Heaven's Light is Our Guide

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Rajshahi University of Engineering & Technology, Bangladesh

A Fast Method for Predicting CU Partition in VVC Intra Partition

Author

Md. Merajul Rahman Shipon

Roll No. 1803173

Department of Computer Science & Engineering

Rajshahi University of Engineering & Technology

Supervised by

Rizoan Toufiq

Assistant Professor

Department of Computer Science & Engineering

Rajshahi University of Engineering & Technology

ACKNOWLEDGEMENT

First of all, I would like to thank almighty Allah, for his grace and blessings as well as for providing me with the diligence and enthusiasm along the way to accomplishing my thesis work.

I also want to express my sincere gratitude, admiration and heartfelt appreciation to my supervisor **Rizoan Toufiq**, Assistant Professor, Department of Computer Science & Engineering, Rajshahi University of Engineering & Technology, Rajshahi. Throughout the year, he has not only provided me with the technical instructions and documentation to complete the work, but he has also continuously encouraged me, offered me advise, assisted me, and cooperated sympathetically whenever he deemed necessary. His constant support was the most successful tool that helped me to achieve my result. Whenever I was stuck in any complex problems or situation he was there for me at any time of the day. Without his sincere care, this work not has been materialized in the final form that it is now at the present.

I am also grateful to respected **Prof. Dr. Md. Ali Hossain**, Head of the Department of Computer Science & Engineering and all the respective teachers of Department of Computer Science & Engineering, Rajshahi University of Engineering & Technology, Rajshahi for their valuable suggestions and inspirations from time to time.

Finally, I would like to convey my thanks to my parents, friends, and well-wishers for their true motivations and many helpful aids throughout this work.

April 26, 2024 — Md. Merajul Rahman Shipon

RUET, Rajshahi

Heaven’s Light is Our Guide

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Rajshahi University of Engineering & Technology, Bangladesh

CERTIFICATE

This is to certify that this thesis report entitled ***“A Fast Method for Predicting CU Partition in VVC Intra Partition ”*** *submitted by* ***Md. Merajul Rahman Shipon, Roll:1803173*** *in partial fulfillment of the requirement for the award of the degree of Bachelor of Science in Department of Computer Science & Engineering of Rajshahi University of Engineering & Technology, Bangladesh is a record of the candidate own work carried out by him under my supervision. This thesis has not been submitted for the award of any other degree.*

Supervisor	External Examiner
______________________________	______________________________
Rizoan Toufiq	
Assistant Professor	
Department of Computer Science & Engineering	Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology	Rajshahi University of Engineering & Technology
Rajshahi-6204	Rajshahi-6204

ABSTRACT

In multimedia applications, video encoding is essential for the effective storage and transfer of video content. Versatile Video Coding (VVC), one of the more recent advanced video coding standards, has greatly increased compression efficiency over earlier codecs, resulting in appreciable bandwidth reductions. However, because VVC requires a lot of computing power, its computational complexity presents a problem. In this study, we address the computational complexity issue of VVC by proposing a heuristic-based fast approach to predict Coding Tree Unit (CTU) partitions. The proposed method utilizes variance analysis to make early termination decisions during the partitioning process. Specifically, if the variance within a CTU is below a certain threshold, the method terminates further splitting, thus reducing unnecessary computational overhead. By assessing the gradient of pixel values, we prioritize QT partitions over MT partitions, effectively eliminating MT partitions when the gradient value satisfies certain conditions. In cases where neither variance nor gradient-based decisions suffice, we introduce a variance-of-variance approach to select the optimal partitioning scheme among QT and MT partitions. Furthermore, our methodology incorporates edge-based features extracted from the video frames to enhance the accuracy of partition predictions. Experimental results demonstrate the effectiveness of our approach, achieving a remarkable 48.17% reduction in encoding time. However, a slight increase in the Bjøntegaard Delta Bitrate (BDBR) (1.38%) is observed, highlighting the trade-off between computational efficiency and video quality. This study opens the door for further developments in the field of video compression optimization by supporting current work in this area.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Introduction

In today's modern world, videos have become a vital component of our day-to-day existence. Video is an indispensable medium for communication, entertainment, education, news, advertisement, surveillance, telemedicine, and various other applications [4]. From entertainment to education, videos play an important role in various aspects of our lives. Whether it's watching a movie or a tutorial, videos have the power to engage and captivate us in a way that other mediums cannot because videos can engage and inform audiences through rich visual and auditory experiences.

As the importance of video continues to grow, the demand for video files in high definition, including Ultra High Definition (UHD) and even higher formats such as 4K and 8K, has necessitated continuous innovation in video encoding and decoding techniques. The widespread availability of video content in high-definition and ultra-high resolution, coupled with the rise of online video platforms and streaming services, underscores the increasing importance of efficient video processing and compression techniques [5].

Video compression, also known as video coding or encoding, is essential for lowering the amount of video data, for storage and transmission while maintaining perceptual quality [6]. Efficient video compression not only conserves bandwidth and storage resources but also enables the delivery of high-quality video content across diverse networks and devices. Over the years, various international standards for video compression have been developed to establish common frameworks and ensure interoperability among different video processing systems and devices [7]. Notable standards include MPEG-2, H.264/AVC (Advanced Video Coding), H.265/HEVC (High-Efficiency Video Coding), and the latest Versatile Video Coding (VVC/H.266) standard [8].

The development of standards for video coding has driven significant advancements in compression efficiency, enabling the widespread adoption of digital video technologies in broadcasting, multimedia applications, and emerging domains like augmented reality (AR) and virtual reality (VR) [9].

1.2 Overview

Over the past years, video data has surpassed all other sources as the most widely consumed type of data. The rate of multimedia consumption has increased dramatically during the COVID-19 epidemic. Video data represents 80 percent of all Internet traffic in 2021, where it was 67 percent in 2016.[10]. Due to the explosive growth of video applications (broadcast services, video conferencing, video surveillance, industry video, 3D video, social media, mobile video, etc), the demand for high-resolution and quality video is growing rapidly. Video formats like 4K, and ultra-high definition (UHD) are currently becoming a favorite [11]. Those types of video formats are widely used for video conferencing, streaming services, and other major applications.

With increasing quality demand, the videos' transmission bandwidth and storage memory requirements are both increasing in direct proportion. Delivering top-notch videos efficiently to a broad range of viewers, (with limited bandwidth and storage) more advanced video coding methods are required. The Moving Picture Experts Group (MPEG) and the Video Coding Experts Group (VCEG) collaborated to issue a Call for Proposals (CfP) addressing the issue of video encoding and its improvements [12]. The Joint Video Experts Team (JVET) was established on October 27, 2017, through the collaboration of the Video Coding Experts Group (VCEG) and Moving Picture Experts Group (MPEG) [13]. In November 2020, JVET established the Versatile Video Coding (VVC/H.266) as a new video coding standard after carefully assessing the responses received from the Call for Proposals (CfP). Several novel coding techniques have been integrated into VVC, such as the cross-component linear model prediction (CCLM), matrix weighted intra prediction (MIP), multiple transform selection (MTS), adaptive loop filtering (ALF), and the quad-tree plus multi-type tree (QTMT) structure for coding unit (CU) partition, among others [14]. These new features have allowed VVC to achieve better coding performance and more potential applications. VVC can reduce bit-rate overall by 43.81%, in contrast to its forerunner HEVC [15]. However, the coding complexity of VVC has increased by 19 times compared to the HEVC [16]. The coding unit in VVC utilizes the QTMT division method, which is the main factor contributing to the heightened complexity resulting from the distinction between the two block division methods.

Subsequent to the JVET standard VVC, numerous attempts have been undertaken to minimize the intricacy of HEVC, its expansion, and VVC. While several fast HEVC techniques have resulted in considerable encoding time savings in earlier research, they cannot be directly applied to VVC due to the complete differences in the partition structure between the two standards. Nonetheless, these studies can be classified into two categories: heuristic approaches and data-centric approaches. The heuristic techniques make use of certain intermediate encoding attributes, like spatial correlation and textural complexity to create a statistical model for CU partition. The data-driven methods require a huge amount of data to automatically learn the CU partition.

This study aims to enhance the VVC intra mode in order to decrease the computational complexity. The research uses the heuristic approach to create a model for early prediction of CU partition. The VVC's intra-mode consumes the most time due to the intricate Rate-Distortion Optimization (RDO) procedure. The computational complexity will be reduced if the number of CUs that require the RDO process is decreased. Hence, this study aims to create a rapid algorithm for early prediction of CU partition, resulting in a decrease in the number of CUs that need to undergo the RDO process. The research used the VVC reference software VTM to measure the time complexity.

The following are the primary contributions of our work:

- In addition to the commonly used early termination strategy, which skips all subsequent

CU partitions directly, we additionally use the Sobel operator to compute gradient features in order to decide on the QT partition and terminate asymmetric rectangular partitions. Subsequently, employing a novel methodology that deviates significantly from the traditional direct calculation of the CU variance for homogeneity evaluation, a single partition is chosen from the five QTMT partitions based on the variance of variance of each sub-CU.

- In comparison to anchor VTM-20.0, the proposed rapid method decreases the encoding time by 48.17% while only slightly increasing the BDBR by 1.38%.

1.3 Motivation

These days, videos are everywhere. As multimedia technology advances, so does the availability of ultra-high definition (UHD), 4k, and 8k video, which has led to the enormous expansion of visual data. Videos are the most effective type of media to engage the audience as they offer a combination of visual and auditory elements. The applications of videos are wide-ranging and diverse. Some of the applications are

1. **Entertainment and Media :** Film and Television, Streaming Services, Gaming.
2. **Education and Training :** Educational videos for online courses, Video-based skill training, and demonstrations.
3. **Communication :** Video Conferencing, Social Media.
4. **Advertising and Marketing :** Digital Marketing, Showcasing products and services through visual storytelling.
5. **Healthcare :** Telemedicine, Medical Training.
6. **Surveillance and Security :** Real-time surveillance using video cameras.
7. **Research and Science :** Visual Data Analysis, Simulation, and Modeling.
8. **Art and Design :** Digital Art, VR, and AR Applications.
9. **Transportation and Navigation :** Video displays for navigation and driver assistance.

With these vast applications, the demand for higher picture quality and frame rates is increasing rapidly. As a result, the storage required for storing the videos and the bandwidth for transmitting the videos are increasing proportionally. So, in order to store and transmit the videos efficiently, enhance user experience, and energy efficiency advanced video encoding strategies are required. VVC is a promising video encoding standard, but the time complexity of VVC makes it infeasible to use in regular devices.

The primary objective of this study is to minimize the computational intricacy of VVC by making premature forecasts regarding CU partition, particularly in the domain of intra mode. Intra coding, which heavily depends on data within the present frame, plays a crucial role in attaining optimal compression efficiency in video encoding [5]. The RDO process in VVC intra-mode significantly contributes to the extensive time complexity. It is crucial to accurately predict the partitioning of Coding Units (CUs) in VVC intra-coding in order to optimize the computational complexity and maintain quality. This research aims to propose a novel method for predicting CU partitions in VVC intra-coding., with the objective of improving compression efficiency and enabling real-time video processing applications.

1.4 Objectives

The primary aim of this study is to enhance the most recent video coding standard, VVC, such that the time complexity is minimized and it can even be used in mobile phones like its predecessor HEVC.

The primary objectives of this research are:

1. **Analyzing the Versatile Video Coding :**
    - Understand how VVC works in the Intra-mode and how to use the VVC reference software, VTM.
2. **Motion Estimation :**
    - Develop a method to efficiently determine motion in Coding Units (CUs) and skip homogeneous blocks during CU partitioning based on variance and Sobel operator.

3. **Predicting CTU partitioning :**

    - Develop a method to efficiently and correctly predict the partitioning mode for non-homogeneous blocks based on the variance of variance.

4. **Implementing a Fast Method for VVC Intra Partition :**

    - Create and execute an innovative strategy to enhance the intra-coding process in Versatile Video Coding (VVC) with the aim of minimizing computational complexity.

