5.2 Deep CNN Network Architecture of I-frames' Patches Bit Rate Predictor

Figure 5.1 depicts the structure of the bit rate predictor for I-frame patches. The proposed CNN-based predictor, known as S-BR-NET, takes frames' patches as input and predicts the corresponding patches' bit rates. To prepare the bit rate values for training S-BR-NET, the frames of the video dataset were divided into patches prior to encoding. These patches were then encoded in I-frame mode at five different quantization parameters (QPs), specifically $QP \in 28, 32, 36, 40, 44$. The resulting bit rates of the encoded patches at each QP were stored for training purposes. The trained S-BR-NET consists of a CNN-based spatial feature extractor network (S-BR-CNN), which is also trained concurrently during this training procedure.

As described in Chapter 4, the VGG network is a suitable choice for designing a

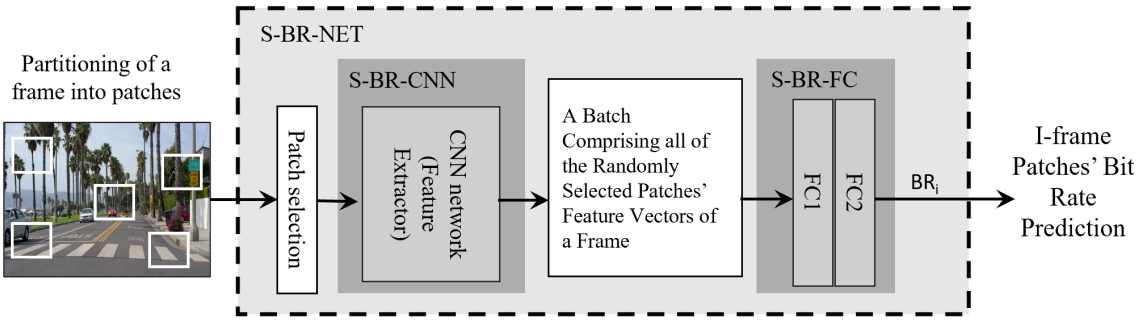

Figure 5.1: Patches' bit rate predictor architecture.

CNN architecture for bit rate prediction in encoding. Table 5.1 outlines the layers of the patch-wise bit rate predictor, ranging from Layer 1 to Layer 18. To prevent overfitting and excessive computational complexity, a pooling layer follows every two CNN layers. The max-pool layers have a kernel size of 2×2 and a stride size of 2×2. Each layer utilizes a 3×3 kernel size to reduce computational complexity and leverage the non-linear characteristics of the network. Zero-padding is applied to the input of each layer to maintain a constant size after convolving with the kernel. The stride of each CNN layer is set to 1×1 to preserve the output size. The Rectified Linear Unit (ReLU) activation function [64] is used in all the CNN layers. Following the pooling layer (Layer 15), two fully connected layers are employed to map the extracted spatial features to the bit rate array. The output size of Layer 15 consists of 64 feature maps with dimensions $S_P/2^5 \times S_P/2^5$, where S_P represents the patch size. The output sizes of FC1 and FC2 are 120 and 5, respectively.

The cost function, is defined in Equation 5.8, comprises the mapping function $\psi_{IBR}(\cdot)$, where P_i and ω in $\psi_{IBR}(P_i, \omega_{IBR})$ are i'th patch and weights of ψ respectively. $\psi_{IBR}(\cdot)$ maps patch features to the bit rate of each patch. Equation (5.8) shows the cost function of the patch-wise bit rate predictor. BR_{Ii} is the bit rate of the i'th

patch of an I-frame.

$$J = |BR_{Ii} - \psi(P_i, \omega_{IBR})| \tag{5.8}$$

Table 5.1: Layers of deep learning CNN network of pre-encoded I-frame's bit-rate predictor

#	Type	Kernel	Stride	Activation	Outputs
01	Conv.	3×3	1×1	ReLU	32
02					
03	Pool.	-	2×2	-	32
04	Conv.	3×3	1×1	ReLU	32
05					
06	Pool.	-	2×2	-	32
07	Conv.	3×3	1×1	ReLU	64
08					
09	Pool.	-	2×2	-	64
10	Conv.	3×3	1×1	ReLU	64
11					
12	Pool.	-	2×2	-	64
13	Conv.	3×3	1×1	ReLU	128
14					
15	Pool.	-	2×2	-	128
16					120
17	FC			ReLU	5
18					120
19	FC			ReLU	5

5.3 Deep CNN I-frame Bit Rate Predictor

The relationship between a frame's optimized bit rate and its block-level predicted bit rates is depicted in Equation 5.7. Therefore, predicting the bit rates of patches can lead to predicting the bit rate of a frame. However, it is important to note that

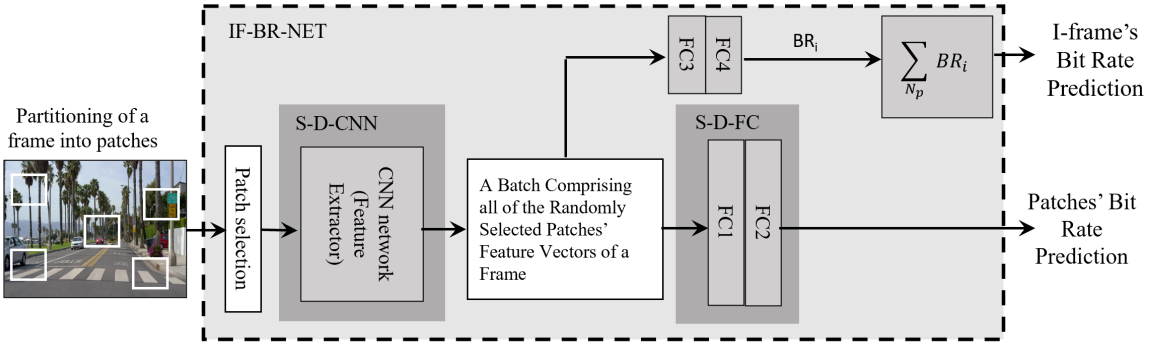

Figure 5.2: Frames' bit rate predictors architecture.

the encoded bit rate of chopped patches is higher than the allocated bit rate when the patch is encoded as part of a frame. In our approach, we utilized the high-efficiency video coding (HEVC) standard [80] for frame and patch encoding. When HEVC encodes a frame, it divides the frame into blocks at four different depth levels or coding units (CUs), namely CU_0, CU_1, CU_2, and CU_3, which correspond to sizes of 64×64, 32×32, 16×16, and 8×8, respectively. The selection of CU block sizes is based on the complexity of the I-frame texture, where smaller CUs are allocated to high-texture areas and larger CUs to low-texture areas. In regions with low complexity texture, lower level CUs (CU_0 and CU_1) are encoded with a smaller number of DCT coefficients. Consequently, the sum of the bit rates of a frame's encoded patches is always higher than the bit rate of the frame itself.

Taking into account the observations mentioned above, we have developed an innovative I-frame bit rate predictor called IF-BR-NET, as illustrated in Figure 5.2. This method predicts both the bit rates of individual patches and the total bit rate of the I-frame. The calculation of the predictor's output is formulated in Equation 5.9. By considering the characteristics of the encoding process and the block-level predictions, IF-BR-NET is able to estimate the bit rates accurately and provide

valuable insights into the overall bit rate of the I-frame.

$$R_{IF} = \sum_{N_P} BR_{Bi}(QP) \tag{5.9}$$

In the proposed IF-BR-NET, where R_{IF} represents the bit rate of the I-frame, N_P denotes the number of randomly extracted patches used for predicting the I-frame's bit rate, and BR_{Bi} corresponds to the bit rate of the i'th random patch an assigned QP. To train IF-BR-NET, a batch containing all the random patches extracted from a frame is fed as input, while the target output is the bit rate of that specific I-frame. The training process involves two main steps. First, the S-BR-CNN network, which has been trained to extract spatial features from the patches, processes the input batch. Subsequently, the S-IF-FC network, comprising two fully connected layers (FC3 and FC4), shown as layers 18 and 19 in Table 5.1, takes the batch of spatial feature vectors as input. The training of FC3 and FC4 is accomplished by comparing the summation of the patches' bit rates, as depicted in Equation 5.9, with the actual bit rate of the I-frame. The cost function for training FC3 and FC4 is defined in Equation 5.10.

$$J = |BR_{IF} - \sum_{i=0}^{N_P} \tau_{IBR}(\psi_{IBR_{feature}}(P_i, \omega*), \theta_{IBR})| \tag{5.10}$$

where τ_{IBR} maps a batch of patches' feature vectors to the I-frame's bit rate. BR_{IF} is the actual I-frame bit rate, $\psi_{IBR_{feature}}$ maps patches to their feature vectors, ω^*_{IBR} is optimized weights of ψ_{IBR} mapping function calculated with loss function of Equation 5.8, θ_{IBR} is the weight of τ_{IBR} mapping function, and N_P is the number of random

patches utilized for prediction I-frame bit-rate. To address the limited number of I-frames available for training, an augmentation technique was employed. Specifically, 10% of the patches from an I-frame were randomly selected and combined into a batch, serving as a training sample. This process was repeated a hundred times, each time randomly selecting a different set of patches. As a result, the number of training samples for I-frames was augmented by a factor of a hundred, providing a larger and more diverse dataset for training the model.

5.4 Deep CNN based P-frame's Patches' Bit Rate Predictor

P-frames, being based on the residue between frames and previous I-frames, are influenced by both textural and non-textural characteristics of the frame sequence, which in turn affect their bit rate and quality. To extract textural features, patches from raw I-frames are fed through the S-BR-CNN network, which maps them to spatial feature vectors. On the other hand, for capturing the temporal or non-textural features of P-frames, a dense motion estimation (ME) technique is employed. This dense ME map captures the disparities between P-frames and the corresponding I-frame. Subsequently, as depicted in Figure 5.3, a CNN structure named T-BR-CNN is utilized to map the dense average ME maps to temporal feature vectors.

To predict P-frame bit rates at the patch level, the GOPs (Group of Pictures) are partitioned into patch sizes. Each partition is then encoded separately using five different Quantization Parameters (QPs), specifically chosen to encompass the practical dynamic range of each video bit rate. Since the P-frame bit rate is influenced

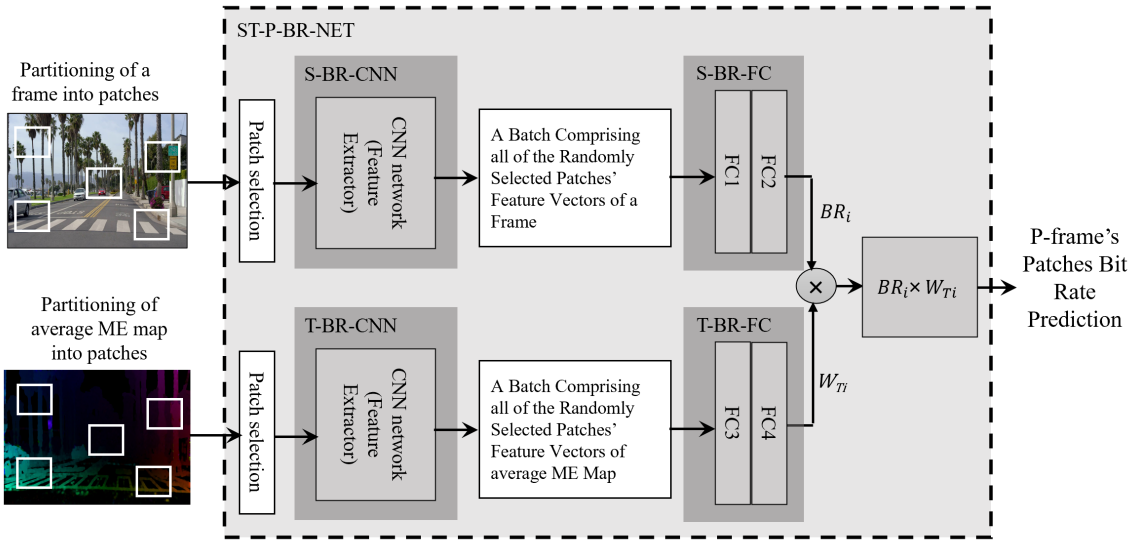

Figure 5.3: Frames' bit rate predictors architecture.

by the residue and can vary significantly even between successive P-frames within a GOP, the proposed method addresses this issue by labeling the patches of P-frames with their average bit rate. This approach helps mitigate the problem of P-frame bit rate fluctuations. Additionally, an average Motion Estimation (ME) map is prepared for each GOP. This average ME map is generated by taking the average of the ME maps of the P-frames within the GOP.