5. **Time Complexity Reduction & Quality Observation :**

    - Evaluate the proposed optimizations in terms of reducing time complexity while observing the impact on video quality.

The objectives aim to decrease the computational intricacy while maintaining the video coding quality through the Versatile Video Coding (VVC/H.266) framework by addressing specific challenges related to motion estimation, partitioning prediction, and computational optimization.

1.5 Challenges

During the research, we have faced many challenges. Some of the challenges are.

1. **Reliance on Handcrafted Features in Heuristic Process :**

    - This research aims to develop a fast method for CU partitioning in the VVC intra-mode using a heuristic approach. The heuristic approaches are greatly dependent on handcrafted feature extraction. This research used the Sobel operator, variance, and variance of variance as features for motion estimation and predicting CU partition.

2. **Homogeneity Detection :**

    - Implement robust algorithms for accurately determining the homogeneity of coding blocks, utilizing advanced statistical methods to classify blocks effectively.

3. **Partition Mode Selection :**

- Design a decision-making method to automatically select the optimal partition mode (e.g., QTMT modes) for non-homogeneous blocks based on the variance of variance of all possible partition blocks.

4. **Definition and Adaptation of Thresholds :**

    • One of the major challenges was determining the threshold to classify a CU as homogeneous or non-homogeneous based on its variance and Sobel operator.

5. **Lack of learning resources :**

    • Video compression, also known as video coding, is a complex research field with limited online resources. Unlike Machine Learning and Deep Learning, it is much harder to learn about video coding.

6. **Time Consuming:**

    • Another challenge was time. The video coding processes are hugely time-consuming and require constant focus. For some videos, it may take 4 to 5 hours to analyze the encoding decoding of a single frame.

1.6 Thesis Organization

The report is organized into six chapters including this chapter: ***Introduction*** where all the related topics are discussed which are needed for understanding the research work. The outline of the remaining book are structured in the following manner:

Chapter 2

Topic - Video & Video Coding

This chapter describes the principles of digital video, including source types and metrics for visual quality. It also presents video compression, the process of video encoding and decoding, and the most recent video coding norms.

Chapter 3

Topic - Literature Review

This section discusses various efforts aimed at reducing the time complexity of the most recent video coding standard, VVC/H.266.

Chapter 4

Topic - Methodology

This chapter discusses the proposed fast method for predicting the CU partition for VVC intra-partition. It also discusses the used video sequences in the research.

Chapter 5

Topic - Result & Performance Analysis

This chapter analyses the experimental result and performance of the proposed architecture along with the comparison with related works. The metrics which were used to evaluate the performance of encoding and decoding are also described here.

Chapter 6

Topic - Conclusion & Future Work

Through this chapter, the research work has been concluded. This article gives a summary of my research's findings. I have also tried to make an effort to highlight limitations and potential future work areas of my work.

1.7 Conclusion

The first chapter establishes the scene by emphasizing how common videos are in today's society and how important they are in a wide range of areas, such as communication, entertainment, healthcare, and education. It emphasizes the growing need for high-definition video content and the resulting requirement for advanced video encoding methods in order to efficiently store and transfer such data. The establishment of video coding norms resulted in the formation of the Versatile Video Coding (VVC/H.266) standard by JVET. The VVC can reduce bit-rate by about 43.81% than HEVC. However, the computational complexity has increased by 19 times. Motivated by these trends, the objective of this study is to decrease the computational complexity of VVC intra-mode by implementing novel techniques for predicting early CU partition. So that fewer CUs need to go through the RDO process. The stated objectives include an in-depth investigation of VVC as well as the development and implementation of quick approaches to maximize intra-coding performance.

In conclusion, this chapter establishes the rationale for investigating VVC intra-mode complexities and outlines the primary objectives and challenges of the research. By addressing these challenges and objectives, this study aims to contribute novel insights and methodologies towards optimizing VVC encoding processes, ultimately, pushing forward the cutting-edge in video compression technologies.

Chapter 2

Video & Video Coding

2.1 Introduction

The word video means ”a recording of an image or of moving images” [17].

Video can be defined from many aspects. Such as:

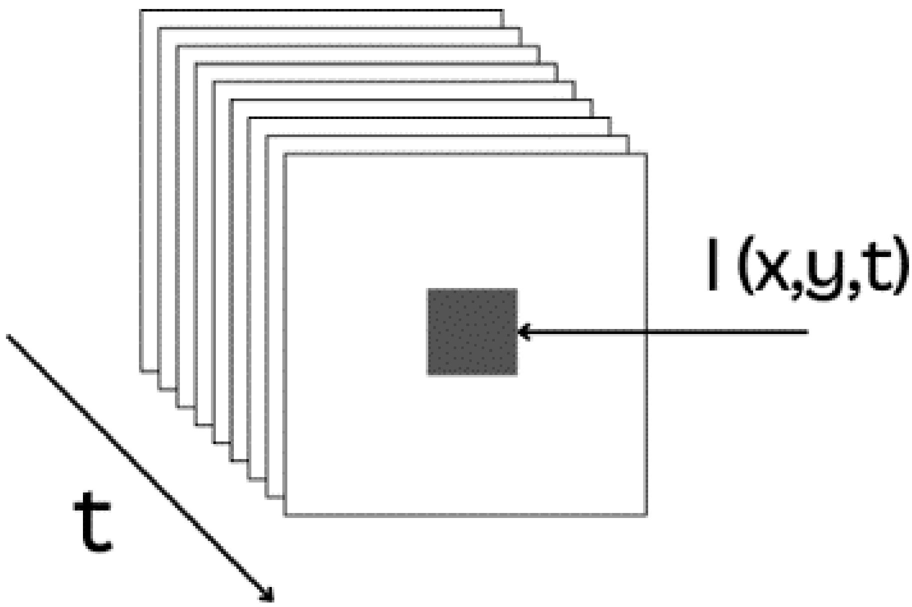


Figure 2.1: A typical Video representation.

1. **General Definition :**

   - A video is a series of pictures that are played in rapid succession, typically with accompanying sound, to create the illusion of motion and convey visual information. This multimedia format combines moving images and sound and is commonly created and viewed electronically on displays such as those found on computers, televisions, and mobile devices [18].

2. **Technical Definition :**

    - A video is a digital depiction of moving visual images that have been captured, encoded, and saved in a specific format [19].

In the context of space and time, a natural visual scene demonstrates continuity. To represent a natural visual scene digitally, it is essential to capture both its spatial and temporal elements. This task involves spatially sampling the real scene by placing it on a rectangular grid within the video image plane, and temporally recording a series of still frames or components of frames at regular time intervals. Digital video essentially serves as a representation of this sampled video scene in digital format.

The invention of video is generally credited to Louis Le Prince, with his creation of the first-ever motion picture using a single-lens camera in 1888. This video, titled ”Roundhay Garden Scene,” is a short, silent clip depicting people walking in a garden [20].

Figure 2.2: Snapshot of ”Roundhay Garden Scene”

[20]

From then the video format underwent a period of significant development.

2.2 Types of Video

Videos can be categorized into distinct groups according to their resolution, which determines the quality and the degree of detail of the video. Some common categories of videos are:

1. Standard Definition :

i **QCIF (Quarter Common Intermediate Format)**

- Resolution: 176 × 144 pixels.
- Pixel Rate (at 30 fps): 760,320 pixels per second.
- Description: Low-resolution format used in early video conferencing and digital video applications.

ii **SIF(525) (Source Input Format)**

- Resolution: 352 × 240 pixels.
- Pixel Rate (at 30 fps): 2,534,400 pixels per second.
- Description: Standard resolution for NTSC television systems, offering improved quality over QCIF.

iii **4SIF(525) (4-times Source Input Format)**

- Resolution: 704 × 480 pixels
- Pixel Rate (at 30 fps): 10,108,800 pixels per second
- Description: Enhanced version of SIF(525) with higher resolution, used in professional video production.

2. High Definition (HD) :

i **720p HD (High Definition)**

- Resolution: 1280 × 720 pixels.
- Pixel Rate (at 30 fps): 27,648,000 pixels per second.

- Description: Standard high-definition format with good image quality, commonly used for digital television broadcasts and online streaming.

ii **1080p Full HD (Full High Definition)**

- Resolution: 1920 × 1080 pixels
- Pixel Rate (at 30 fps): 62,208,000 pixels per second
- Description: Full high-definition format offering enhanced clarity and detail compared to 720p, suitable for Blu-ray discs, streaming services, and professional video production.

3.Ultra High Definition (UHD) :

i **4k**

- Resolution: 3840 × 2160 pixels.
- Pixel Rate (at 30 fps): 248,832,000 pixels per second.
- Description: Ultra high-definition format with four times the resolution of Full HD, ideal for large screens and cinematic experiences.

ii **8K**

- Resolution: 7680 × 4320 pixels.
- Pixel Rate (at 30 fps): 746,496,000 pixels per second.
- Description: Cutting-edge ultra high-definition format providing exceptional detail and realism, used in advanced digital cinema and professional video production.

For practical purposes, video must be compressed since uncompressed video at these resolutions requires a very large storage or transmission capacity.

2.3 Applications of Video

In the realm of digital media and technology, video serves a pivotal role across numerous applications, each contributing to the advancement and utilization of visual content. These applications are instrumental in diverse fields, showcasing the practical significance and impact of video technology:

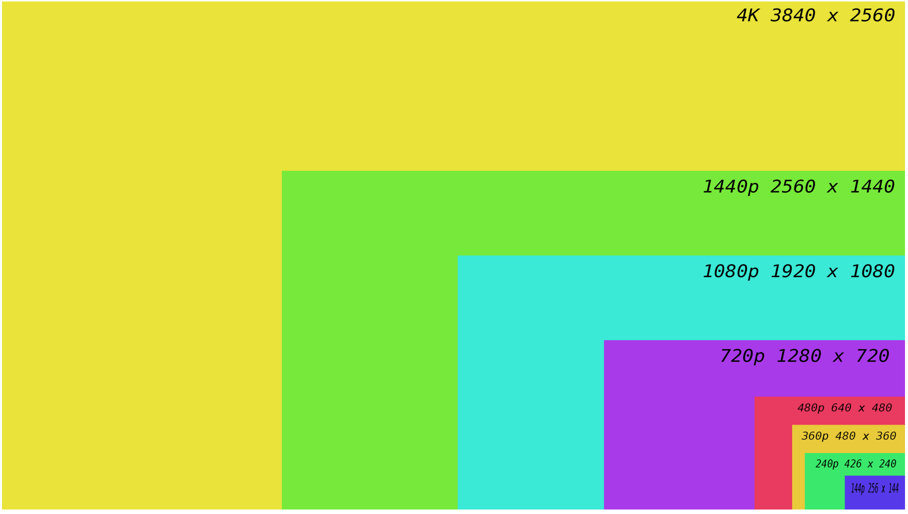


Figure 2.3: Types of video and their resolution comparison

[21]

1. **Entertainment and Media**: Video plays a central role in entertainment and media industries, powering movies, television shows, music videos, and gaming experiences. It enables immersive storytelling and engages audiences through captivating visual narratives.

2. **Education and Training**: Video is integral to modern education, facilitating e-learning platforms, online courses, and training materials. It enhances educational content with visual demonstrations, making complex topics more accessible and engaging.

3. **Communication and Collaboration**: Video conferencing technology enables remote communication, virtual meetings, and interviews. Social media platforms leverage video content for interactive communication and content sharing.

4. **Marketing and Advertising**: Digital marketing heavily relies on video content for advertising campaigns, brand promotions, and customer engagement. Video ads are effective in conveying messages and showcasing products or services.

5. **Healthcare and Medicine**: In healthcare, video technology supports telemedicine services, medical training, and surgical procedures. It enhances patient education and enables remote consultations.

6. **Security and Surveillance**: Video surveillance systems are critical for monitoring public spaces, facilities, and private properties, enhancing security and safety measures.

7. **Scientific Research and Exploration**: Videos from satellites, drones, and scientific instruments aid in research, environmental monitoring, and exploration endeavors. They capture critical data and facilitate analysis in various scientific disciplines.

8. **Art and Creative Expression**: Video serves as a medium for artistic expression, experimental film, and multimedia installations. Artists leverage video technology to push creative boundaries and convey unique perspectives.

9. **Industrial and Engineering Applications**: In industrial settings, video assists in quality control processes, assembly line monitoring, and engineering simulations. It supports visualization of complex systems and aids in problem-solving.

These applications underscore the versatility and impact of video technology across industries, driving innovation, communication, and creativity. By exploring these diverse applications, we gain insights into the multifaceted nature of video and its transformative role in modern society.

2.4 Video Quality Measurement

The quality of the video pictures that are shown to a viewer must be ascertained to assess and compare video data. Because there are actually numerous variables that might influence the outcome, measuring the visual quality is a challenging and frequently imperfect skill. In general, the quality of a video can be assessed by considering both subjective and objective characteristics

2.4.1 Subjective quality

Although video quality is subjective by nature, it is impacted by a wide range of subjective elements. For example, temporal fidelity, or whether movement seems ”smooth” and realistic, and spatial fidelity, or how well certain areas of the picture can be viewed and if there is significant noticeable distortion.

ITU-R Recommendation BT.500-11 [22] provides a comprehensive overview of various test protocols employed to evaluate subjective quality. Among these protocols, the Double Stimulus Continuous Quality Scale (DSCQS) is widely adopted. Nevertheless, it is important to acknowledge that the results obtained through DSCQS can exhibit significant variations depending on the assessor and the specific video sequence under examination.

2.4.2 Objective quality

Subjective quality measurement is often challenging and costly, making it advantageous to have the ability to measure quality using an algorithm. Engineers working on video encoding and decoding systems frequently rely on objective quality measurement tools, with the Peak Signal to Noise Ratio (PSNR) being a widely utilized metric.

PSNR:

The Peak Signal to Noise Ratio (PSNR) is evaluated using the logarithmic approach, which relies on the Mean Squared Error (MSE) between the original and modified video or image frame. The MSE is compared to (2n - 1)^2, where n represents the number of bits in a single instance of the picture. This value represents the square of the highest possible signal level in the image.

$$\text{PSNR}_{\text{dB}} = 10 \log_{10} \frac{(2^n - 1)^2}{\text{MSE}}$$

PSNR is a widely used quality metric utilized to evaluate the ”quality” of encoded and decoded video images due to its ease of calculation.

2.5 Video Compression

The procedure of compressing data into fewer bits is known as compression. The procedure of transforming digital video into an electronic form that fits either transmission or preservation, usually with fewer bits, is called video encoding (also known as video compression). For video to be practically stored and transmitted, it must be reduced. ”Raw,” or not encoded, video normally needs a high bitrate—roughly 216 Mbits for one second of raw Standard Definition footage.

The compressor (encoder) and the decompressor (decoder), two complementing systems, are involved in encoding. Before being communicated or stored, the encoder transforms the

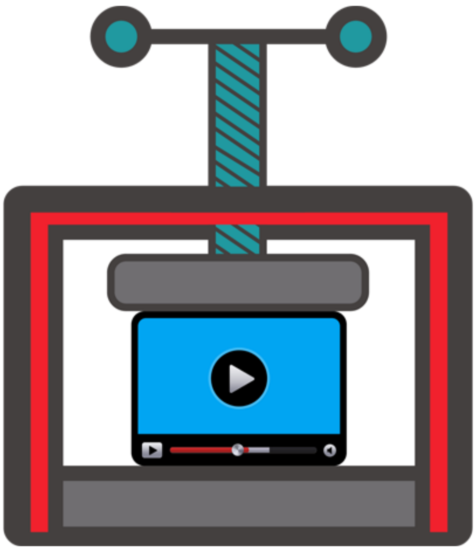


Figure 2.4: Video Compression

raw data into a reduced format that requires fewer bits. Then, the decoder transforms the compacted version into an exact copy of the genuine video information. It is common to refer to the encoder/decoder combination with the acronym CODEC (en**CO**der/**DEC**oder).

Duplicity, or elements not required to accurately reproduce the information, are removed from

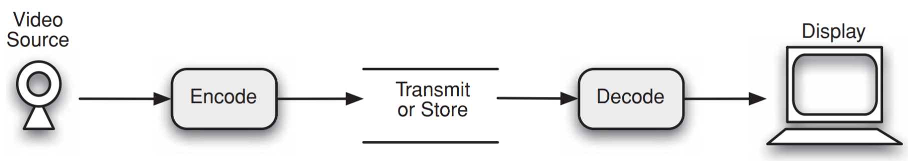


Figure 2.5: Basic Video Coding process block diagram[1]

compressed data. A multitude of data kinds exhibit probabilistic duplication and may be efficiently compressed by lossless compression, ensuring that what is returned at the decoder's end is a replica of the initial information. Regrettably, lossless encoding of picture and video data results in a relatively mild reduction. Greater encryption can only be attained with lossy compression methods. Significantly greater compression efficiency can be obtained at the cost of impairment of visual appeal in a lossy compressing scheme, where the decoded data differs from the original information. Lossy video encoding methods are built upon the principle of removing subjective redundancy, which refers to elements in the image or video that can be removed without significantly affecting the viewer's perception of visual quality.

2.6 Video Codec

A video CODEC creates an exact duplicate or estimate of the original frame by encoding an original picture or video frame into a reduced size and then decoding it. Three primary essential components make up a video encoder: an entropy encoder, a spatial model, and a prediction model. The prediction model receives an uncompressed, or ”raw,” video stream as input. By taking advantage of any commonalities among nearby frames of video and/or neighboring image instances, the prediction model aims to minimize duplication. Normally, this is achieved by generating a prediction of the current video frame or segment of video data. The residual

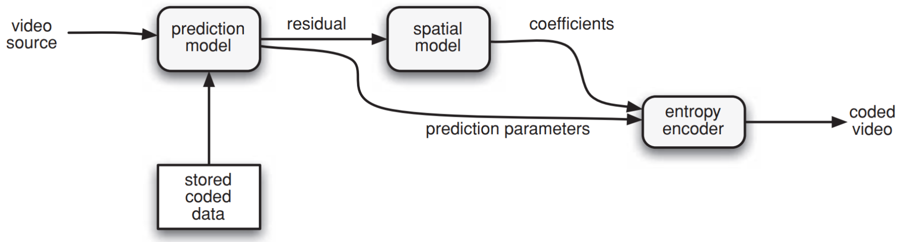


Figure 2.6: Basic Encoder [1]

frame is created by the prediction model through the subtraction of the predicted frame from the current frame, in addition to a set of model parameters that define the intra-prediction type or describe the motion compensation process. This residual frame is then employed by the spatial model to exploit similarities between adjacent samples within the residual frame in order to reduce spatial redundancy. The spatial model produces a sequence of quantized transform coefficients. The prediction model parameters are compressed using an entropy encoder to remove statistical redundancy in the data. The compressed bit stream or file is produced by the entropy encoder for transmission or storage purposes. A video frame is reconstructed by the video decoder using the compressed bit stream, with the decoding process being essentially the reverse of the encoding process.
Popular video Codecs are:

1. MPEG-4 Part 2 (DivX, Xvid) – Published in 1999
2. H.264/AVC (Advanced Video Coding) – Published in 2003
3. VP9 – Published in 2012

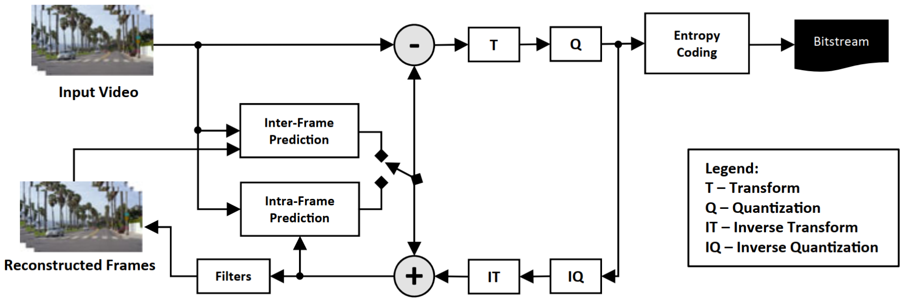


Figure 2.7: VVC encoder structure. [2]

4. H.265/HEVC (High-Efficiency Video Coding) – Published in 2013

5. AV1 (AOMedia Video 1) – Published in 2018

JVET published the Versatile Video Codec (VVC/H.266), the most recent video codec standard, in 2020.

2.7 Versatile Video Coding

All main video compression standards, including AVC and HEVC, are built on the block-based hybrid video encoding technique, which was used in the development of VVC. Similar to other hybrid codecs, VVC divides the video sequence's frames into blocks for separate processing. Various techniques such as intra- or inter-frame estimations, transformation, quantization, entropy coding, inverse quantization, inverse transformation, and filtering are employed in the processing of these blocks. All the latest video codecs on the market today follow these procedures. The goal of using this set of procedures is to store the video data with less duplicate data [1].

Spatial Redundancy : In a video frame, spatial redundancy characterizes the correlation or resemblance between adjacent samples or pixels. Because of this redundancy, which suggests that neighboring pixels frequently have similar values, effective compression techniques can be used to minimize data redundancy and enhance video transmission or storage.

Temporal Redundancy : Temporal redundancies in a video sequence refer to the recurring patterns or similarities observed between consecutive frames. The resemblance or correlation between successive frames is described by this phrase, which allows video compression techniques to take advantage of this redundancy for effective transmission or storage.

The examination of spatial redundancy in a frame involves the utilization of intra-frame prediction. In the intra-mode, only the previously processed video data from the frame being compressed is employed. Residues represent the differences between the original and predicted blocks. Subsequently, the residues undergo transform and quantization procedures to remove spatial redundancies in the frequency domain. Ultimately, the quantized coefficients are processed by the entropy coding in order to lessen the entropic recurrence.

To determine the optimal block size, optimal estimation mode, and various other encoding choices, VVC employs a sophisticated Rate-Distortion Optimization (RDO) procedure. In order to determine the optimal arrangement that minimizes the Rate-Distortion cost (RD-cost), a wide range of coding options are evaluated. The RD-cost is calculated based on the bit rate required for estimation and the variation between the original and estimated blocks.

2.8 Block Partitioning

Every frame in a video clip may be segmented into cuts, each of which represents a frame region independent from adjacent frame sections in terms of data dependency. The cuts are organized into Coding Tree Units (CTUs), which can then be divided into smaller Coding Units (CUs). For modern video codecs to reduce data efficiently, block partitioning is crucial. Consequently, in order to accommodate block sizes bigger than HEVC and provide an effective reduction rate, the VVC researchers looked into a number of fresh block partitioning architectures. The input frame is divided into CTUs per the VVC standard, each of which covers a quadrilateral area of no more than 128 by 128 pixels.

VVC incorporates the Multi-Type Tree (MTT) segmenting architecture in addition to the existing HEVC Quadtree (QT) structure, using a coding-tree-based partitioning technique. This enables the generation of rectangle CU forms through Binary Tree (BT) and Ternary Tree (TT). The combination of QT and MTT architectures is referred to as QTMT, or Quadtree with Nested Multi-Type Tree. With the QTMT architecture, six different types of partitioning are supported.

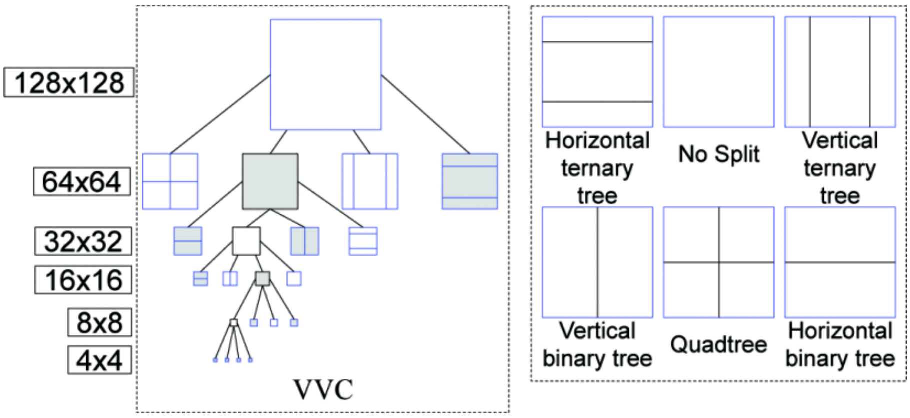


Figure 2.8: An illustration of QTMT partitioning structure [3].