Figure 5.3 illustrates the CNN pipeline T-BR-CNN, which takes the average Motion Estimation (ME) map of patches and extracts temporal features specific to the GOP. T-BR-CNN and S-BR-CNN can operate in parallel, and their output vectors can be concatenated or passed through regressors and multiplied to generate predictions for the bit rates of P-frame patches. Figure 5.3 (ST-P-BR-NET) presents an overview of the P-frame patch-wise bit rate predictor. ST-P-BR-NET takes I-frame patches and the average ME map of P-frames as input. It predicts the bit rates of

Table 5.2: Layers of deep learning CNN network of pre-encoded I-frame's bit rate predictor

#	Type	Kernel	Stride	Activation	Outputs
01	Conv.	3×3	1×1	ReLU	32
02					
03	Pool.	-	2×2	-	32
04	Conv.	3×3	1×1	ReLU	32
05					
06	Pool.	-	2×2	-	32
07	Conv.	3×3	1×1	ReLU	64
08					
09	Pool.	-	2×2	-	64
10					128
11	FC			ReLU	5
12					128
13	FC			ReLU	5

the I-frame patches using S-BR-CNN and multiplies the result by the output of T-BR-CNN. The multiplication result is then compared to the actual average bit rates of the P-frame patches to determine the prediction error, which is utilized to train T-BR-CNN.

Table 5.2 provides an overview of the layers in T-BR-CNN, which consists of six CNN layers. Following every two CNN layers, there is a max-pool layer (layers 1 to 9). A regressor is employed after the CNN layers, comprising two fully connected layers (layers 10 to 11). The cost function used to train the CNN layers and fully connected layers in T-BR-CNN is specified in Equation 5.11.

$$J = |BR_{Pi} - \psi_{IBR}(P_i, \omega^*_{IBR}) \times \psi_{PBR}(ME_i, \omega_{PBR})| \tag{5.11}$$

where ME_i and ω_{PBR} in $\psi_{PBR}(P_i, \omega_{PBR})$ are i'th patch of average ME map and weights of ψ_{PBR}, respectively. $\psi_{PBR}()$ maps ME feature vectors to the bit rate of

each patch. BR_{Pi} is the bit-rate of i'th patch of a frame.

5.5 Deep CNN Network Architecture for GOP-Level P-frame Bit Rate Predictor

The GOP-level P-frame bit rate predictor (PF-BR-NET) is depicted in Figure 5.4. PF-BR-NET consists of S-BR-CNN and T-BR-CNN networks responsible for extracting spatial and temporal features from P-frames. To facilitate the patch-to-frame transformation module of PF-BR-NET, two regressors are utilized. The first regressor includes FC3 and FC4, while the second regressor is a pipeline of two fully connected layers, FC7 and FC8. The training process for FC3 and FC4 is explained in Section 5.3. The layer parameters for FC7 and FC8 can be found in Table 5.2 (layers 12 to 13). The purpose of FC7 and FC8 is to predict the temporal weights of local bit rates (W_{Ti}) in Equation 5.12, which represents the mapping of patch bit rates to P-frame bit rates considering the influence of both the I-frame's texture (BR_i) and the temporal features of the P-frames. Since P-frames' bit rates exhibit a strong correlation with their residue, features extracted from ME maps are employed to determine the weights in Equation 5.12. Thus, the regressor consisting of FC7 and FC8 takes the temporal feature vectors as input and is trained to estimate the temporal weights required in Equation 5.12.

$$R_{PF} = \sum_{N_P} BR_i \times W_{Ti}(QP) \tag{5.12}$$

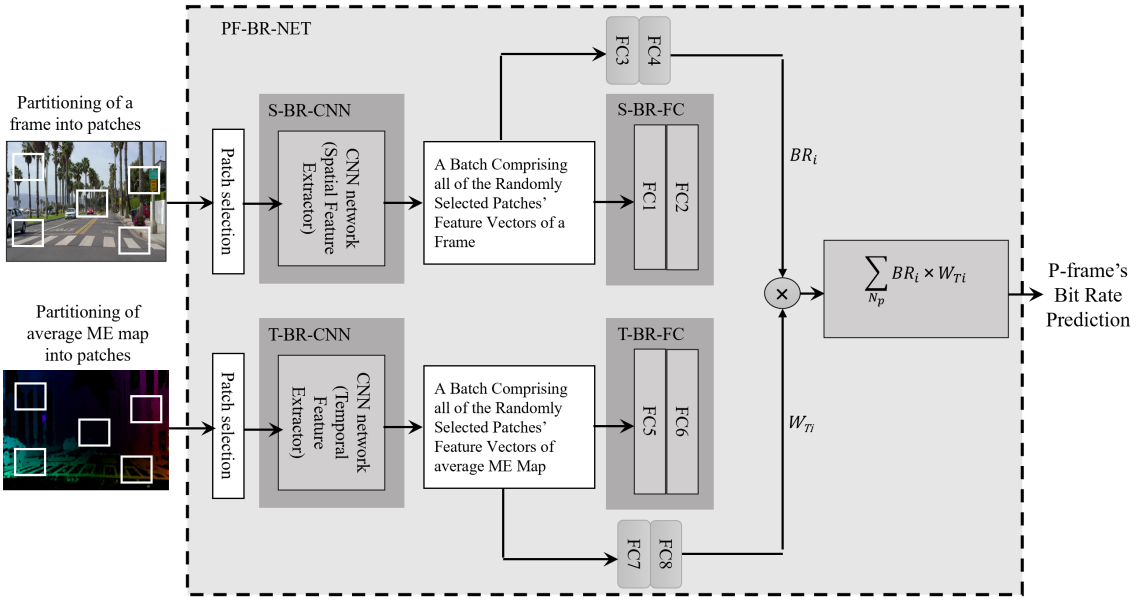

Figure 5.4: Frames' bit rate predictor architecture.

where BR_i is the bit rate of I-frame patches predicted by I-frame's regressor (FC3 and FC4 layers explained in Sections 5.3). W_{Ti}s are the temporal weights.

The loss function for training FC7 and FC8 is characterized in Equation 5.13.

$$J = |BR_{PF} - \sum_{N_P} \tau_{IBR}(\psi_{IBR_{feature}}(P_i, \omega_{IBR}*), \theta^*_{IBR}) \times \tau_{PBR}(\psi_{PBR_{feature}}(ME_i, \omega_{PBR}*), \theta_{PBR})| \tag{5.13}$$

where τ_{PBR} maps a batch of textural feature vectors to an I-frame bit rate. BR_{PF} is the actual P-frame's bit rate, $\psi_{PBR_{feature}}$ maps ME maps to their feature vectors, ω^*_{PBR} is optimized weights of ψ_{PBR} mapping function calculated with loss function of Equation (5.11), θ_{PBR} is the weight of τ_{PBR} mapping function, and N_P is the number of random patches utilized for prediction of P-frame's bit rate.

5.6 Experimental Result

5.6.1 Bit Rate Dataset

The proposed method for predicting I-frame and GOP bit rates has been tested and trained using the dataset introduced and investigated in [33]. The dataset comprises videos with a resolution of 1920×1080 and higher, which have been divided into GOPs at key frames. In total, 200 different GOPs have been utilized for the training, validation, and test datasets. For training purposes, 75% of the GOPs were randomly selected. From the remaining GOPs in the dataset, 15% and 10% were allocated for testing and validation, respectively. Each GOP consists of one I-frame and 15 subsequent P-frames. The bit rates of the encoded frames were provided using HEVC (H.265). The GOPs were encoded at five different quantization parameters (QPs), specifically $QP \in 28, 32, 36, 40, 44$, ensuring the inclusion of a practical dynamic range of bit rates.

5.6.2 Patch-wise I-frame bit-rate prediction accuracy

To train the S-BR-NET network shown in Figure 5.1, the patches of I-frames were labeled with their corresponding bit rates. To obtain these bit rate values, the GOPs were divided into patches of size 128×128, and these patches were encoded using HEVC at the five QPs mentioned in Section 5.6.1. During the encoding process, HEVC was set to the I-frame encoding mode, and the bit rates of the patches were recorded for each QP.

For the validation and testing of S-BR-NET, as illustrated in Figure 5.1, the trained patch-wise I-frame bit rate predictor takes the patches of I-frames as input

and predicts their respective bit rates. The predicted bit rates of the patches are then compared with their actual bit rates. The accuracy of the predicted bit rates was evaluated using the Pearson Linear Correlation Coefficient (PLCC) and the Spearman Coefficient (SROCC). The PLCC measures the linear correlation between the predicted and actual bit rates, while the SROCC evaluates the monotonic relationship between them.

The PLCC and SROCC scores for the patch-wise bit rate prediction of I-frames at different QPs are presented in Table 5.3. The results demonstrate the high accuracy of the patch-wise I-frame bit rate prediction.

Table 5.3: PLCC and SROCC of predicted I-frames' patches' bit rates.

QP	28		32		36		40		44	
	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC
	0.98	0.98	0.98	0.98	0.98	0.97	0.98	0.97	0.97	0.96

5.6.3 Patch-wise P-frame Bit Rate Prediction Accuracy

To train and test the ST-BR-NET network shown in Figure 5.3, the patches of P-frames are labeled with the average bit rate of the corresponding patches within the GOP. To obtain these bit rate labels, the GOPs are partitioned into patches of a specified size, and these patches are encoded using HEVC at the five QPs mentioned in Section 5.6.1. During the encoding process, HEVC is set to the mode that encodes the first frame as an I-frame and the subsequent frames as P-frames (IPPP...). The bit rates of the patches within the P-frames are then recorded for each QP.

Since the bit rates of P-frames can exhibit significant fluctuations, even among successive frames, the average bit rate of patches within each P-frame is calculated to provide a more stable bit rate label.

To test the ST-BR-NET network, as illustrated in Figure 5.3, the network takes pre-encoded patches of the I-frame and the average patches of the motion estimation (ME) map as inputs, and predicts the average bit rate of the patches. The predicted average bit rates are compared with the actual average bit rates of the patches, and the Pearson Linear Correlation Coefficient (PLCC) and Spearman Coefficient (SROCC) are calculated to evaluate the accuracy of the predictions for the test samples at different QPs.

Table 5.4 presents the PLCC and SROCC scores for the patch-wise GOP bit rate prediction. It can be observed that the accuracy of patch-level P-frame bit rate prediction is lower compared to I-frame bit rate prediction. This is reasonable since P-frame production is more complex, involving both spatial and temporal features.

Table 5.4: PLCC and SROCC of predicted P-frames' patches bit rate.

QP	28		32		36		40		44	
	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC
	0.77	0.67	0.71	0.64	0.64	0.6	0.63	0.62	0.6	0.61

5.6.4 Frame-wise I-frame Bit Rate Prediction Accuracy

To predict the bit rates of I-frames and train the IF-BR-NET network, as depicted in Figure 5.2, we randomly selected 10% of the patches extracted from each I-frame. These selected patches were then used as input for the trained IF-BR-NET. The mean absolute error (MAE) of the I-frame bit rate prediction at different QPs for the test video samples is presented in Table 5.5. The results in Table 5.5 demonstrate an average relative error of under 0.2 for most of the I-frame bit rates. This high accuracy in bit rate prediction contributes to precise QP control.