If a coding unit (CU) is categorized as having no split, the encoding process is performed using the current dimensions of the CU. Alternatively, the CU can be partitioned into four sub-CUs of equal size using the QuadTree (QT) method. In the case of Binary Tree (BT), the CU is divided into a pair of identical sub-CUs either vertically (BTV) or horizontally (BTH). Similarly, the Ternary Tree (TT) approach allows for the division of a CU into three sub-CUs using a 1:2:1 proportion, either vertically (TTV) or horizontally (TTH).

The QTMT splitting architecture gives more freedom in representing block dimensions and forms by the encoder. As a result, these block partitioning architectures have greater coding performance since they can adjust to different video properties. Yet, this great flexibility also translates into a large computational effort because the split options are assessed during the RDO process to find the best CTU partitioning.

2.9 Entropy Coding

Entropy coding plays a crucial role in the video compression procedure of Versatile Video Coding (VVC), which lowers data redundancy and achieves effective encoding. In the encoding pipeline, the transformation and quantization stages are usually followed by the entropy coding stage. A collection of symbols that indicate different parts of a video clip is transformed into a reduced bitstream by the entropy encoder, making it ready for transport or storage. The entropy encoder of the VVC utilizes the Context Adaptive Binary Arithmetic Coding (CABAC) in the same manner as HEVC [23]. At the quantization outputs along with the lateral data (motion vectors, prediction modes, etc.), the CABAC carries out a zero-loss entropy reduction.

2.10 Conclusion

In conclusion, this chapter's examination of video technology has brought to light its development, uses, and approaches for evaluating its quality. Video coding is essential for lowering digital video data size so that it may be sent and stored more effectively. Sophisticated encoding techniques are used by several video codecs, including AVC, HEVC, and VVC, to reduce video data by taking advantage of temporal and spatial redundancy. The most recent standard, VVC, provides notable improvements in compression effectiveness by using creative block partitioning and entropy coding techniques. Enhancements in multimedia communication, compression efficiency, and video quality are all being fueled by these advancements in video coding technology.

Chapter 3

Literature Review

3.1 Introduction

Numerous research on the subject of intra-partitioning has been carried out throughout the years, covering a wide range of encoding standards created by numerous organizations as well as the pioneers of current standards like AVC/H.264, HEVC/H.265 and VVC/H.266. The objective of these investigations is to enhance the standard and effectiveness of video compression through the examination of the intricacies of intra-partitioning methods.

The newest video coding standard, VVC, can drastically lower bitrates but at the expense of massive computational complexity. There are numerous approaches for VVC that lower computing complexity. Three general categories can be used to group the methods. First, there is the conventional correlation-based(Heuristic) approach, which primarily uses gradient or variance information to identify textures. The second is the standard machine learning approach, which often makes use of support vector machines (SVM) and decision trees (DT). An alternative method involves the utilization of artificial neural networks, where a network is constructed and subsequently employed for the purpose of classification. The outcome of the network shows an alternative option for the intra-partition issue: if it is necessary to divide the CU further.

3.2 Related Works

3.2.1 Heuristic Approach

The study conducted by Zhang et al. [24], suggested an effective CU partition technique is suggested to keep encoding efficiency while lowering computing complexity. This technique uses the residual coefficient distribution of CUs and texture information to bypass low-probability partition types beforehand. According to experimental data, our fast algorithm maintains the video's objective quality while achieving a considerable 52.3% reduction in coding time at a small increase of 1.31% in BD-BR.

In the research conducted by Zhao et al. [25], they propose a rapid method that utilizes both pixel content similarity and the most likely partition pattern list (MPPPL). Initially, the MPPPL is constructed by considering the average texture complexity difference of sub-coding units across different partition modes. Subsequently, the sub-block pixel mean differences are utilized to optimize the MPPPL or determine the optimal partition mode. Consequently, the study focuses on analyzing the selection procedures for reference lines in intra prediction, where pixel content similarity is employed to eliminate unnecessary reference lines. The experimental results demonstrate that the suggested approach not only achieves a minimal increase of 1.23% in BDBR (Bjøntegaard Delta Bit Rate) but also significantly reduces the encoding time by 52.26% compared to VTM-13.0.

Tsai et al. [26] conducted a study that proposed a rapid intra-coding method based on an analysis of visual perception. This technique utilizes the concept of just-noticeable distortion (JND) to identify visually distinct (VD) pixels within a coding unit (CU) by considering the average backdrop luminance. By applying specific thresholding conditions, the technique selectively deactivates intra sub-partitions and matrix weighted intra prediction based on variances in the number of VD pixels across different MTT divides of a CU.

To expedite the decision-making process for horizontal and vertical splitting in binary and ternary trees, the suggested methodology incorporates machine learning techniques, specifically random forest classifiers. These classifiers leverage the quantization parameter and VD pixel information to optimize the splitting process.

The effectiveness of the proposed technique is supported by experimental results, which demonstrate a remarkable 47.26% reduction in encoding time. Additionally, there is an average marginal improvement of 1.535% in Bjøntegaard Delta Bitrate (BDBR) under the All Intra configuration.

In their research, Zhao et al. [27] conducted a study that proposed a partition algorithm with low complexity, which makes use of edge features. Initially, the coding frame's edges are extracted and categorized into vertical and horizontal edges through the utilization of the Laplacian of Gaussian (LOG) operator. Subsequently, feature values are computed for these edges by dividing coding units (CUs) equally into four sub-blocks in both the horizontal and vertical directions. By employing these feature values, unnecessary partition patterns are identified and skipped in advance. The partition process for CUs lacking edges is terminated based on the nearby CU depth information. Experimental data reveals that this suggested approach reduces the encoding time by an average of 54.08%, while only resulting in a 1.61% increase in BDBR (Bjøntegaard delta bit rate) compared to VTM-13.0.

3.2.2 Machine Learning Approach

The research carried out by Naima Zouidi and colleagues (2022)[28] presents a novel intra-mode decision framework based on multitask learning (MTL), which makes use of a vast dataset for intra-prediction. Through the application of MTL, the framework enhances the encoding decisions by concurrently learning various tasks, including intra-prediction mode selection. Results from experimental assessments show that the suggested framework leads to a notable decrease in complexity, by as much as 30%, alongside a marginal rise in the Bjontegaard bit rate (BD-BR).

The research carried out by Mingying Li and colleagues [29] utilizes trained Support Vector Machine (SVM) classifier models to quickly identify CU partition modes within a fast CU size decision scheme. Furthermore, the study incorporates an enhanced search step to decrease the amount of intra-prediction modes taken into account during Rate-Distortion Optimization (RDO) mode selection. Results from simulations show the efficacy of the suggested approaches, achieving a significant 55.24% decrease in encoding runtime with minimal effect on Bjontegaard Delta Bit Rate (BDBR).

The research conducted by Ye Li et al. [30] delves into two primary strategies: intra-frame mode selection and coding unit (CU) partition decision-making, aimed at enhancing coding efficiency and reducing computational requirements. Initially, a partitioning decision algorithm based on decision trees is proposed to handle the flexible partitioning structure of QTMT. By utilizing textural features to determine whether to skip the MT division based on partitioning needs, this method combines QT and MT division decisions. The use of four decision tree classifiers enables the effective identification of various partition types when MT partitioning is required. Furthermore, the study introduces an ensemble-learning based approach to optimize mode prediction termination for intra-frame mode selection. This technique streamlines the mode selection process by efficiently terminating redundant candidate modes through reordering and assessing prediction accuracy. The results of the experiments demonstrate the effectiveness of these strategies; on average, they result in a time savings of 54.74% compared to the VVC test model (VTM), with only a slight increase in the Bjontegaard Delta Bit Rate (BDBR) by 1.61%.

The study conducted by Luo et al. [31], presents an effective Intra-prediction technique designed to maximize VVC performance. The suggested approach tackles a number of significant issues with video coding. First, a multi-modes fusion method is presented to combine prediction modes from nearby blocks, acknowledging that a single prediction mode might not always produce the best results and that these blocks show a strong correlation. By making the most of spatial correlation, this method increases prediction accuracy. Secondly, an adaptive template matching method is meant to handle complex areas more efficiently. This approach uses various block-size-based weighting algorithms to efficiently handle a variety of texture properties. In addition, the algorithm adaptively creates a new chroma mode by integrating the Cross-Component Linear Model (CCLM) with the Derived Mode (DM).

3.2.3 Neural Network Approach

The research conducted by Fatma Belghith1 et al. [32] introduces a novel deep-QTMT partition strategy that utilizes a rapid convolutional neural network-ternary tree (CNN-TT) to overcome the challenges associated with RDO-QTMT partitioning. The primary objective of this strategy is to minimize encoding time by accurately predicting the optimal intra-QTMT decision tree. Initially, a database is established containing CU-based ternary tree (TT) partition depths for various video content. Subsequently, a CNN-TT model is developed to predict the QTMT partition at 32x32, employing a three-level TT structure. Distinct threshold values are assigned to each level during the model's implementation based on the expected probabilities of the CNN-TT model. This methodology strikes a balance between coding efficiency and encoding complexity. The experimental findings suggest that the deep-QTMT partition strategy delivers satisfactory rate-distortion performance while significantly reducing encoder time (with average reductions ranging from 23% to 58%).

Li et al. [33] conducted a study that introduces a novel approach for reducing the complexity of Versatile Video Coding (VVC) encoding. The approach utilizes DenseNet and decision tree (DT) classifiers to make fast coding unit (CU) partition decisions. The process involves employing DenseNet to extract spatial feature vectors, which forecast the border probabilities of 4×4 blocks within 64×64 coding units. These feature vectors are then used as input for the DT classifier to identify the top division modes with higher prediction probabilities. By excluding less likely modes, the computational complexity is significantly reduced. The algorithm's selection of the ideal partition mode is based on rate-distortion (RD) cost evaluation. Experimental results on VTM10.0 demonstrate a substantial 47.6% reduction in encoding time, accompanied by a negligible increase of only 0.91% in Bjøntegaard delta bit rate (BDBR). This study highlights the effectiveness of combining DenseNet-based feature extraction and DT classification in achieving efficient CU partitioning in VVC..

The study conducted by Liu et al. [34], presented a CNN-based technique to quicken the VVC interpartitioning process. Initially, a fresh depiction of QTMT partitioning is presented, which is obtained from the partition path. Second, a multi-scale motion vector field at the CTU level is used to predict ideal partition paths via a CNN based on U-Net. Within CTUs, the CNN forecasts the QT depth and MT split decisions for every grid cell. Third, CNN predictions are used by an effective partition pruning technique to omit pointless RDO evaluations. Finally, efficiency and complexity are balanced in an adaptive threshold selection approach. According to experimental results, acceleration under RAGOP32 ranges from 16.5% to 60.2%, with a BD-rate efficiency reduction of 0.44% to 4.59%, which is higher than other options.

The study conducted by Song et al. [35], presented a CNN network and FSVM-based decision technique for quick CU partitioning. First, ideal encoding depths are predicted by a CNN model trained on depth data from inter-frame correlations. To further reduce encoding complexity, FSVM is then utilized to forecast CU partitioning modes within the projected depth range. According to experimental data, the technique achieves a good trade-off between video quality and encoding speed, saving 53.55% of the encoding time while only increasing BDBR by 1.47%.

3.3 Conclusion

Through the application of innovative techniques, all research aimed at reducing time complexity in video compression has produced encouraging results. Interestingly, nevertheless, the bit rate has been seen to increase along with the reduced time complexity. This research will focus on reducing the time complexity as well as maintaining the bit rate as much as possible by using the heuristic approaches.

Chapter 4

Methodology

4.1 Introduction

The introduction of novel coding technologies by Versatile Video Coding (VVC), such as quadtree with nested multi-type tree (QTMT), has significantly improved the efficiency of VVC coding. Nonetheless, this enhancement is accompanied by a notable increase in computational complexity, primarily due to the RDO process used to determine the optimal division for Coding Tree Units (CTUs). This study offers a quick solution for CU partitioning prediction in VVC intra mode. So that the time complexity is decreased and fewer CUs need to go through the RDO process.

Our research is based on the authorized JVET documentation and the applicable VTM-20.2 standard program. Because the VTM allows for both symmetric and asymmetric splits, a CU's height and width might be equal or unequal. The largest possible dimension for a CU is 128×128 as per VTM 20.2 initial configuration, and it needs to be split by QT initially. As MaxBtSize and MaxTtSize both have a value of 32, each of the sub-CUs of size 64×64 can be divided by either QT or remain intact. Therefore, for 64 x 64 CUs, the only issue is whether to split them or nothing. We are simply required to select between the five QTMT divisions for CUs that are no bigger than 32×32. We don't look at the 64×64 split choice in this study since it can result in a significant reduction in encoding efficiency if the division is decided upon directly.

We use a heuristic approach in our research to predict the partitioning of CU. So, we must take into account the correlation within the pixels plus the change in propensity in a column or a row to create a workable algorithm. Responding to this division appears to be an effective approach because distinct textures inside a CU are probable to be segregated into distinct sections. But first, the edge information of the current frame can be obtained by utilizing the Sobel operator.

4.2 Proposed Algorithm

The proposed algorithm contains four steps. The first step is to apply the Sobel operator on the current frame horizontally and vertically and merge the two results to get the edge information. Secondly, we calculate the variance for the 32x32 CU in order to determine whether to divide it any further or skip it. All five partitions will be omitted and a significant amount of computational complexity will be reduced by stopping the CU from being further divided. This is a very crucial step and widely used for partition prediction because homogeneous areas can be detected using the variance. Thirdly, we use the gradient information to make the decision about the QT partition of the CU. If QT is chosen then the rest of the four will be skipped. Finally, we use the variance of variance of all possible sub-CUs to select one of the five partitioning structures of QTMT.

4.2.1 The Four Step Algorithm

The four-step algorithm is described below:

Step One: Generate the edge information

To improve feature extraction from the video frames, edge detection is done using the Sobel operator. The following steps are included in this technique:

1. Horizontal Edge Detection: Compute the gradient of the grayscale video frame along its horizontal (x-) axis as the first step. The derivative is computed in the x-direction by applying the Sobel operation to the video frame. This process captures variations in pixel intensity along rows of the image, emphasizing edges that are primarily horizontal in orientation.

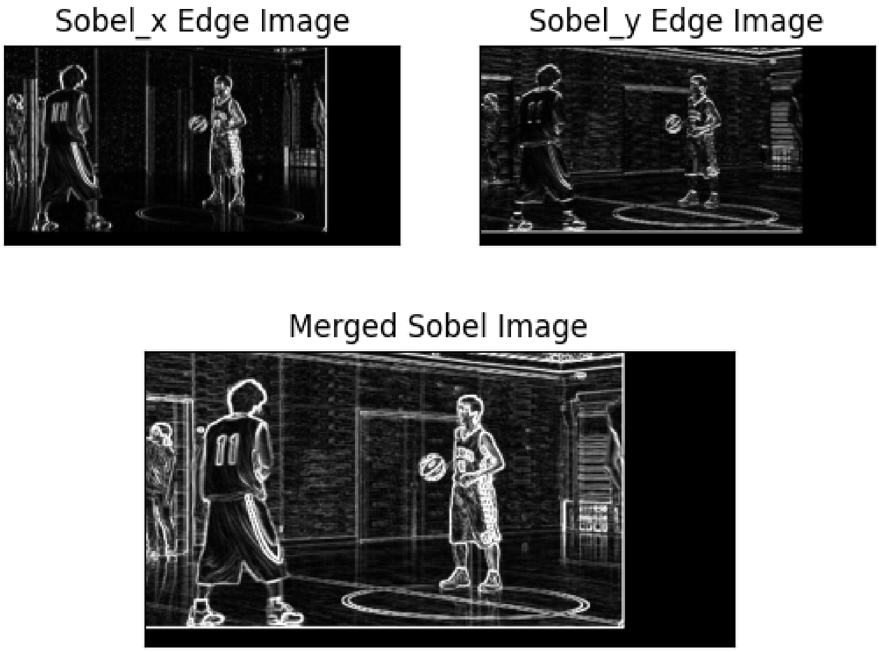


Figure 4.1: Visualization of edged image (frame number 1, POC = 0) of BasketballPass (240p) sequence.

2. Vertical Edge Detection: In a similar manner, the gradient in the vertical (y-axis) direction of the grayscale video frame is computed using the Sobel operator. Using this process, edges with a predominantly vertical orientation are detected, recording variations in pixel intensity along the image's columns.

3. Edge Information Fusion: The resultant gradient images are merged or combined to create a comprehensive edge map when the horizontal and vertical edge detection operations are finished. In order to provide a more comprehensive representation of the edges present in the video frame, this fusion procedure merges the edge information obtained from both horizontal and vertical orientations.

Step Two: Variance-based early skipping of CU

Once the edge frame has been generated, the next step involves assessing the homogeneity of the block. If the block is determined to be homogeneous, there is no need to further divide it. The fundamental concept of this approach was initially introduced in a HEVC acceleration scheme, which is documented in [36]. We have incorporated this approach as a preprocessing step in our implementation. To calculate the variance of the original pixel values within the 32 $\times$ 32 Coding Unit (CU), Equation (4.1) is utilized. If the computed variance is lower than a predetermined threshold (TH1), the CU is classified as having a flat texture.

$$\text{var} = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} (X(i,j) - \mu)^2 \tag{4.1}$$

1. In this scenario:

- W denotes the width of the block,
- H signifies the height of the block,
- $X(i, j)$ denotes the pixel value at position (i, j),
- μ represents the average value across all pixels in the CU

the term var in the equation refers to the computed variance. W and H are both 32 for this method. We compare the calculated variance with a threshold TH1. If the variance is lower than the TH1 we assume that the block is homogeneous and skip any further splitting of the block.

This explains why we utilize variance to decide whether or not to close down the CU division early. While gradient features can vary greatly even in level locations, flat areas often show minor variances. Variance is a better way to detect things that are seemingly smooth, like fur or grass, but may yet contain subtle variances. Consequently, in 32 × 32 CUs, employing variance rather than gradient aids in precisely identifying regions appropriate for early partition termination.

Step Three: Gradient-based QT selection

Subsequently, the summation of gradients for each pixel is computed. The formula (4.2) is applied to determine the gradient horizontally, while the formula (4.3) is utilized to determine the gradient vertically. D_X is derived by summing the absolute values of all D_x components. Likewise, D_Y is obtained by summing the absolute values of all D_y components.

$$\mathrm{D_X} = \sum_{i=1}^{W} \sum_{j=1}^{H} \mathrm{abs}(\mathrm{D_x}(i,j)|) \tag{4.2}$$

$$\mathrm{D_Y} = \sum_{i=1}^{W} \sum_{j=1}^{H} \mathrm{abs}(\mathrm{D_y}(i,j)|) \tag{4.3}$$

The Sobel function is utilized to get the D_x and D_y. The derivative $D_x(i,j)$ is given by:

$$D_x(i,j) = M_{i,j} \times \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{4.4}$$

The derivative $D_y(i,j)$ is given by:

$$D_y(i,j) = M_{i,j} \times \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{4.5}$$

Here,

- M_i, j is the 3×3 matrix of pixels positioned at the location it is calculating at this moment.
- The current center pixel coordinates are represented by i and j within a row and column, respectively.

The decision-making process in step 3 is contingent upon the fulfillment of the three specified conditions:

$$1 < \frac{D_X}{D_Y} > TH2 \quad OR \quad 1 < \frac{D_Y}{D_X} > TH2$$

$$D_X > TH3$$

$$D_Y > TH3$$

In such a scenario, the Quadtree (QT) partition is chosen while skipping the Multi-type Tree (MT) partition.

Step Four: Variance of Variance-based partition selection from QTMT

The mathematical expressions for the five partitions are depicted in the equations (4.6) to (4.10). The variance of each of the 32 × 32 coding unit's sub-coding units for each of the five partitioning conditions will be calculated separately if the 32 × 32 coding unit fails to meet the requirements from steps 2 and 3. We compute the variance of the original pixel values for each sub-coding unit (CU) for each QTMT partition in order to get a set of variances. We next calculate these sets' variances to get five numbers, each of which represents a distinct division. For the present coding unit (CU), we designate the partition with the highest variance value as the only partition. For example, in the BH partition, the left and right halves have values of k of 1 and 2, respectively. The dimensions and average pixel intensity of the k-th sub-CU are denoted as w_k, h_k, and μ_k, respectively. According to the corresponding partition requirements, the variables μ_{QT}, μ_{BT}, μ_{BV}, μ_{TH}, and μ_{TV} represent the average variance values for each sub-CU.

$$\mathrm{var}_{\mathrm{QT}} = \frac{1}{4}(\sum_{k=1}^{4}(\frac{1}{w_k \times h_k}\sum_{i=1}^{w_k}\sum_{j=1}^{h_k}(X(i,j)-\mu_k)^2 - \mu_{\mathrm{QT}}))^2 \tag{4.6}$$

$$\mathrm{var}_{\mathrm{BH}} = \frac{1}{2}\sum_{k=1}^{2}\left(\frac{1}{w_k \times h_k}\sum_{i=1}^{w_k}\sum_{j=1}^{h_k}(X(i,j)-\mu_k)^2 - \mu_{\mathrm{BH}}\right)^2 \tag{4.7}$$

$$\mathrm{var}_{\mathrm{BV}} = \frac{1}{2}\sum_{k=1}^{2}\left(\frac{1}{w_k \times h_k}\sum_{i=1}^{w_k}\sum_{j=1}^{h_k}(X(i,j)-\mu_k)^2 - \mu_{\mathrm{BV}}\right)^2 \tag{4.8}$$

$$\mathrm{var}_{\mathrm{TH}} = \frac{1}{3}\sum_{k=1}^{3}\left(\frac{1}{w_k \times h_k}\sum_{i=1}^{w_k}\sum_{j=1}^{h_k}(X(i,j)-\mu_k)^2 - \mu_{\mathrm{TH}}\right)^2 \tag{4.9}$$

$$\mathrm{var}_{\mathrm{TV}} = \frac{1}{3}\sum_{k=1}^{3}\left(\frac{1}{w_k \times h_k}\sum_{i=1}^{w_k}\sum_{j=1}^{h_k}(X(i,j)-\mu_k)^2 - \mu_{\mathrm{TV}}\right)^2 \tag{4.10}$$

The number of sub-coding units (sub-CUs) connected to each partition structure is represented by the denominator in each term of the first fraction on the right side of the formula. In this instance, k denotes the index of the k-th sub-CU.

The decision criterion in step 4 is as follows: if the maximum value among varQT, varBH, varBV, varTH, and varTV exceeds a certain threshold varn, then the partition corresponding to that maximum variance is selected as the final partition. For instance, if varTH has the highest value among the five variances, then TH is chosen as the ultimate partition.