By considering the Q-R model expressed in Equation 5.3 and utilizing the actual

bit rates of I-frames at the assigned QPs, we are able to determine the optimized model parameters for Equation 5.3. The optimization problem of the Q-R model, which establishes the relationship between QP and I-frames' bit rate, is defined in Equation 5.14.

$$[\alpha^*_{IF}, \beta^*_{IF}] = \arg \min_{\alpha_{IF}, \beta_{IF}} \sum_{i=0}^{N_{QP}} |BR_{IFi} - \alpha_{IF} \times 2^{(\beta_{IF} \times QP_i)}| \tag{5.14}$$

where α^*_{IF} and β^*_{IF} are optimized Q-R parameters for I-frame bit rate prediction, N_{QP} is the number of assigned QPs, and BR_{IFi} is the bit rate of an I-frame sample at the i'th QP. The predicted QP is computed by substituting the predicted bit rate of the I-frames into the optimized Q-R model, as defined in Equation 5.15.

$$QP_{pred} = \frac{log(\frac{BR_{pred}}{\alpha^*_{IF}})}{\beta^*_{IF}}, \tag{5.15}$$

where QP_{pred} and BR_{pred} represent the predicted QP and bit rate, respectively. Subsequently, the predicted QP is compared to the assigned actual QP, and their mean absolute error (MAE) is calculated. The MAE is determined at each QP, where $QP \in 28, 32, 36, 40, 44$, for the test videos. The resulting MAE values are presented in Table 5.6. Based on the MAE observations, it can be concluded that the I-frame's bit rate predictor exhibits reliable bit rate prediction, and the selected QP effectively achieves the assigned bit rate.

5.6.5 Frame-wise P-frame Bit Rate Prediction Accuracy

As mentioned in the preceding sections of this chapter, the prediction of P-frame bit rates is conducted at the GOP level. Thus, the proposed P-frame bit rate predictor,

PF-BR-NET, takes a batch of I-frame patches and a batch of average ME map patches as input and generates the predicted average bit rate for the P-frames within a GOP as output. To reduce computational costs, only 10% of the patches from each frame are randomly selected for training and testing the network. Following the prediction of the average P-frame bit rate by the PF-BR-NET network, the results are compared against the actual average values, and the mean absolute error (MAE) is calculated. The relative error of the average P-frame bit rate prediction is presented in Table 5.7. Notably, the accuracy of average bit rate prediction for P-frames (GOP) in videos with dynamic textures (such as river, sea, and forest) and faster motion is relatively lower. Enhancing the prediction accuracy of GOP bit rates for videos with dynamic textures and fast motions can be achieved by employing a larger number of training samples specifically targeting such scenarios. By utilizing the Q-R model described in equation 5.3 and incorporating the actual bit rates of P-frames at their respective assigned QPs, the model parameters for equation 5.3 are determined. The optimization problem for the Q-R model can be formulated as follows:

$$[\alpha^*_{PF}, \beta^*_{PF}] = \arg \min_{\alpha_{PF}, \beta_{PF}} \sum_{i=0}^{N_{QP}} |BR_{PFi} - \alpha_{PF} \times 2^{(\beta_{PF} \times QP_i)}| \tag{5.16}$$

where α^*_{PF} and β^*_{PF} are optimized Q-R parameters for P-frame bit rate prediction, N_{QP} is the number of assigned QPs, and BR_{PFi} is the average bit rate of a GOP's P-frames at the i'th QP. By substituting the predicted average bit rates of P-frames into the optimized Q-R model, as expressed in equation 5.17, the corresponding predicted

QPs can be estimated.

$$QP_{pred} = \frac{log(\frac{BR_{pred}}{\alpha^*_{PF}})}{\beta^*_{PF}} \tag{5.17}$$

where QP_{pred} and BR_{pred} represent the predicted QP and bit rate, respectively. The predicted QP is then compared with the actual assigned QP, and the mean absolute error (MAE) between them is calculated. Table 5.8 presents the MAE of P-frame QP prediction. The table demonstrates that the P-frame bit rate predictor exhibits reliable performance for the majority of test sample tracks. However, for videos with continuous dynamic textures and movements, such as "water," the QP prediction displays a higher MAE and lower accuracy. This outcome is expected due to the more complex spatial and temporal features present in these videos.

Table 5.5: Relative error of I-frames' bit rate prediction and curve fitting.

QP	28	32	36	40	44
1. Aspen	0.18	0.14	0.11	0.13	0.15
2. Blue sky	0.01	0.07	0.11	0.11	0.08
3. Controlled burn	0.08	0.03	0.09	0.09	0.12
4. Ducks take off	0.29	0.24	0.24	0.3	0.35
5. In to the tree	0.19	0.3	0.35	0.39	0.47
6. Old town cross	0.28	0.1	0.07	0.12	0.12
7. Park joy	0.1	0.05	0.08	0.04	0.03
8. Pedestrian area	0.05	0.08	0.05	0.04	0.03
9. Red kayak	0.09	0.13	0.17	0.13	0.1
10. Rush field cuts	0.31	0.05	0.07	0.08	0.12
11. Tractor	0.19	0.14	0.14	0.08	0.1
12. Ritual dance	0.16	0.17	0.2	0.23	0.23
13. Touchdown-pass	0.16	0.04	0.08	0.06	0.08
14. Driving-POV	0.0	0.06	0.08	0.03	0.04
15. Pier Seaside	0.23	0.16	0.23	0.17	0.08
16. Cross Walk	0.39	0.51	0.57	0.39	0.2
17. Square and Timelapse	0.51	0.66	0.63	0.44	0.21

Table 5.6: Average absolute error of I-frame QP prediction.

QP	28	32	36	40	44
1. Aspen	0.5	0.9	1.0	1.3	1.4
2. Blue sky	0.0	0.5	1.0	1.5	1.5
3. Controlled burn	0.0	0.0	0.33	1.0	1.33
4. Ducks take off	1.5	3.5	3.5	4.0	3.5
5. In to the tree	2.0	3.0	4.0	4.0	3.0
6. Old town cross	1.0	1.0	1.0	1.0	1.0
7. Park joy	0.0	1.75	2.0	1.75	1.5
8. Pedestrian area	0.0	1.0	0.0	0.0	0.0
9. Red kayak	0.0	1.33	1.33	1.33	1.33
10. Rush field cuts	1.0	1.0	0.0	1.0	1.0
11. Tractor	0.33	1.0	1.33	1.33	0.67
12. Ritual dance	1.5	1.5	2.0	2.0	2.0
13. Touchdown-pass	1.0	0.5	0.0	0.5	0.5
14. Driving-POV	0.0	0.0	0.0	0.0	0.0
15. Pier Seaside	2.0	2.0	1.0	1.0	1.0
16. Cross Walk	4.0	3.0	3.0	3.0	2.0
17. Square and Timelapse	5.0	4.5	4.0	3.0	2.5

Table 5.7: Relative error of GOPs' bitrate prediction.

QP	28	32	36	40	44
1. Aspen	0.26	0.3	0.27	0.32	0.51
2. Blue sky	0.38	0.46	0.46	0.24	0.07
3. Controlled burn	0.46	0.59	0.6	0.43	0.37
4. ducks take off	0.71	0.73	0.76	0.78	0.78
5. In to the tree	0.23	0.17	0.39	0.74	1.16
6. Old town cross	0.27	0.21	0.47	0.31	0.04
7. park joy	0.73	0.71	0.68	0.61	0.45
8. Pedestrian area	0.4	0.47	0.52	0.55	0.58
9. Red kayak	0.64	0.75	0.82	0.86	0.87
10. Rush field cuts	0.19	0.31	0.45	0.58	0.66
11. Tractor	0.34	0.38	0.33	0.32	0.46
12. Ritual dance	0.53	0.67	0.78	0.85	0.89
13. Touchdown-pass	0.28	0.28	0.38	0.48	0.55
14. Driving-POV	0.03	0.15	0.27	0.43	0.54
15. Pier Seaside	0.14	0.7	0.83	0.54	0.07
16. Cross Walk	0.4	0.65	0.81	0.9	0.95
17. Square and Timelapse	0.22	0.28	0.37	0.47	0.53

Table 5.8: Average absolute error of GOPs' QP prediction.

QP	28	32	36	40	44
1. Aspen	0.2	1.6	1.4	1.6	1.8
2. Blue sky	2.0	3.5	1.5	0.5	2.0
3. Controlled burn	0.67	1.67	2.33	3.33	4.0
4. Ducks take off	0.0	4.0	7.5	8.5	9.5
5. In to the tree	0.0	0.0	1.0	2.0	3.0
6. Old town cross	0.0	0.0	1.0	1.0	2.0
7. Park joy	0.0	4.0	8.0	9.5	3.5
8. Pedestrian area	11.0	8.0	6.0	5.0	4.0
9. Red kayak	0.0	4.0	7.67	10.67	12.67
10. Rush field cuts	0.0	2.0	3.0	4.0	5.0
11. Tractor	1.33	1.67	1.67	2.67	4.0
12 Ritual dance	0.0	4.0	7.0	9.0	11.0
13. Touchdown-pass	0.0	1.5	2.5	4.5	5.5
14. Driving-POV	0.0	1.0	2.0	2.0	3.0
15. Pier Seaside	2.0	2.0	2.0	2.0	2.0
16. Cross Walk	0.0	3.0	5.0	8.0	11.0
17. Square and Timelapse	0.0	3.0	5.0	7.5	2.5

Chapter 6

Deep CNN-Based Method for Spatially and Temporally Scaled Encoding

With the growing demand for high-resolution and high frame-rate content, efficient compression techniques are necessary to meet bitrate constraints during video storage and delivery. Predicting the bitrate and distortion of video frames prior to encoding is crucial to avoid frame skipping or exceeding available bandwidth. High-resolution and high frame-rate videos often exhibit a significant amount of spatial and temporal redundancy. To address this, spatial downscaling or temporal down-conversion can be employed to reduce redundancy before encoding. Subsequently, upscaling can be performed after decoding return the video to its native resolution and/or frame rate.

In Chapter 3, we proposed an I-frame adaptive scaled encoding method that utilizes hand-crafted features. This chapter introduces a deep learning approach based on the framework presented in Chapters 4 and 5. The goal is to predict the optimized

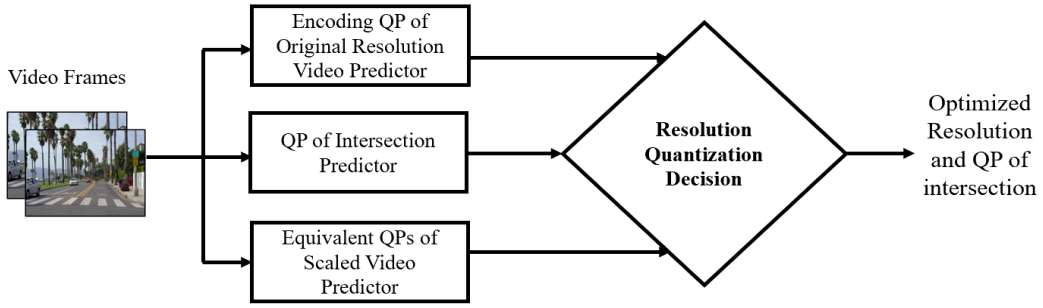

Figure 6.1: Overview of proposed spatially adaptive encoder.

resolution for video encoding and determine the intersection QP between the original and scaled resolution I-frames and P-frames. We present a CNN-based method to predict the intersection QP by considering the rate-distortion characteristics of the original and down-converted videos. To ensure comprehensive analysis, we compare the accuracy of both hand-crafted and deep learning methods.

6.1 Deep CNN QP prediction of spatially scaled video

Figure 6.1 illustrates the framework of the spatially adaptive encoder (SA-ENC), which aims to optimize the parameters of spatial resampling prior to video encoding in order to enhance rate-distortion performance. The SA-ENC predicts the quantization parameter (QP) of the intersection point between the rate-distortion curves of the original and scaled videos. If the predicted QP of the intersection is lower than the QP of the native resolution video, the encoder proceeds to encode the scaled video at the scaled resolution. The method for selecting the QP to encode the videos within the allocated bit rate was detailed in Chapter 4.

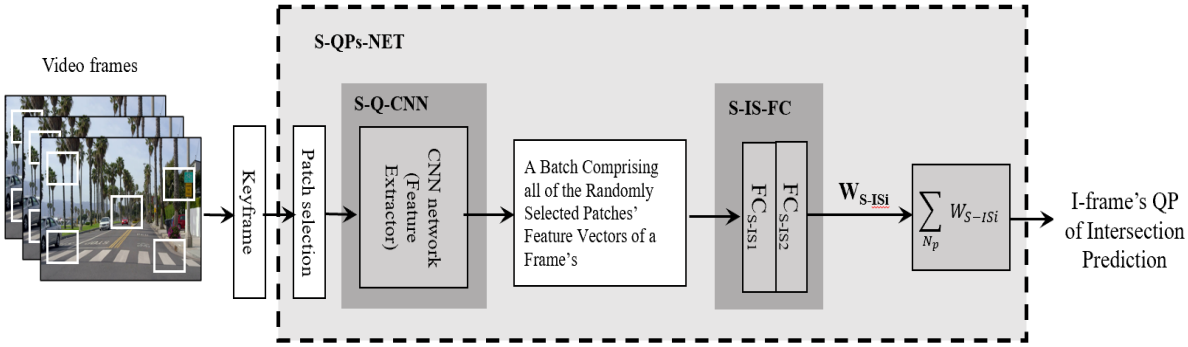

Figure 6.2: CNN-based QP of intersection predictor network for spatially adaptive encoding.

In a video GOP (Group of Pictures), an I-frame is followed by a series of P-frames, which may exhibit significant variations in bit rates and quality. Consequently, the QP of the intersection point for the I-frame and P-frames within a GOP could differ. Therefore, we conducted a comprehensive study and developed separate training methods for the proposed SA-ENC technique applied to I-frames and P-frames. The proposed method employs an innovative patch-based deep CNN-based approach for encoding resolution determination. The subsequent sections provide a detailed description of the proposed method for predicting the QP of the intersection points for both I-frames and P-frames.

6.1.1 Deep CNN QP prediction of spatially scaled I-frame

Figure 6.2 illustrates the deep CNN network designed to predict the QP of the intersection point for scaled I-frame encoding. The adaptive spatially scaled encoding network, denoted as S-QPs-NET, consists of the trained feature extractor CNN network (S-Q-CNN) introduced in Chapter 3, followed by two regressors and two pooling layers. The S-Q-CNN serves as the CNN-based feature extractor, trained specifically

for patch-based perceptual quality prediction.

In the S-QPs-NET network depicted in Figure 6.2, I-frame patches are inputted, and the trained S-Q-CNN network extracts the corresponding feature vectors. These feature vectors are then mapped to the QP of the intersection by the S-IS-FC regressor and the subsequent pooling layer, as shown in Figure 6.2.

The weights of S-IS-FC are optimized to align with the target QP of the intersection. The parameters of the frame QP prediction network, or S-QPs-NET network, are detailed in Table 6.1, where Layers 1 to 15 represent the layers of S-Q-CNN, and Layers 16 to 17 correspond to the parameters of the S-IS-FC fully connected layers.

The cost function utilized to train the fully connected layers, S-IS-FC, is presented in Equation 6.1.

$$J_{SIS} = |IS_S - \sum_{N_P} \tau_{SIS}(\psi_{IQ_{feature}}(P_i, \omega_{IQ}*), \theta_{SIS})| \tag{6.1}$$

where the loss function is denoted as J_{SIS}, and the mapping function of the S-IS-FC regressor is represented by τ_{SIS}. Specifically, τ_{SIS} maps a batch of textural feature vectors to spatial IS-QP weights W_{S-ISi} as depicted in Figure 6.2. Here, IS_S refers to the actual spatial IS-QP.

To extract the feature vectors from I-frame patches, we employ the mapping function $\psi_{IQ_{feature}}$, which was introduced and trained in Chapter 4. The optimized weights of the ψ_{IQ} mapping function are denoted as ω^*_{IQ}. Furthermore, θ_{SIS} represents the weight of the τ_{SIS} mapping function. It is important to note that N_P corresponds to the number of random patches used for predicting the GOP's spatial IS-QP.

Table 6.1: Layers of deep learning CNN network of pre-encoded adaptive spatial scaling encoder predictor.

#	Type	Kernel	Stride	Activation	Outputs
01	Conv.	3×3	1×1	ReLU	32
02					
03	Pool.	-	2×2	-	32
04	Conv.	3×3	1×1	ReLU	32
05					
06	Pool.	-	2×2	-	32
07	Conv.	3×3	1×1	ReLU	64
08					
09	Pool.	-	2×2	-	64
10	Conv.	3×3	1×1	ReLU	64
11					
12	Pool.	-	2×2	-	64
13	Conv.	3×3	1×1	ReLU	128
14					
15	Pool.	-	2×2	-	128
16					120
17	FC			ReLU	5

6.1.2 Deep CNN QP prediction of spatially scaled P-frame

A GOP may comprise hundreds of P-frames; therefore, predicting and optimizing P-frames parameters is important for an optimized adaptive encoding. P-frames are built based on their disparities with the keyframes; thus, the encoding parameters can be correlated by spatial or temporal features of P-frames. But as in spatially adaptive encoding, we are scaling frames resolution and decreasing spatial redundancy, which may cause loss of sharpness and cause blurriness artifacts at the decoded P-frames, so textural features of frames effects scaled resolution P-frames and consequently the QP of intersection. Consequently, we utilized S-QPs-NET network shown in figure 6.2 for predicting IS-QP. We have used S-Q-CNN network shown in Figure 6.2 for extracting spatial features of I-frames. For predicting IS-QP, we trained the S-IS-FC

regressor to predict IS-QP. The loss function for used for training S-IS-FC is stated in Equation 6.1, and the training procedure is the same as training of the network for prediction IS-QP of I-frames.

Table 6.2: Layers of deep learning CNN network of temporal feature extractor and temporal weight predictor.

#	Type	Kernel	Stride	Activation	Outputs
01	Conv.	3×3	1×1	ReLU	32
02					
03	Pool.	-	2×2	-	32
04	Conv.	3×3	1×1	ReLU	32
05					
06	Pool.	-	2×2	-	32
07	Conv.	3×3	1×1	ReLU	64
08					
09	Pool.	-	2×2	-	64
10					128
11	FC			ReLU	5

6.2 Deep CNN QP prediction of Temporally scaled P-frame

In the preceding section, we introduced a CNN-based method to reduce spatial resolution before encoding. However, as the demand for a more realistic visual experience grows, there is a simultaneous increase in frame rate, resulting in an escalation of temporal redundancy. To address this, down-converting the video frame rate prior to encoding and subsequently up-converting it after decoding can enhance the rate-distortion curve, reducing the encoding bit rate while maintaining video quality. Nonetheless, the temporal redundancy of a video is heavily dependent on its content. A video featuring a sports scene, for instance, may exhibit high-speed motions and

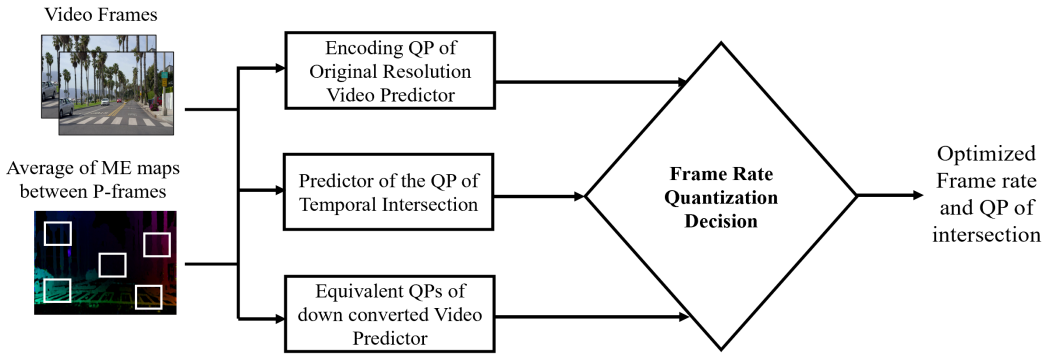

Figure 6.3: Overview of proposed temporally adaptive encoder.

fewer redundant frames compared to a weather forecast video. Hence, we delve into the study of temporally adaptive video encoding in this section, where we propose a patch-based deep CNN method to optimize the video frame rate for encoding based on its temporal features.

Figure 6.3 illustrates the comprehensive pipeline of temporally scaled adaptive encoding (TSA-ENC). It demonstrates how TSA-ENC takes I-frames and motion estimation (ME) maps to extract both spatial and temporal features of a GOP. Subsequently, TSA-ENC predicts the temporal IS-QP. To determine the appropriate QP for the allocated bit rate, we leverage the method proposed in Chapter 5.

In adaptive temporal encoding, considering that a GOP commences with an I-frame and is followed by P-frames, only the P-frames are subject to temporal sampling. To achieve this, we remove every other P-frame from the videos, encode the down-sampled video, decode it, and finally temporally upscale it to its original frame rate. Similar to spatial scaling adaptive encoding, the rate-distortion curves of the temporally scaled video and the original video may intersect at a specific QP value, which we denote as the temporal QP of intersection (T-IS-QP). This implies that at a bit rate lower than the intersection bit rate, the temporally downscaled video

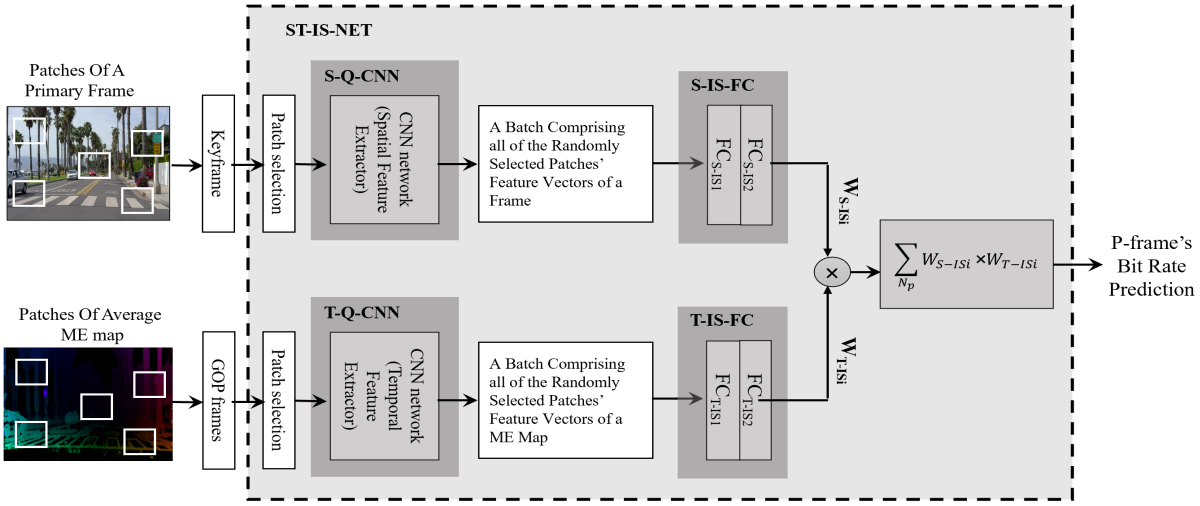

Figure 6.4: Overview of proposed temporally adaptive encoder.

exhibits higher quality than the original video at the same bit rate. Consequently, accurate prediction of the T-IS-QP is crucial for maximizing encoding performance.