4.2.2 Algorithm

Data: Fc - Current Frame

Data: Fe_x - Sobel_x(Fc)

Data: Fe_y - Sobel_y(Fc)

Data: Fe - merge(Fe_x, Fe_y)

```
if CU_size = 32 × 32 then
    var_cu ← var(CU);
    for i from 1 to 32 do
        for j from 1 to 32 do
            Dx(i, j) ← Sobel(M(i, j));
            Dy(i, j) ← Sobel(M(i, j));
        end
    end
    DX ← Σ_{i,j} |Dx(i, j)|;
    DY ← Σ_{i,j} |Dy(i, j)|;
    if var_cu < TH1 then
        skip SPLIT;
    end
    else if (1 < DX/DY < TH2 or 1 < DY/DX < TH2) and (DX > TH3 and DY > TH3) then
        Split(QT);
    end
    else
        varQT ← var(var(subCU1), var(subCU2), var(subCU3), var(subCU4));
        varBH ← var(var(subCU1), var(subCU2));
        varBV ← var(var(subCU1), var(subCU2));
        varTH ← var(var(subCU1), var(subCU2), var(subCU3));
        varTV ← var(var(subCU1), var(subCU2), var(subCU3));
        Split(PartitionOfMax(varQT, varBH, varBV, varTH, varTV));
    end
end
```

4.3 Work Flow Diagram

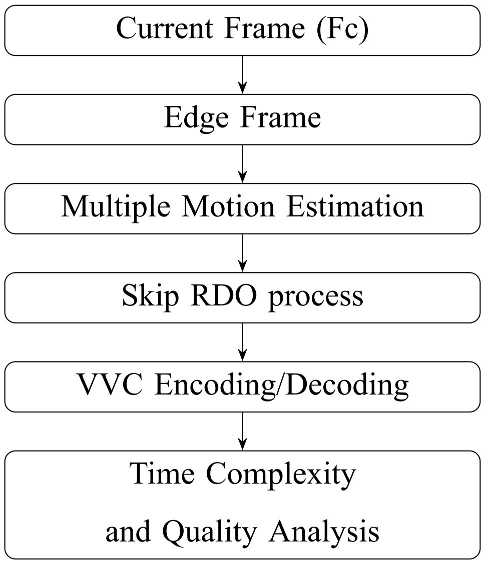


Figure 4.2: Work flow Diagram

The detailed work flow is described below:

1. **Current Frame (Fc):**
    - Load the video sequence.
    - Extract frames sequentially from the video sequence.
2. **Edge Frame Generation:**
    - Implement edge detection methods, such as the Sobel operator, on the present frame (Fc) in order to produce an edge frame.
    - Utilize the Sobel operator on the current frame for calculating both horizontal and vertical gradients.
    - Merge the gradient results to obtain the edge information of the frame.
3. **Multiple Motion Estimation:**
    - Utilize the heuristic information (e.g., variance, gradient, variance of variance) to estimate multiple motions in a coding unit.
    - Employ the four-step fast partitioning algorithm to predict the partition of the coding unit.

4. **Skip Rate-Distortion Optimization (RDO) Process:**

   - Use the prediction from the four-step fast partitioning algorithm to skip the RDO process for certain coding units (CUs).

5. **Versatile Video Coding (VVC) Encoding/Decoding:**

   - Employ the VVC reference software, VTM 20.2, for the encoding and decoding of the video sequences.
   - Utilize the modified VTM 20.2 to encode and decode the video sequences by means of the proposed method.

6. **Time Complexity and Quality Analysis:**

   - Evaluate the efficiency of encoding and decoding by considering time complexity and Bjøntegaard Delta Bit Rate (BDBR).
   - Evaluate the video quality using PSNR.

4.4 Conclusion

This research presents an effective method based on edge detection and variance-based techniques for predicting the partitioning of the Coding Unit (CU) in the intra-mode of Versatile Video Coding (VVC). By using these methods, we maximize encoding efficiency and simplify operations for video compression without sacrificing quality.

Chapter 5

Result & Performance Analysis

5.1 Introduction

Video encoding algorithms are crucial for achieving effective compression without compromising visual quality. This research introduces a novel video encoding technique that emphasizes the prediction of CTU partitioning to improve coding efficiency and decrease computational complexity within the VVC intra-mode. The proposed method incorporates the identification and application of distinct thresholds (TH1, TH2, TH3) in conjunction with strategic GOP and QP configurations to attain optimal encoding outcomes. Through comprehensive experimentation with various video sequences, we assess the effectiveness of our approach by analyzing metrics like PSNR, RD curves, BDBR, and encoding time savings.

5.2 Experimental Setup

5.2.1 Derivation of Threshold

We used three thresholds in our proposed algorithm. The thresholds are TH1, TH2, TH3. To determine the value of these thresholds following equations are used.

$$\text{TH1} = \alpha \times \text{QP} \tag{5.1}$$

$$\text{TH2} = \beta \tag{5.2}$$

$$\text{TH3} = \gamma \tag{5.3}$$

To determine the optimal values of alpha, beta, and gamma, a trial and error method was employed. Initially, two video sequences of distinct types and classes were randomly selected. The evaluation criteria involved analyzing the Bjøntegaard Delta Bit Rate (BDBR) and time savings achieved with each configuration. The objective was to minimize the BDBR (indicative of coding efficiency) while maximizing the time savings associated with the encoding process.

Determination of α :

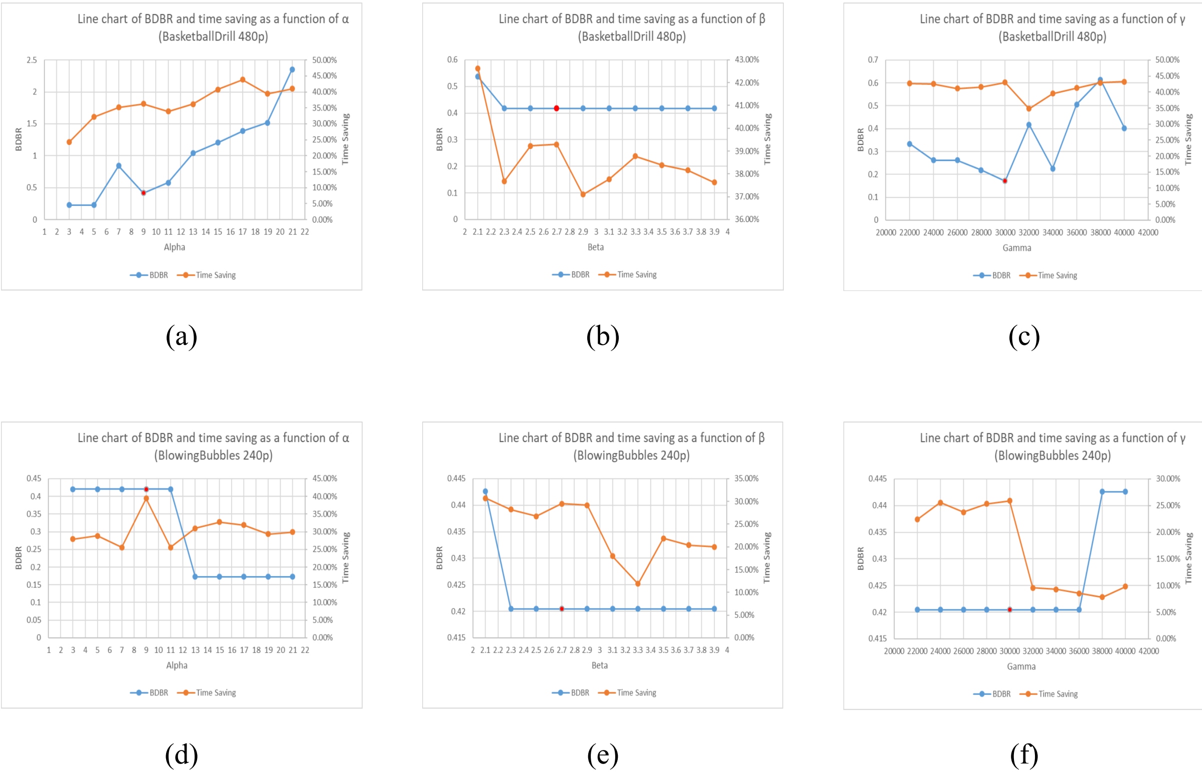


Figure 5.1: Derivation of α, β, γ

To determine the value of alpha, the value of beta and gamma were maintained constant. The value of alpha was iterated over 3 to 21 with an increment of 2 every time. During this iterative process, the video sequences were encoded using the constant beta, gamma value, and varied alpha values. The resulting encoded video streams were then evaluated based on BDBR and time savings. In figure 5.1a and 5.1d, we saw that for both video sequences alpha = 9 was the optimal value. So $\alpha = 9$ was selected.

Determination of β :

After identifying the optimal value of alpha ($\alpha = 9$), the next step involved iterating over the value of beta within a specified range, while keeping the values of alpha and gamma constant. Specifically, beta was varied from 2.1 to 3.9 with an increment of 2. During this iterative process, the video sequences were encoded using the fixed alpha value, constant gamma value, and varied beta values. The resulting encoded video streams were then evaluated based on BDBR and time savings. In figure 5.1b and 5.1e, we saw that for both video sequences beta = 2.7 was the optimal value. So $\beta = 2.7$ was selected.

Determination of γ :

After identifying the optimal value of alpha ($\alpha = 9$) and beta $\beta = 2.7$, the next step involved iterating over the value of gamma within a specified range, while keeping the values of alpha and gamma constant to the optimal value. Specifically, gamma was varied from 22000 to 40000 with an increment of 2000. During this iterative process, the video sequences were encoded using the fixed alpha, beta value, and varied gamma values. The resulting encoded video streams were then evaluated based on BDBR and time savings. In figure 5.1c and 5.1f, we saw that for both video sequences gamma = 30000 was the optimal value. So $\gamma = 30000$ was selected.

5.2.2 GOP and QP Settings

The experimental setup included configurations for the Group of Pictures (GOP) settings and Quantization Parameter (QP) settings as follows:

GOP Settings : The Group of Pictures (GOP) settings define the structure of the video frames in terms of coding units and prediction dependencies. The following GOP settings were utilized for our experiments:

- Intra Mode was Utilized for encoding.
- GOP Size was Set to -1. indicating that each frame is encoded independently without inter-frame prediction.

QP Settings : Quantization Parameter (QP) settings control the quantization levels applied during the video encoding process. The trade-off between compression efficiency and visual quality is influenced by these settings. Our experiments utilized the following QP settings:

- Four distinct Quantization Parameter (QP) values were employed.
- QP = 22, 27, 32, 37

5.2.3 Test Video Sequences

A variety of video sequences were chosen to assess the effectiveness of our video coding algorithms across a range of visual and temporal attributes. Each test sequence was encoded using the experimental configurations described earlier. The test video sequences are described in table 5.1.

Table 5.1: Details of Test Video Sequences

Class	Sequence Name	Resolution	Frame Rate (fps)
B	BasketballPass	416x240	50
	BlowingBubbles	416x240	50
	BQSquare	416x240	60
	RaceHorses	416x240	30
C	BasketballDrill	832x480	50
	PartyScene	832x480	50
	BQMall	832x480	60
	RaceHorses	832x480	30
D	BasketballDrive	1920x1080	50
	Cactus	1920x1080	50
	BQTerrace	1920x1080	60
	ParkScene	1920x1080	24

5.3 Partitioning Analysis

Block partitioning is a crucial aspect of video encoding. It involves dividing video frames into smaller blocks or coding units, which are then independently processed and compressed. The objective of block partitioning is to effectively depict video content while reducing computational complexity and bit rate. Effective block partitioning strategies can significantly impact encoding efficiency, compression ratio, and video quality.

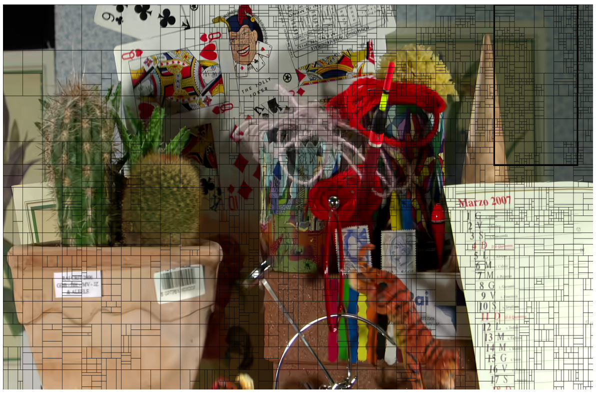
The goal of our proposed algorithm was to predict the CU partitions. Figure 5.2 shows the partition of BasketballDrive 1080p first frame with QP=32. The VTM 20.2 has partitioned many homogeneous CUs while our method has skipped them effectively.