Figure 6.4 illustrates the proposed patch-based deep CNN networks used to predict IS-QP-T for the adaptive temporal encoding. The ST-IS-NET, depicted in figure 6.4, consists of two streams: the spatial network (top stream) and the temporal network (bottom stream). These networks comprise the trained S-Q-CNN and T-Q-CNN, respectively, which perform patch-based spatial and temporal feature extraction. The training procedure for S-Q-CNN and T-Q-CNN was explained in Chapter 4, and we utilize the extracted spatial and temporal feature vectors.

The IS-QP-T represents the QP at which the original and temporally downscaled GOP exhibit the same bit rate and quality. Consequently, the IS-QP-T is correlated with the quality of the original and temporally scaled GOP. The temporally scaled GOP consists of two types of frames in terms of distortion: the P-frames with encoding distortion and the reconstructed P-frames with interpolation distortion. The

distortion of the reconstructed frames is influenced by the similarity between adjacent frames. If the disparity between adjacent frames increases, the interpolation distortion may also increase. Hence, the proposed method leverages the textural features of the I-frame and the average ME maps, generated by comparing removed frames and their adjacent P-frame, to predict the temporal QP of intersection, IS-QP-T.

The layers of the spatial and temporal streams are presented in Table 6.1 and 6.2, respectively. In Table 6.1, Layers 1 to 15 belong to the S-Q-CNN network, while Layers 16 to 17 represent the S-IS-FC regressor layers. In Table 6.2, Layers 1 to 9 correspond to the T-Q-CNN network layers, and Layers 10 to 11 denote the parameters of the S-IS-FC regressor layer. The regressors shown in Figure 6.4 consist of two fully connected layers. The S-IS-FC and T-IS-FC regressors are trained iteratively.

The predicted IS-QP-pred in Equation 6.2 corresponds to the actual temporal IS-QP. W_{S-ISi} and W_{T-ISi} represent the spatial and temporal weights of the i-th patch, respectively, used to generate the predicted QP of intersection.

$$T\text{–}IS\text{–}QP\text{–}pred = \sum_{N_P} W_{S-ISi} \times W_{T-ISi} \tag{6.2}$$

The outputs W_{S-ISi} and W_{T-ISi} correspond to the outputs of the S-IS-FC and T-IS-FC regressors, respectively. T-IS-QP-pred represents the predicted temporal QP of intersection. The value of N_P indicates the number of randomly selected patches used to predict the P-frame T-IS-QP. The S-IS-FC and T-IS-FC regressors take CNN-based extracted features as input to predict the spatial and temporal weights of T-IS-QP-pred. These regressors are trained alternately. The training procedure for the S-IS-FC and T-IS-FC regressors is outlined in Algorithm 1. The cost functions for training the S-IS-FC and T-IS-FC regressors are presented in Equations 6.3 and 6.4, respectively.

Algorithm 1 Algorithm for training CNN-based temporally adaptive encoder.

Initialization **:** Epoch number (Epch-num)
number of samples (Smpl-num)
number of patches, N_P
$k1 \leftarrow 0$
$k2 \leftarrow 0$
Data **:** I-frame patch extraction.
Average ME map patch extraction.
while $k1 \leq Epch\text{–}num$ **do**
  while $k2 \leq Smpl\text{–}num$ **do**
    Extract I-frame patches, spatial features, by S-Q-CNN
    Extract ME map patches, temporal features, by T-Q-CNN
    $W_S \leftarrow$ Pass spatial features through S-IS-FC
    $W_T \leftarrow$ Pass temporal features through T-IS-FC
    $Predicted\ IS\text{–}QP \leftarrow \sum_{N_P}(W_S * W_T)$
    if k2 is odd **then**
      Minimize J_{SIS} cost function.
    else
      Minimize J_{TIS} cost function.
    end if
    $k_2 \leftarrow k_2 + 1$
  end while
  $k_1 \leftarrow k_1 + 1$
end while

$$J_{TIS} = \arg\min_{\theta_{TIS}} |IS_T - \sum_{N_P} \tau_{SIS}(\psi_{IQ_{feature}}(P_i, \omega_{IQ}*), \theta^*_{SIS}) \times \tau_{TIS}(\psi_{PQ_{feature}}(ME_i, \omega_{PQ}*), \theta_{TIS})| \tag{6.3}$$

$$J_{SIS} = \arg\min_{\theta_{TIS}} |IS_T - \sum_{N_P} \tau_{SIS}(\psi_{IQ_{feature}}(P_i, \omega_{IQ}*), \theta_{SIS}) \times \tau_{TIS}(\psi_{PQ_{feature}}(ME_i, \omega_{PQ}*), \theta^*_{TIS})| \tag{6.4}$$

where τ_{SIS} represents the mapping function of the S-IS-FC regressor, which maps a batch of textural feature vectors to spatial IS-QP weights, W_{S--ISi}. Similarly, τ_{TIS} denotes the mapping function of the T-IS-FC regressor, which maps a batch of motion estimation (ME) map feature vectors to temporal IS-QP weights, W_{T-ISi}. The variable IS_T represents the actual temporal IS-QP. Additionally, $\psi_{PQ_{feature}}$ is the mapping function that maps ME maps to their corresponding feature vectors, and $\psi_{IQ_{feature}}$ maps I-frame patches to their feature vectors. These mapping functions, $\psi_{PQ_{feature}}$ and $\psi_{IQ_{feature}}$, were introduced and trained in Chapter 4. The optimized weights of the ψ_{PQ} and ψ_{IQ} mapping functions are denoted by ω^*_{PQ} and ω^*_{IQ}, respectively. Furthermore, θ_{SIS} and θ_{TIS} represent the weights of the τ_{SIS} and τ_{TIS} mapping functions, respectively. The variable N_P signifies the number of random patches utilized for the prediction of the GOP's temporal IS-QP.

6.3 Experimental Results

The video dataset used for training and testing was introduced in Chapters 4 and 5. It consisted of GOPs divided into three subsets: training, validation, and testing. A total of 300 GOPs were available for these subsets, with the training set comprising 75 percent, the validation set comprising 10 percent, and the testing set comprising 15 percent of the GOPs. Each GOP followed a structure of one keyframe followed by 15 P-frames (IPP...P). To facilitate training, each frame was partitioned into patches of size 64×64. Randomly selecting 10 percent of the patches 100 times resulted in a training video set containing 30,000 GOP-level samples. The accuracy of the prediction methods was evaluated by comparing two schemes: the proposed CNN-based methods presented in this chapter and an ML-based method that utilized hand-crafted spatial or spatio-temporal features.

6.3.1 QP of Intersection Prediction For Spatial scaling

I-frame's QP of Intersection Prediction

To predict IF-IS-QP, we utilize the trained network shown in Figure 6.4. The S-QPs-NET network takes the I-frames of the videos, randomly selects 10 percent of the patches from the I-frames, and predicts the IF-IS-QP. We then compare the predicted IF-IS-QP to the actual extracted IF-IS-QP of the test videos and calculate their mean absolute error (MAE) to measure the accuracy of the CNN method in predicting IF-IS-QP.

In order to evaluate the performance of the CNN method, we compare the accuracy of the predicted IF-IS-QPs using the deep CNN S-QPs-NET with an ML-based

method that utilizes hand-crafted features. The machine learning method was introduced in Chapter 3, where we demonstrated that the Sobel filter of the I-frame and the PSNR of the scaled-resolution I-frame are the most effective hand-crafted spatial features for predicting QP of intersection.

Table 6.3 presents the accuracy of IF-IS-QP prediction using these hand-crafted features and the machine learning method (ML-M), as well as the proposed CNN method (Prop-M). The magnitude of IS-QP is correlated with both the quality and bit rate of the original and scaled-resolution videos, as it represents the bit rate at which both videos have the same quality. The table demonstrates the superior or comparable performance of the proposed CNN network compared to the ML-based method using hand-crafted features.

P-frame's QP of Intersection Prediction

To predict the QP of intersection for P-frames (PF-IS-QP), we trained the S-QPs-NET scheme, which takes I-frames as input and predicts the IS-QPs for P-frames. Table 6.4 presents the mean absolute error of IS-QP prediction using the CNN method (Prop-M) and the hand-crafted ML-based method (ML-M) introduced in Chapter 3. The table indicates that Prop-M and ML-M exhibit similar prediction accuracy for P-frame prediction.

Figure 6.5 showcases some of the test sample results for spatial scaling of I-frames. The figure displays the original and scaled resolution rate-distortion curves of the encoded I-frame. The red circle represents the predicted QP of intersection by the ML-based method, while the blue square represents the prediction made by the proposed method.

6.3.2 Temporally scaling QP of Intersection Prediction

To predict the temporal QP of intersection, we utilized the ST-IS-NET network proposed in Figure 6.4. ST-IS-NET takes the I-frame and the average of ME maps, which are generated by comparing the removed frame and its adjacent frame, as input to predict the temporal QP of intersection. Additionally, we employed an ML-based prediction method that utilizes hand-crafted features. Zhang et al. [95] utilized the Haar wavelet for video quality assessment prediction with different frame rates, and Afonso et al. [6] employed this feature for temporal adaptive encoding.

Table 6.5 presents the accuracy of temporal intersection prediction measured by the mean absolute error (MAE) for both the ST-IS-NET network (Prop-M) and the ML-based method (ML-M). The table demonstrates the superior performance of the CNN-based method in predicting the QP of temporal intersection.

Figure 6.6 showcases some of the test sample results for temporal scaling of P-frames. The figure displays the rate-distortion curves of the original GOPs and the down-converted GOPs. The red circle represents the predicted QP of intersection by the ML-based method, while the blue square represents the prediction made by the proposed method.

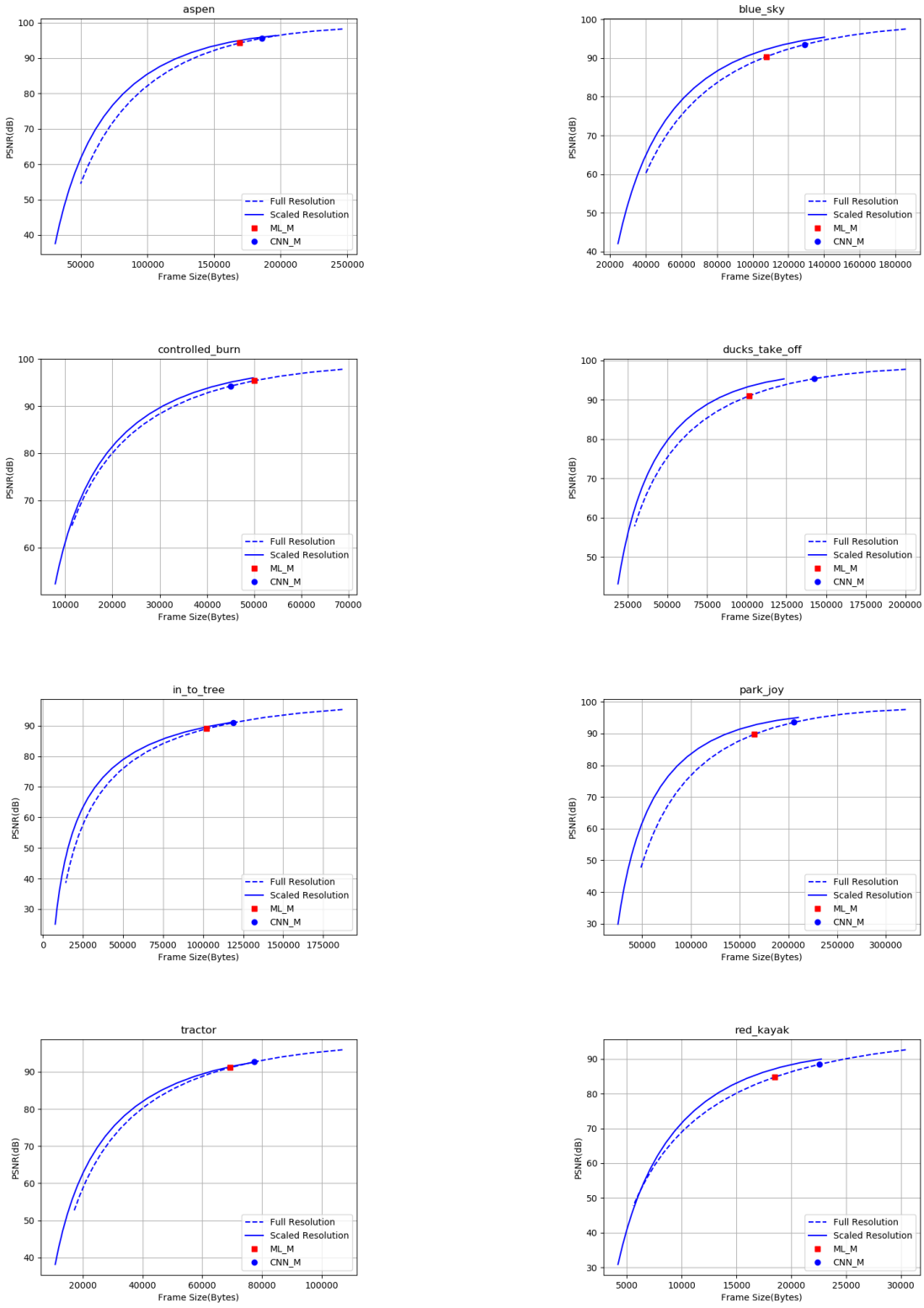

Figure 6.5: IS-QP prediction performance of 8 tested sequences for I-frames encoded at two resolution. Each data point represents the average value of all frames for a given QP.

Table 6.3: Mean absolute error of spatial IS-QP prediction for I-frames.

QP	ML-M	Prop-M
1. Aspen	1.4	0.8
2. Blue sky	1.3	0.5
3. Controlled burn	1.13	1.33
4. ducks take off	1.3	1.0
5. In to the tree	0.6	1.0
6. Old town cross	0.8	1.0
7. park joy	1.1	0.5
8. Pedestrian area	1.2	1.5
9. Red kayak	1.53	1.0
10. Rush field cuts	0.2	1.0
11. Tractor	0.67	0.0
12. Ritual dance	1.3	0.5
13. Touchdown-pass	1.4	1.0
14. Driving-POV	0.8	1.0
15. Pier Seaside	1.4	2.0
16. Cross Walk	3.0	4.0
17. Square and Timelapse	1.6	2.5

Table 6.4: Mean absolute error of spatial IS-QP prediction for P-frames.

QP	ML-M	Prep-M
1. Aspen	1.2	0.51
2. Blue sky	0.1	0.61
3. Controlled burn	0.6	1.44
4. ducks take off	0.5	1.44
5. In to the tree	0.6	0.79
6. Old town cross	2.0	0.54
7. park joy	0.7	0.26
8. Pedestrian area	0.8	0.42
9. Red kayak	2.07	0.84
10. Rush field cuts	0.2	0.0
11. Tractor	0.8	0.76
12. Ritual dance	1.5	0.62
13. Touchdown-pass	0.1	1.06
14. Driving-POV	1.2	0.74
15. Pier Seaside	5.6	2.64
16. Cross Walk	1.0	2.27
17. Square and Timelapse	4.3	3.42

Table 6.5: Mean absolute error of temporally IS-QP prediction.

QP	ML-M	Prep-M
1. Aspen	6.9	4.0
2. Blue sky	0.9	4.9
3. Controlled burn	16.49	8.23
4. ducks take off	0.52	1.04
5. In to the tree	21.53	15.56
6. Old town cross	0.15	6.51
7. park joy	0.84	1.3
8. Pedestrian area	5.3	2.68
9. Red kayak	4.81	2.09
10. Rush field cuts	10.87	4.98
11. Tractor	6.98	3.03
12. Ritual dance	3.24	1.12
13. Touchdown-pass	10.54	4.01
14. Driving-POV	3.34	6.65
15. Pier Seaside	8.37	2.31
16. Cross Walk	11.27	3.32
17. Square and Timelapse	8.11	1.01

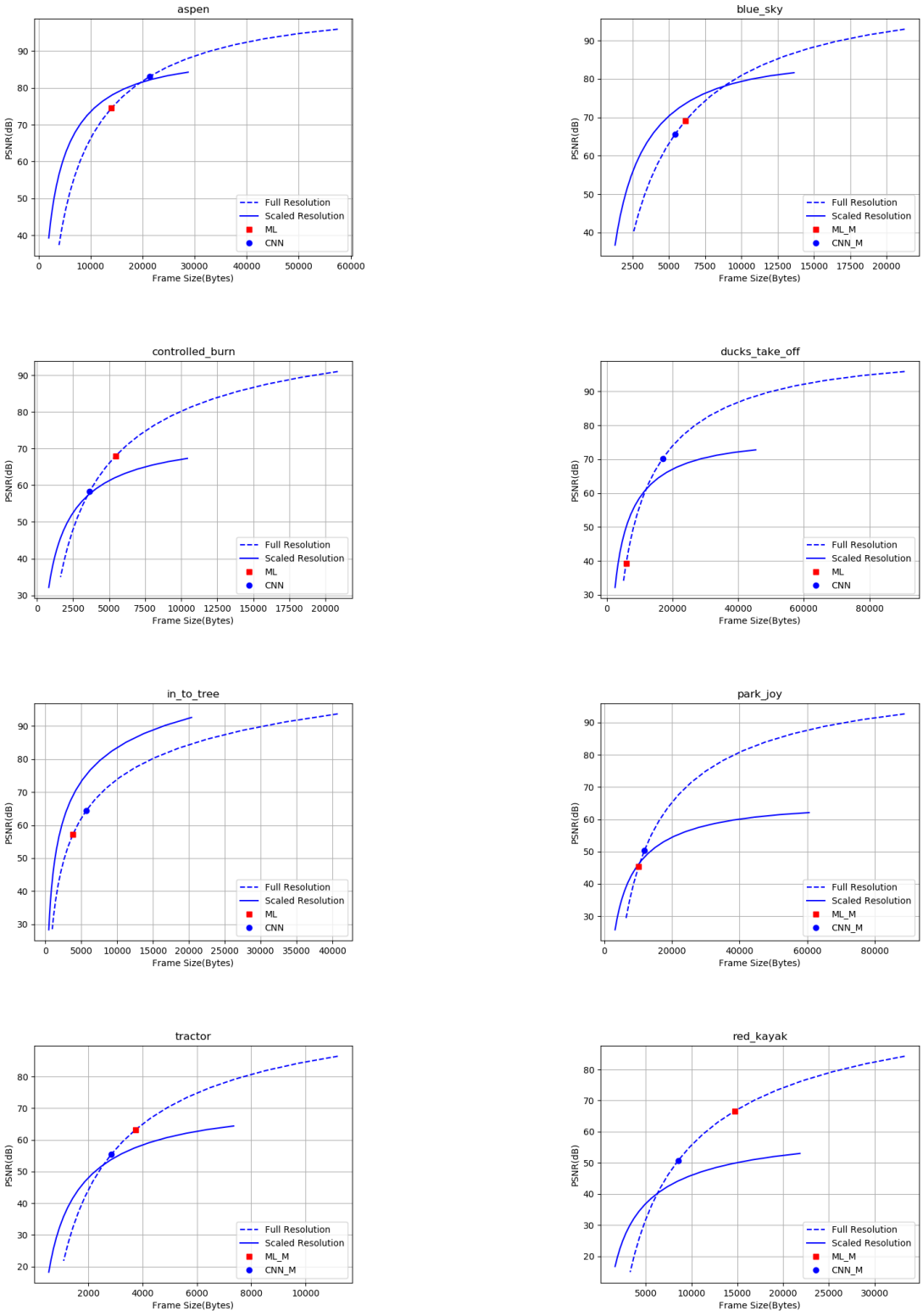

Figure 6.6: IS-QP prediction performance of 8 tested sequences for GOPs encoded at two frame-rate. Each data point represents the average value of all frames for a given QP.

Chapter 7

Conclusions and Future Work

This thesis focuses on the optimization of video encoder parameters to predict and maximize the rate-distortion performance. Distortion and bit rate serve as efficiency indicators for compression algorithms, and their control is achieved through the quantization parameter (QP). Additionally, frame resolution and video frame rate are video characteristics that significantly influence encoding performance. To provide a comprehensive understanding of the research problem, the thesis begins with a review of previous studies on distortion assessment, rate control, and adaptive resolution and frame rate encoding. Image quality assessment (IQA) plays a crucial role in measuring and predicting the perceived quality of processed images; hence, a thorough examination of previous methods in IQA was conducted. Rate control (RC) is closely associated with rate-distortion models employed within the compression algorithm. RC models allocate appropriate bit rates to CTUs/macroblocks, frames, and Group of Pictures (GOPs) to minimize distortion while maintaining consistent perceived visual quality across frames to avoid flickering.

The contributions of this thesis are presented in Chapters 3, 4, 5, and 6. In Chapter 3, we conducted a comprehensive study on the impact of spatially downscaling frames before encoding and subsequently upscaling them after decoding. To validate the benefits of frame rescaling we compared the rate-distortion curves of the original resolution frames with those of the rescaled frames. Furthermore, we extracted carefully engineered textural features from the frames and employed a machine learning model to characterize the bit rate intersection. In addition to textural features, we also examined features related to motion estimation statistics and inter-frame statistics. We conducted separate investigations on the rate-distortion curve intersections for Intra-frame and inter-frame scenarios. The selection of the best textural and non-textural features was based on the PCC and SRCC indexes, ensuring robust feature representation.

In Chapter 4, we introduced a novel deep CNN-based perceived quality predictor for HEVC. Predicting compression quality before encoding poses challenges similar to those encountered in No-reference Image Quality Assessment (IQA), as it involves evaluating degradation without access to both raw and distorted images. Another hurdle in developing the proposed encoder quality predictor is the requirement of a diverse dataset to train the deep CNN and mitigate overfitting. To address this issue, we utilized Video Multimethod Assessment Fusion (VMAF) as the perceived quality index. In contrast to previous IQA methods that labeled patches based on frame scores, we took into consideration the content differences between patches and labeled them based on their extracted VMAF scores. Since quality prediction prior to encoding primarily focuses on the patch level, we presented and employed three methods to transform patch-level predictions into frame-level predictions. A novel

patch-to-frame transformation method called "percent averaging" was introduced to predict the VMAF quality of frames. To enhance the accuracy of perceived quality prediction, we also considered the type of encoded frame. For inter-frame predictions, our proposed CNN network extracts both textural and temporal features from the frames. We captured the motion estimation map between the I-frame and inter-frame to enable temporal feature extraction. Additionally, for textural feature extraction using CNN, we utilized the gray-scaled raw frame.

Chapter 5 introduced a novel deep CNN-based bit rate control and predictor. Rate control plays a crucial role in encoding as it directly impacts the quality of encoded frames by preventing overflow and underflow as explained in Section 1.2. Overflow in the receiver buffer leads to frame skipping and reduces the available bit rate budget for subsequent frames, while underflow results in unnecessary degradation of frame quality. Compression algorithms incorporate internal rate control models that have evolved with different encoder generations. HEVC, for instance, employs a rate-distortion optimization model to minimize distortion while maintaining an acceptable bit rate. Despite advancements in encoder rate control, there have been numerous research efforts to further improve its performance. However, these prior studies heavily rely on information from previous frames. While utilizing the bit rate and feature information from previous frames can lead to accurate rate control for frames with smooth motion, it often fails to provide reliable control for fast motions and frames with occluded areas, such as dynamic video frames. Recent advancements in machine learning techniques have aided in rate control for dynamic videos by utilizing engineering-crafted features to characterize rate control models. However, when dealing with high-resolution frames, the accuracy of feature extraction becomes

limited. To address these challenges, we propose a novel patch-based deep CNN-based rate control method. Our CNN-based rate control model operates based on rate-quantization functions, independent of previous frame information, while extracting frame and video features using an end-to-end CNN approach.