Similarly, for the Cactus 1080p first frame with QP=37, our method has performed more efficient partitioning than the VTM 20.2. Figure 5.3 shows the comparison.

(a) Partition by VTM 20.2.

(b) Partition by our method

Figure 5.2: Partition of BasketballDrive 1080p QP = 32 POC=0



(a) Partition by VTM 20.2

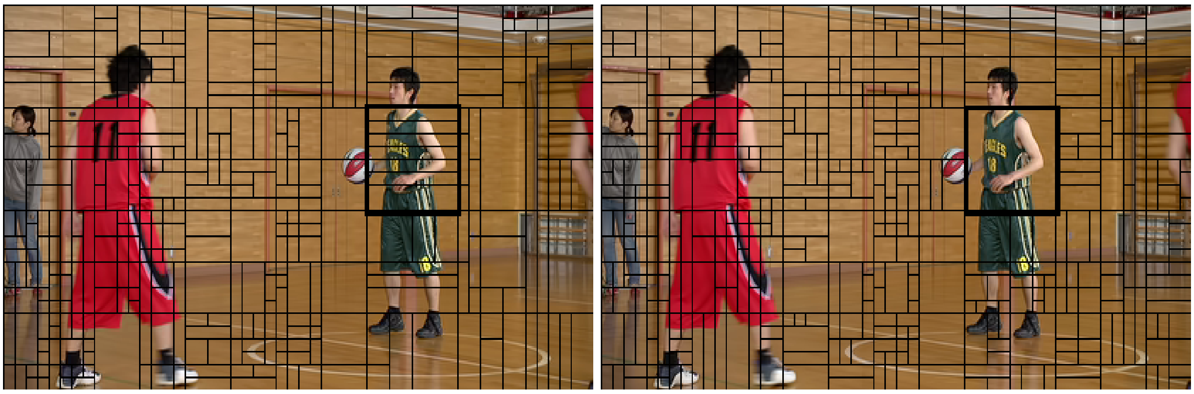
(b) Partition by our method

Figure 5.3: Partition of Cactus 1080p QP=37 POC=0

(a) Partition by VTM 20.2

(b) Partition by our method

Figure 5.4: Partitioning for BQMall 480p QP =27 POC=5



(a) Partition by VTM 20.2

(b) Partition by our method

Figure 5.5: Partitioning for BasketballPass 240p QP =22 POC=0

(a) Encoded/decoded by VTM encoder.

(b) Encoded/decoded by our method

Figure 5.6: Visual Comparison of BQTerrace 1080p sequence.

Figure 5.4 and 5.5 also show that in so many homogeneous areas our method has performed more efficient partition.

5.4 Visual Comparison

An advantageous outcome of our method is the observation that visual comparisons between the original encoder and our heuristic-based approach reveal no discernible difference in video quality. Figure 5.6,5.7,5.8 clearly shows that there is no visually distinguishable difference between the original approach and our approach.

(a) Encoded/decoded by VTM encoder.

(b) Encoded/decoded by our method

Figure 5.7: Visual Comparison of PartyScene 480p sequence.

(a) Encoded/decoded by VTM encoder.

(b) Encoded/decoded by our method

Figure 5.8: Visual Comparison of RaceHorsesD 240p sequence.

5.5 Performance Analysis (PSNR & RD curve)

5.5.1 PSNR

Peak Signal-to-Noise Ratio (PSNR) is a commonly used metric for the objective measurement of video quality. It assesses the accuracy of a compressed video sequence in comparison to the original uncompressed version. PSNR quantifies the ratio between the maximum potential power of a signal and the power of noise that distorts its fidelity. The calculation of PSNR involves the utilization of the following formula:

$$\mathrm{PSNR} = 10 \times \log_{10}\left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}}\right)$$

Within this equation:

- The symbol MAX signifies the highest achievable pixel value within the image or video frame.
- The abbreviation MSE represents the Mean Squared Error calculated between the original and reconstructed frames.

The PSNR metric is highly useful for objectively assessing the quality degradation introduced by compression algorithms. It provides a quantitative measure of visual fidelity, enabling comparison and evaluation of different encoding techniques. A higher PSNR value indicates a superior quality of reconstruction.

The table 5.2 displays the video sequences along with their respective PSNR gains, where the negative values represent a decrease in PSNR after compression or processing compared to the original, uncompressed video data.

The PSNR loss observed in our method is minimal and can be considered negligible. Across the evaluated video sequences, the PSNR decrease ranged from -0.12 dB to -0.002 dB, indicating a marginal impact on the perceived quality. This minor degradation in PSNR values underscores the effectiveness of the proposed method in maintaining high visual fidelity.

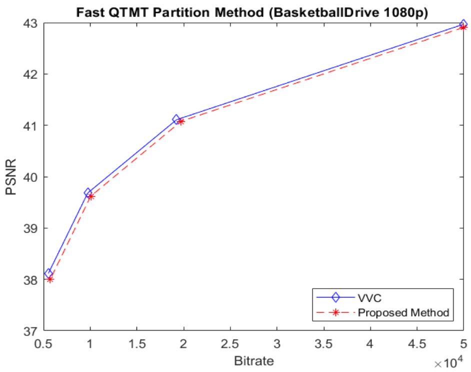
Table 5.2: Video Sequences and PSNR Gains

Video Sequence	**PSNR Gain**
BasketballDrive	-0.11
BQTerrace	-0.06
Cactus	-0.06
Kimono	-0.06
ParkScene	-0.09
BasketballDrill	-0.07
BQMall	-0.05
PartyScene	-0.12
RaceHorses	-0.02
BasketballPass	-0.004
BlowingBubbles	-0.03

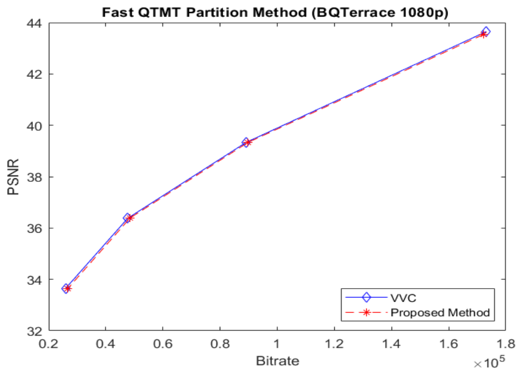
5.5.2 RD Curve

The rate-distortion curve, also known as the RD curve, is a graphical representation of the relationship between the bitrate and the quality of the encoded video. It is commonly used in video encoding to analyze the performance of various encoding configurations and codecs. By plotting the bitrate on the x-axis and the PSNR on the y-axis, the RD curve offers valuable insights into the impact of encoding parameters on compression efficiency and video quality. This fundamental tool aids in understanding the trade-off between the rate (bitrate) and the distortion (video quality) in video encoding.

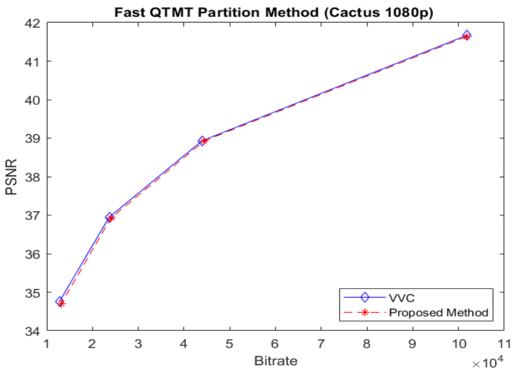
In our study, we extensively utilized the Rate-Distortion (RD) curve to analyze the performance of our proposed encoding method. By examining the shape and characteristics of the RD curve, we can confidently conclude that our method demonstrates strong performance in relation to the trade-off between compression efficiency and visual quality, it is necessary to consider alternative approaches. Some of the RD curves are shown in figure 5.9, 5.10, 5.11.