Our training approach involves two levels: first, we train the CNN bit rate predictor at the patch level and then at the frame level. For patch-wise bit rate prediction, we encode a series of patches individually and label them with the corresponding bit rates obtained from the encoding process. The CNN network is trained to map these patches to their respective bit rates, enabling it to learn to extract patch features. We utilize the extracted patch features to train a regressor that takes into account the effects of different areas within a frame on the frame's bit rate, thereby predicting the frame-wise bit rate. In the case of inter-frame predictions, we incorporate an additional stream of deep CNN networks that extract features from the motion estimation (ME) map of both inter-frame and Intra-frame frames. Our proposed method eliminates the dependency on previous frames and avoids the need for manual feature selection in the rate control process, providing more accurate and efficient rate control.

Chapter 6 focuses on our innovative CNN-based spatially and temporally adaptive encoding method. We present an end-to-end CNN-based intersection quantization parameter (QP) predictor that eliminates the need for hand-crafted features. The results of our proposed method demonstrate its superior accuracy compared to previous approaches that rely on manually designed spatial and temporal features.

The results obtained from the chapters highlight the improved performance of the proposed methods on the test dataset. However, there are certain factors that need to

be addressed in order to improve the distortion prediction, rate control, and adaptive encoding, making them suitable for implementation as integral components of HEVC compression algorithms.

We introduced a machine learning-based spatially adaptive encoding method that leverages hand-crafted features. Conventionally, hand-crafted features are extracted at the frame level, but in high-resolution frames, the variations between frame blocks can be substantial. To address this, we proposed extracting spatio-temporal features at the patch level and training the machine learning network accordingly. This approach allows for more precise analysis and adaptation within individual patches. For frame-level quality, rate, or QP prediction, we can employ another machine learning model to perform a weighted aggregation, considering the importance of each patch within the frame. This multi-level approach enhances the overall encoding performance and enables effective adaptation to the characteristics of high-resolution frames.

We focused on optimizing encoding parameters specifically for Intra-frame and inter-frame scenarios. Intra-frames are encoded independently and can serve as key frames for scene changes, while inter-frames often exhibit similarities to the preceding frames. By combining the spatio-temporal features extracted by our CNN-based approach with additional information from adjacent frames, we can enhance the optimization of inter-frame encoding. This integration allows for a more comprehensive analysis of the inter-frame content, taking into account both the CNN-based features and the contextual information from neighboring frames. As a result, the overall encoding optimization for inter-frames is improved, leading to enhanced compression performance.

The choice of dataset plays a crucial role in training deep CNN networks. To address the challenge of overfitting, it is essential to have a large diverse set of videos, as this significantly enhance the accuracy of the network's predictions. In our study, we utilized a publicly accessible high-definition (HD) video dataset due to limitations in obtaining HD video resources. However, with access to a larger and more diverse dataset, we could explore more complex structures while reducing the risk of encountering overfitting issues. By having a greater number of training samples, we could explore feature fusion techniques that may lead to even more precise predictions. Therefore, expanding the dataset resources would provide the opportunity to further enhance the accuracy and capabilities of the network.

Bibliography

[1] (1992). Video codec for audiovisual services at px64 kbit/s, H.261.

[2] (1993). Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s—Part (MPEG-1).

[3] (1999). Coding of audio-visual objects—part 2: Visual, ISO/IEC 14496-2 (MPEG-4 Visual version 1).

[4] Afonso, M., Katsenou, A., Zhang, F., Agrafiotis, D., and Bull, D. (2016). Video texture analysis based on hevc encoding statistics. In *Proceedings of the 2016 Picture Coding Symposium (PCS)*, pages 1–5.

[5] Afonso, M., Zhang, F., Katsenou, A., Agrafiotis, D., and Bull, D. (2017). Low complexity video coding based on spatial resolution adaptation. In *Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP)*, pages 3011–3015.

[6] Afonso, M., Zhang, F., and Bull, D. R. (2018). Video compression based on spatio-temporal resolution adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, **29**(1), 275–280.

[7] Aswathappa, B. H. K. and Rao, K. (2010). Rate-distortion optimization using structural information in H. 264 strictly intra-frame encoder. In *Conference Record of the 42nd Southeastern Symposium on System Theory (SSST)*, pages 367–370. IEEE.

[8] Bjontegaard, G. (2008). Improvements of the BD-PSNR model. *VCEG-AI11*.

[9] Bosch, M., Zhu, F., and Delp, E. J. (2011). Segmentation-based video compression using texture and motion models. *IEEE Journal of Selected Topics in Signal Processing*, **5**(7), 1366–1377.

[10] Bosse, S., Maniry, D., Wiegand, T., and Samek, W. (2016a). A deep neural network for image quality assessment. In *Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP)*, pages 3773–3777.

[11] Bosse, S., Maniry, D., Müller, K.-R., Wiegand, T., and Samek, W. (2016b). Neural network-based full-reference image quality assessment. In *Proceedings of the 2016 Picture Coding Symposium (PCS)*, pages 1–5.

[12] Bosse, S., Maniry, D., Müller, K.-R., Wiegand, T., and Samek, W. (2017). Deep neural networks for no-reference and full-reference image quality assessment. *IEEE Transactions on image processing*, **27**(1), 206–219.

[13] Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

[14] Brox, T., Bruhn, A., Papenberg, N., and Weickert, J. (2004). High accuracy optical flow estimation based on a theory for warping. In *Proceedings of the 2004*

Computer Vision-ECCV: 8th European Conference on Computer Vision, Part IV 8, pages 25–36. Springer.

[15] Bruckstein, A. M., Elad, M., and Kimmel, R. (2003). Down-scaling for better transform compression. *IEEE Transactions on Image Processing*, **12**(9), 1132–1144.

[16] Bull, D. R. (2014). *Communicating pictures: A course in Image and Video Coding*. Academic Press.

[17] Chen, J.-Y., Chiu, C.-W., Li, G.-L., and Chen, M.-J. (2010). Burst-aware dynamic rate control for H. 264/AVC video streaming. *IEEE Transactions on Broadcasting*, **57**(1), 89–93.

[18] Chiang, T. and Zhang, Y.-Q. (1997). A new rate control scheme using quadratic rate distortion model. *IEEE Transactions on Circuits and Systems for Video Technology*, **7**(1), 246–250.

[19] Choi, H., Nam, J., Yoo, J., Sim, D., and Bajic, I. (2012). Rate control based on unified RQ model for HEVC. *ITU-T SG16 Contribution, JCTVC-H0213*, pages 1–13.

[20] Daly, S. (2001). Engineering observations from spatiovelocity and spatiotemporal visual models. In *Vision Models and Applications to Image and Video Processing*, pages 179–200. Springer.

[21] Dong, J. and Ling, N. (2009). A context-adaptive prediction scheme for parameter estimation in H. 264/AVC macroblock layer rate control. *IEEE Transactions on Circuits and Systems for Video Technology*, **19**(8), 1108–1117.

[22] Dong, J. and Ye, Y. (2013). Adaptive downsampling for high-definition video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, **24**(3), 480–488.

[23] Emoto, M., Kusakabe, Y., and Sugawara, M. (2014). High-frame-rate motion picture quality and its independence of viewing distance. *Journal of Display Technology*, **10**(8), 635–641.

[24] García, B., López-Fernández, L., Gortázar, F., and Gallego, M. (2019). Practical evaluation of vmaf perceptual video quality for webrtc applications. *Electronics*, **8**(8), 854.

[25] Ghadiyaram, D. and Bovik, A. C. (2015). Massive online crowdsourced study of subjective and objective picture quality. *IEEE Transactions on Image Processing*, **25**(1), 372–387.

[26] Gonzales, R. C. and Wintz, P. (1987). *Digital image processing*. Addison-Wesley Longman Publishing Co., Inc.

[27] Guo, G., Wang, H., Bell, D., Bi, Y., and Greer, K. (2003). KNN model-based approach in classification. In *Proceedings of the 2003 On The Move to Meaningful Internet Systems 2003: CoopIS, and ODBASE: OTM Confederated International Conferences*, pages 986–996. Springer.

[28] Haralick, R. M., Shanmugam, K., and Dinstein, I. H. (1973). Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, pages 610–621.

[29] He, Z. and Mitra, S. K. (2002). Optimum bit allocation and accurate rate control for video coding via/spl rho/-domain source modeling. *IEEE Transactions on Circuits and Systems for Video Technology*, **12**(10), 840–849.

[30] Hollander, M., Wolfe, D. A., and Chicken, E. (2013). *Nonparametric statistical methods*. John Wiley & Sons.

[31] Hosking, B., Agrafiotis, D., Bull, D., and Easton, N. (2015). Spatial resampling of IDR frames for low bitrate video coding with HEVC. In *Proceedings of the 2015 Visual Information Processing and Communication VI*, volume 9410, pages 180–185. SPIE.

[32] Hosking, B., Agrafiotis, D., Bull, D., and Eastern, N. (2016). An adaptive resolution rate control method for intra coding in HEVC. In *Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1486–1490.

[33] Huang, Q., Wang, H., Lim, S. C., Kim, H. Y., Jeong, S. Y., and Kuo, C.-C. J. (2017). Measure and prediction of HEVC perceptually lossy/lossless boundary QP values. In *Proceedings of the 2017 Data Compression Conference (DCC)*, pages 42–51.

[34] IGDB (Year Accessed). Igdb - internet game database. Accessed on Date Accessed.

[35] Jenab, M. and Shirani, S. (2023). Deep cnn-based pre-encoding perceptual quality control and prediction. In *Proceedings of the 2023 IEEE International Conference on Image Processing (ICIP)*, pages 1–4.

[36] Jenab, M., Amer, I., Ivanovic, B., Saeedi, M., Liu, Y., Sines, G., and Shirani, S. (2018). Content-adaptive resolution control to improve video coding efficiency. In *Proceedings of the 2018 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–4.

[37] Jiang, M. and Ling, N. (2005). On enhancing H. 264/AVC video rate control by PSNR-based frame complexity estimation. *IEEE Transactions on Consumer Electronics*, **51**(1), 281–286.

[38] Jiang, M. and Ling, N. (2006). Low-delay rate control for real-time H. 264/AVC video coding. *IEEE Transactions on Multimedia*, **8**(3), 467–477.

[39] Jing, X., Chau, L.-P., and Siu, W.-C. (2008). Frame complexity-based rate-quantization model for H. 264/AVC intraframe rate control. *IEEE Signal Processing Letters*, **15**, 373–376.

[40] Kamaci, N., Altunbasak, Y., and Mersereau, R. M. (2005). Frame bit allocation for the H. 264/AVC video coder via Cauchy-density-based rate and distortion models. *IEEE Transactions on Circuits and Systems for Video Technology*, **15**(8), 994–1006.

[41] Kang, L., Ye, P., Li, Y., and Doermann, D. (2014). Convolutional neural networks for no-reference image quality assessment. In *Proceedings of the IEEE conference on computer vision and pattern recognition, ICVP*, pages 1733–1740.

[42] Karczewicz, M. and Wang, X. (2013). Intra frame rate control based on SATD. In *Conference Record of the 13th Meeting of JCTVC-M0257, ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, pages 18–26.

[43] Katsenou, A. V., Afonso, M., Agrafiotis, D., and Bull, D. R. (2016). Predicting video rate-distortion curves using textural features. In *Proceedings of the 2016 Picture Coding Symposium (PCS)*, pages 1–5.

[44] Kim, J. and Lee, S. (2016). Fully deep blind image quality predictor. *IEEE Journal of Selected Topics in Signal Processing*, **11**(1), 206–220.

[45] Kim, J., Nguyen, A.-D., and Lee, S. (2018). Deep CNN-based blind image quality predictor. *IEEE Transactions on Neural Networks and Learning Systems*, **30**(1), 11–24.

[46] Kim, W. J., Yi, J. W., and Kim, S. D. (1999). A bit allocation method based on picture activity for still image coding. *IEEE Transactions on Image Processing*, **8**(7), 974–977.

[47] Kwon, D.-K., Shen, M.-Y., and Kuo, C.-C. J. (2007). Rate control for H. 264 video with enhanced rate and distortion models. *IEEE Transactions on Circuits and Systems for Video Technology*, **17**(5), 517–529.

[48] Lee, B., Kim, M., and Nguyen, T. Q. (2013). A frame-level rate control scheme based on texture and nontexture rate models for high efficiency video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, **24**(3), 465–479.

[49] Lewis, J. P. (1995). Fast template matching. In *Proceedings of the 1995 Vision Interface*, volume 95, pages 15–19.

[50] Li, B., Li, H., Li, L., and Zhang, J. (2012). Rate control by R-lambda model for HEVC. *ITU-T SG16 Contribution, JCTVC-K0103*, pages 1–5.

[51] Li, S., Zhang, F., Ma, L., and Ngan, K. N. (2011). Image quality assessment by separately evaluating detail losses and additive impairments. *IEEE Transactions on Multimedia*, **13**(5), 935–949.

[52] Li, S., Xu, M., Wang, Z., and Sun, X. (2016). Optimal bit allocation for CTU level rate control in HEVC. *IEEE Transactions on Circuits and Systems for Video Technology*, **27**(11), 2409–2424.

[53] Liu, Y., Li, Z. G., and Soh, Y. C. (2006). A novel rate control scheme for low delay video communication of H. 264/AVC standard. *IEEE Transactions on Circuits and Systems for Video Technology*, **17**(1), 68–78.

[54] Liu, Z. (2003). Adaptive basic unit layer rate control for JVT. In *Proceedings of the 2003 JVT 7th Meeting, Pattaya*.

[55] Ma, S., Gao, W., and Lu, Y. (2005). Rate-distortion analysis for H. 264/AVC video coding and its application to rate control. *IEEE Transactions on Circuits and Systems for Video Technology*, **15**(12), 1533–1544.

[56] Mackin, A., Noland, K. C., and Bull, D. R. (2016). The visibility of motion artifacts and their effect on motion quality. In *Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP)*, pages 2435–2439.

[57] Marpe, D., Wiegand, T., and Sullivan, G. J. (2006). The H. 264/MPEG4 advanced video coding standard and its applications. *IEEE Communications Magazine*, **44**(8), 134–143.

[58] Milani, S., Celetto, L., and Mian, G. A. (2008). An accurate low-complexity rate

control algorithm based on (ρ, E_q)-domain. *IEEE Transactions on Circuits and Systems for Video Technology*, **18**(2), 257–262.

[59] Miller, F. P., Vandome, A. F., and McBrewster, J. (2009). *MPEG-2: Lossy Compression, Video Compression, Audio Compression (Data), ATSC (Standards), MPEG Transport Stream, MPEG-1 Audio Layer II, H. 262/MPEG-2 Part 2, MPEG-4, Advanced Audio Coding*. Alpha Press.

[60] Mittal, A., Soundararajan, R., and Bovik, A. C. (2012a). Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, **20**(3), 209–212.

[61] Mittal, A., Moorthy, A. K., and Bovik, A. C. (2012b). No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, **21**(12), 4695–4708.

[62] Moore, D. S. (2009). *Introduction to the Practice of Statistics*. WH Freeman and company.

[63] Moorthy, A. K. and Bovik, A. C. (2011). Blind image quality assessment: From natural scene statistics to perceptual quality. *IEEE Transactions on Image Processing*, **20**(12), 3350–3364.

[64] Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine learning (ICML-10)*, pages 807–814.

[65] Nasiri, R. M., Wang, J., Rehman, A., Wang, S., and Wang, Z. (2015). Perceptual quality assessment of high frame rate video. In *Conference Record of the 17th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6.

[66] Nguyen, V.-A., Tan, Y.-P., and Lin, W. (2008). Adaptive downsampling/upsampling for better video compression at low bit rate. In *Proceedings of the 2008 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1624–1627.

[67] Noland, K. *et al.* (2014). The application of sampling theory to television frame rate requirements. *BBC Research & Development White Paper*, **282**.

[68] Ou, Y.-F., Ma, Z., Liu, T., and Wang, Y. (2010). Perceptual quality assessment of video considering both frame rate and quantization artifacts. *IEEE Transactions on Circuits and Systems for Video Technology*, **21**(3), 286–298.

[69] Ou, Y.-F., Xue, Y., and Wang, Y. (2014). Q-star: A perceptual video quality model considering impact of spatial, temporal, and amplitude resolutions. *IEEE Transactions on Image Processing*, **23**(6), 2473–2486.

[70] Pisner, D. A. and Schnyer, D. M. (2020). Support vector machine. In *Machine learning*, pages 101–121. Elsevier.

[71] Rabe, C., Müller, T., Wedel, A., and Franke, U. (2010). Dense, robust, and accurate motion field estimation from stereo image sequences in real-time. In *Proceedings of the Computer Vision–ECCV: 11th European Conference on Computer Vision*, pages 582–595. Springer Berlin Heidelberg.

[72] Rijkse, K. (1996). H. 263: Video coding for low-bit-rate communication. *IEEE Communications magazine*, **34**(12), 42–45.

[73] Saad, M. A., Bovik, A. C., and Charrier, C. (2012). Blind image quality assessment: A natural scene statistics approach in the DCT domain. *IEEE Transactions on Image Processing*, **21**(8), 3339–3352.

[74] Sanz-Rodríguez, S., del Ama-Esteban, Ó., de Frutos-Lopez, M., and Díaz-de María, F. (2010). Cauchy-density-based basic unit layer rate controller for H. 264/AVC. *IEEE Transactions on Circuits and Systems for Video Technology*, **20**(8), 1139–1143.

[75] Series, B. (2012). Parameter values for ultra-high definition television systems for production and international programme exchange. In *Proceedings of the 2012 ITU-T, Bt. 2020*, pages 1–7.

[76] Sheikh, H. R. and Bovik, A. C. (2006). Image information and visual quality. *IEEE Transactions on Image Processing*, **15**(2), 430–444.

[77] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[78] Søgaard, J., Forchhammer, S., and Korhonen, J. (2015). No-reference video quality assessment using codec analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, **25**(10), 1637–1650.

[79] Sugawara, M., Omura, K., Emoto, M., and Nojiri, Y. (2009). P-30: Temporal sampling parameters and motion portrayal of television. In *Proceedings of the 2009 SID Symposium Digest of Technical Papers*, volume 40, pages 1200–1203.

[80] Sullivan, G. J., Ohm, J.-R., Han, W.-J., and Wiegand, T. (2012). Overview of

the high efficiency video coding (hevc) standard. *IEEE Transactions on circuits and systems for video technology*, **22**(12), 1649–1668.

[81] Utke, M., Zadtootaghaj, S., Schmidt, S., Bosse, S., and Möller, S. (2022). NDNetGaming-development of a no-reference deep CNN for gaming video quality prediction. *Multimedia Tools and Applications*, pages 1–23.

[82] Wang, M., Ngan, K. N., and Li, H. (2014). An efficient frame-content based intra frame rate control for high efficiency video coding. *IEEE Signal Processing Letters*, **22**(7), 896–900.

[83] Wang, M., Ngan, K. N., and Li, H. (2016a). Low-delay rate control for consistent quality using distortion-based Lagrange multiplier. *IEEE Transactions on Image Processing*, **25**(7), 2943–2955.

[84] Wang, N. and He, Y. (2003). A new bit rate control strategy for h. 264. In *Proceedings of the 2003 Fourth International Conference on Information, Communications and Signal Processing*, volume 3, pages 1370–1374.

[85] Wang, S., Rehman, A., Wang, Z., Ma, S., and Gao, W. (2011). SSIM-motivated rate-distortion optimization for video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, **22**(4), 516–529.

[86] Wang, S., Rehman, A., Wang, Z., Ma, S., and Gao, W. (2012). Perceptual video coding based on SSIM-inspired divisive normalization. *IEEE Transactions on Image Processing*, **22**(4), 1418–1429.

[87] Wang, S., Ma, S., Wang, S., Zhao, D., and Gao, W. (2013). Rate-GOP based

rate control for high efficiency video coding. *IEEE Journal of selected topics in signal processing*, **7**(6), 1101–1111.

[88] Wang, S., Rehman, A., Zeng, K., Wang, J., and Wang, Z. (2016b). SSIM-motivated two-pass VBR coding for HEVC. *IEEE Transactions on Circuits and Systems for Video Technology*, **27**(10), 2189–2203.

[89] Wang, Z., Simoncelli, E. P., and Bovik, A. C. (2003). Multiscale structural similarity for image quality assessment. In *Proceedings of the 2003 Conference Record of the Thrity-Seventh Asilomar Conference on Signals, Systems & Computers*, volume 2, pages 1398–1402.

[90] Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, **13**(4), 600–612.

[91] Winkler, S. (2012). Analysis of public image and video databases for quality assessment. *IEEE Journal of Selected Topics in Signal Processing*, **6**(6), 616–625.

[92] Wu, X., Zhang, X., and Wang, X. (2009). Low bit-rate image compression via adaptive down-sampling and constrained least squares upconversion. *IEEE Transactions on Image Processing*, **18**(3), 552–561.

[93] Xu, B., Pan, X., Zhou, Y., Li, Y., Yang, D., and Chen, Z. (2017). CNN-based rate-distortion modeling for H. 265/HEVC. In *Proceedings of the 2017 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4.

[94] Ye, P., Kumar, J., Kang, L., and Doermann, D. (2012). Unsupervised feature learning framework for no-reference image quality assessment. In *Proceedings of*

the 2012 IEEE conference on computer vision and pattern recognition, pages 1098–1105.

[95] Zhang, F., Mackin, A., and Bull, D. R. (2017). A frame rate dependent video quality metric based on temporal wavelet decomposition and spatiotemporal pooling. In *Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP)*, pages 300–304.

[96] Zhang, L. and Li, H. (2012). SR-SIM: A fast and high performance IQA index based on spectral residual. In *Conference Record of the 19th IEEE International Conference on Image Processing*, pages 1473–1476.

[97] Zhang, L., Zhang, L., Mou, X., and Zhang, D. (2011). FSIM: A feature similarity index for image quality assessment. *IEEE Transactions on Image Processing*, **20**(8), 2378–2386.

[98] Zhang, Y., Liu, H., Yang, Y., Fan, X., Kwong, S., and Kuo, C. J. (2021). Deep learning based just noticeable difference and perceptual quality prediction models for compressed video. *IEEE Transactions on Circuits and Systems for Video Technology*, **32**(3), 1197–1212.

[99] Zhang, Z. (2016). Introduction to machine learning: k-nearest neighbors. *Annals of Translational Medicine*, **4**(11).

[100] Zhou, M., Zhang, Y., Li, B., and Lin, X. (2017). Complexity correlation-based CTU-level rate control with direction selection for HEVC. *ACM Transactions on Multimedia Computing, Communications, and Applications*, **13**(4), 1–23.

[101] Zhou, M., Wei, X., Wang, S., Kwong, S., Fong, C.-K., Wong, P. H., Yuen, W. Y., and Gao, W. (2019). SSIM-based global optimization for CTU-level rate control in HEVC. *IEEE Transactions on Multimedia*, **21**(8), 1921–1933.

[102] Zhou, S., Li, J., Fei, J., and Zhang, Y. (2007). Improvement on rate-distortion performance of H. 264 rate control in low bit rate. *IEEE Transactions on Circuits and Systems for Video Technology*, **17**(8), 996–1006.