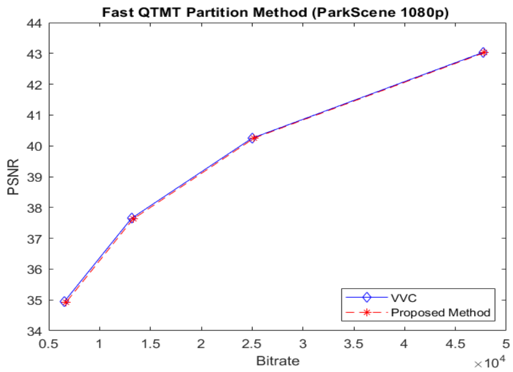
(a) BasketballDrive



(b) BQTerrace

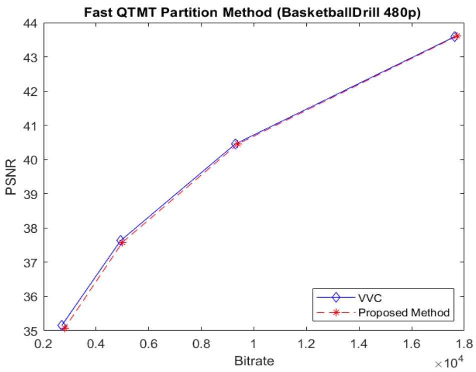


(c) Cactus

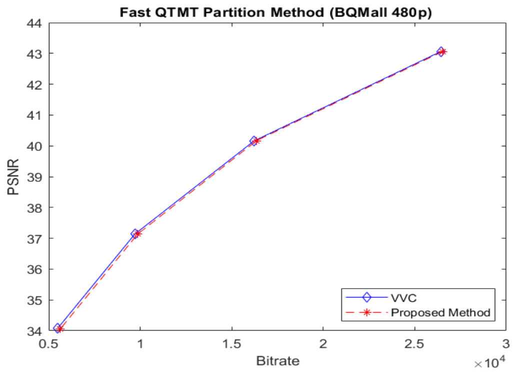


(d) ParkScene

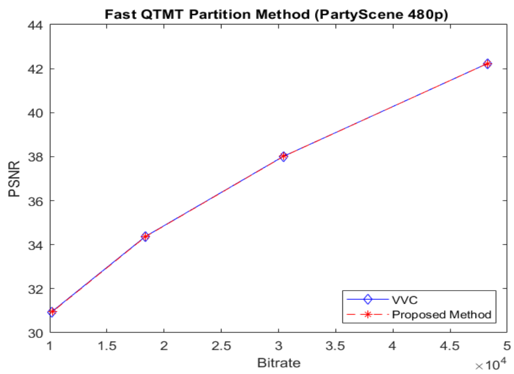
Figure 5.9: RD curve of class B



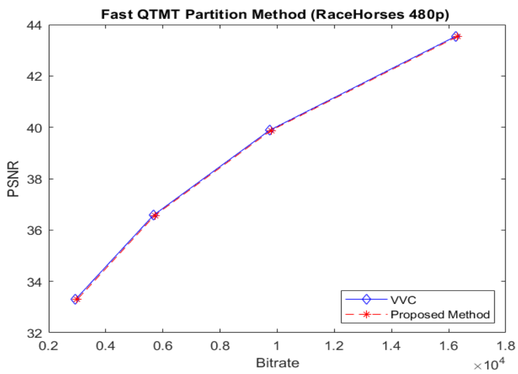
(a) BasketballDrill



(b) BQMall

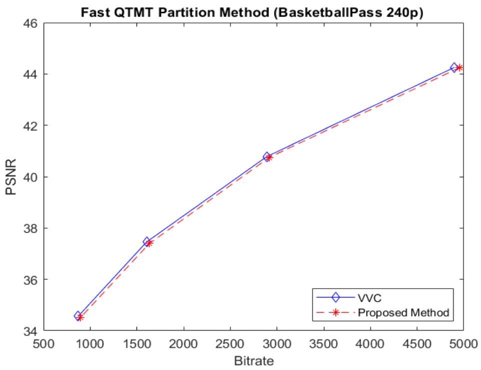


(c) PartyScene

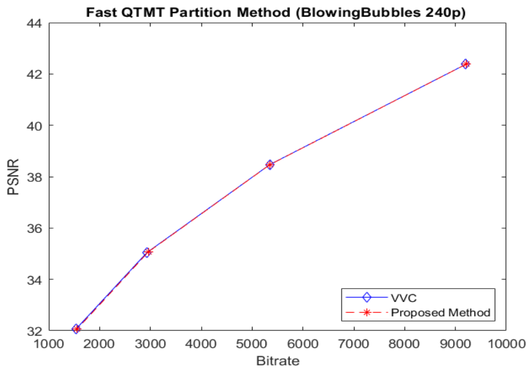


(d) RaceHorsesC

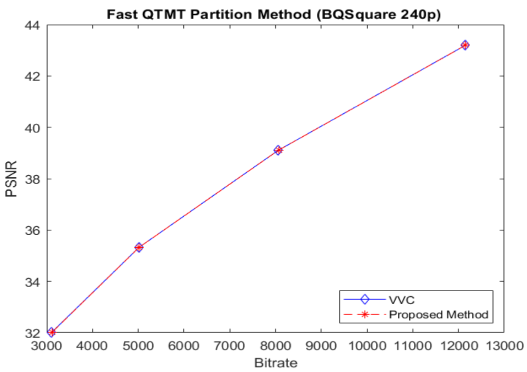
Figure 5.10: RD curve of class C



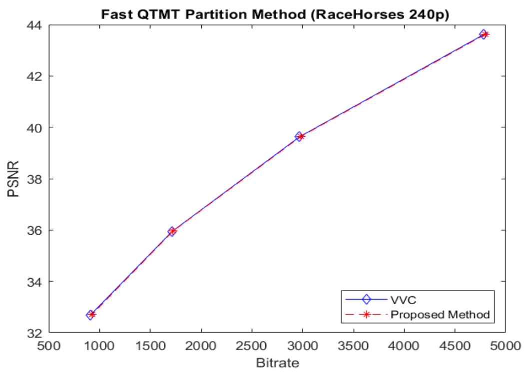
(a) BasketballPass



(b) BlowingBubbles



(c) BQSquare



(d) RaceHorsesD

Figure 5.11: RD curve of class D

5.6 Performance Analysis (BDBR & Time Savings)

BDBR, an acronym for ”Bjøntegaard Delta Bit Rate” (BDBR), was coined in honor of its developer Gisle Bjøntegaard [37]. This metric is employed to evaluate the coding efficiency of various video codecs or setups. It quantifies the bitrate reduction (or escalation) attained by one codec in comparison to another at an equivalent quality level, typically assessed through PSNR (Peak Signal-to-Noise Ratio). A negative BDBR value signifies a bitrate reduction (enhancement), whereas a positive value indicates a rise in bitrate for the same quality level.

To enhance the efficiency of video encoding, a critical analysis of encoding time savings (ΔET) was conducted using a specific formula that compares the efficacy of the suggested approach against the standard VTM encoder VTM 20.2 across various quantization parameters (QP_k). The formula calculates the average time saving by assessing the reduction in encoding time achieved with the modified encoder relative to the standard encoder.

The equation employed for time savings calculation is:

$$\Delta ET = \frac{1}{4} \times \sum_{Q\mathrm{P}_k \in S} \frac{T_{\mathrm{VVC}}(Q\mathrm{P}_k) - T_{\mathrm{PE}}(Q\mathrm{P}_k)}{T_{\mathrm{VVC}}(Q\mathrm{P}_k)}$$

- S = {22, 27, 32, 37}
- $T_{\mathrm{VVC}}(Q\mathrm{P}_k)$ is the time needed for VTM encoder.
- $T_{\mathrm{PE}}(Q\mathrm{P}_k)$ is the time needed for modified VTM encoder.

The outcome of ΔET provides insights into the efficiency enhancement achieved by the proposed method in terms of reducing coding time contrast to the standard VTM 20.2 encoder.

5.6.1 Comparison with VTM 20.2

Table 5.3 shows the encoding decoding efficiency of our approach compared to the VVC reference software VTM 20.2. Our method can save 48.17 % of encoding time at a cost of only 1.38% BDBR loss.

Table 5.3: BDBR and Time Save Compared to VVC VTM 20.2

Class	**Sequence**	**BDBR (%)**	**Time Save (%)**
B	BasketballDrive	4.93	67.93
	BQTerrace	1.19	44.32
	Cactus	1.86	50.31
	ParkScene	1.39	47.51
C	BasketballDrill	1.94	46.04
	BQMall	1.28	43.95
	PartyScene	0.18	38.08
	RaceHorses	0.91	49.65
D	BasketballPass	2.19	43.03
	BlowingBubbles	0.41	55.35
	BQSquare	0.33	44.89
	RaceHorses	0.05	46.95
Average		**1.38**	**48.17%**

5.6.2 Comparison with other fast method

The proposed algorithm is compared with the approach presented in [38] based on BDBR and Time Savings (TS) metrics. able 5.4 provides a detailed comparison of these metrics for various video sequences across different classes.

Based on the comparison presented in Table 5.4, it is evident that the proposed method outperforms F. Chen's approach [38]. The difference in time savings between the proposed method and F. Chen's approach is minimal and almost negligible. While F. Chen's approach achieves a slightly higher time-saving rate (51.23%) than the proposed method (48.17%), the proposed method offers distinct advantages in compression efficiency (BDBR). The proposed method demonstrates superior compression performance with a lower average BDBR of 1.38% compared to F. Chen's method at 1.57%. This signifies that the proposed method excels in preserving video quality and achieving higher compression ratios, making it an optimal choice for applications prioritizing compression efficiency alongside competitive time-saving capabilities.

Table 5.4: Comparison of BDBR and Time Savings (TS)

Class	**Video Sequence**	**F. Chen [38]**		**Proposed**	
		BDBR (%)	TS (%)	BDBR (%)	TS (%)
B	BasketballDrive	2.25	64.01	4.93	67.93
	BQTerrace	2.07	56.07	1.19	44.32
	Cactus	1.95	56.66	1.86	50.31
	Kimono	1.90	63.87	1.39	47.51
C	BasketballDrill	2.01	48.19	1.94	46.04
	BQMall	2.15	55.23	1.28	43.95
	PartyScene	0.60	45.73	0.18	38.08
	RaceHorses	1.16	48.39	0.91	49.65
D	BasketballPass	2.33	45.85	2.19	43.03
	BlowingBubbles	0.77	41.56	0.41	55.35
	BQSquare	0.81	46.06	0.33	44.89
	RaceHorses	0.86	43.17	0.05	46.95
Average		**1.57%**	**51.23%**	**1.38%**	**48.17%**

5.7 Conclusion

To sum up, our suggested approach for video encoding shows a lot of benefits over traditional methods. Through the examination of RD curves, PSNR measures, and encoding time savings, we have verified the efficacy of our methodology. The outcomes demonstrate significant encoding time reductions combined with negligible quality degradation (as determined by BDBR and PSNR). Our approach is a viable alternative for video encoding tasks when preserving high-quality standards is crucial while minimizing computational resources, because of its efficient compression and time-saving features.

Chapter 6

Conclusion & Future Works

6.1 Introduction

Video compression plays a crucial role in numerous multimedia applications by reducing the volume of video data that must be transmitted and stored. The advancement of video codecs such as the Versatile Video Coding (VVC) standard has enabled higher resolution and improved quality video content to be achieved with reduced data space requirements. However, these advancements have introduced a notable challenge in the form of the computational complexity associated with video encoding. The issue of time complexity in video compression has been extensively studied over the years. One effective strategy to streamline the encoding process involves developing efficient algorithms for predicting and optimizing coding unit (CU) partitions.

A number of the articles in the literature review provide novel approaches to reducing the complexity of encoding times while preserving compression effectiveness. For example, Zhang et al. presented a successful texture analysis-based CU partitioning method that significantly reduced coding time by 52.3% while having small effect the bitrate. Similar results were obtained with marginal increases in Bjontegaard Delta Bit Rate (BDBR) and up to 30% complexity reduction using a multitask learning framework for intra-mode decision-making by Naima Zouidi et al. These strategies maximize CU partitioning choices by utilizing heuristic, machine learning, and neural network techniques. They show encouraging outcomes in striking a balance between computing economy and video compression performance.

Our research adds to the continuing efforts to reduce the computational complexity of VVC. Heuristic-based fast prediction of CU partition techniques exhibits the potential to reduce the processing load related to video encoding while preserving good compression efficiency. Our work proposes a heuristic-based fast method for predicting CTU partitioning in the VVC intra-mode. The main goal was to bypass the traditional RDO process for 32*32 blocks, which consumes a huge amount of encoding time. By leveraging variance-based early termination and gradient-based QT selection, we effectively eliminate unnecessary computations and favor optimal partition modes. Additionally, our method incorporates variance of variance metrics for selecting the best partition among the QTMT partitions. Through comprehensive experimental evaluations, we demonstrate that our method provides a significant reduction of 48.17% in encoding time, with a tolerable rise of 1.38% in BDBR compared to the VVC reference software VTM 20.2.

6.2 Limitations

Our main goal in this study was to reduce computational complexity in video compression by creating a quick approach for predicting Coding Tree Unit (CTU) partitions in VVC intra-mode. We assessed our method on a collection of twelve different video sequences provided by the Joint Video Exploration Team (JVET) [39], using them to compare our method's performance with VVC's reference program, VTM 20.2. The limitation of our method is that while it significantly reduces encoding time by 48.17%, there is a corresponding increase in the BDBR (1.38%). This trade-off between time complexity reduction and slight degradation in video quality (as indicated by BDBR) should be considered when evaluating the applicability of our approach in practical scenarios. Achieving optimal encoding efficiency while maintaining video quality at its highest level remains a significant challenge that necessitates further refinement and optimization of our method.

6.3 Future Work

Our study presents a heuristic-based approach for estimating CTU partitions in VVC intra-mode, focusing on reducing computational complexity while acknowledging a trade-off with video quality. In future work, our method could be extended and refined through several approaches. Promising possibilities include incorporating machine learning techniques for adaptive decision-making, dynamically optimizing parameters depending on content features, and investigating other heuristic criteria beyond variance and gradient-based methods.

The utilization of hybrid approaches, which integrate heuristic methods with adaptive algorithms or deep learning, might be advantageous in enhancing encoding efficiency while upholding video quality standards.

6.4 Conclusion

This study focuses on leveraging heuristic information from video frames to estimate motion and predict CU partitioning, aiming to reduce the computational complexity of VVC. Our approach shows promising potential for time reduction while preserving video quality. Research and innovation in this area are essential to expanding video encoding technologies and enhancing overall multimedia processing efficiency as multimedia applications continue to change. I plan to delve deeper into this field of research, exploring additional techniques and methodologies to further enhance video compression efficiency while maintaining high-quality video output.

REFERENCES

[1] I. E. Richardson, *The H.264 Advanced Video Compression Standard*. Wiley, 2nd ed., 2004.

[2] M. Saldanha, G. Sanchez, C. Marcon, and L. Agostini, *Versatile Video Coding (VVC): Machine Learning and Heuristics*. Synthesis Lectures on Engineering, Science, and Technology, Springer, 2022.

[3] I. Siqueira, G. Correa, and M. Grellert, "Complexity and coding efficiency assessment of the versatile video coding standard," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2021.

[4] I. T. U. (ITU), "Itu's role in shaping the ict ecosystem," *ITU News Magazine*, 2020.

[5] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[6] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the Versatile Video Coding (VVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 3896–3918, 2020.

[7] International Organization for Standardization (ISO), "Mpeg standards," *ISO/IEC JTC 1/SC 29/WG 11*, 2021.

[8] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[9] I. T. U. (ITU), "Itu-t h.266/vvc standard: Versatile video coding," *ITU-T*, 2020.

[10] I. Cisco Systems, "Cisco visual networking index predicts global annual ip traffic to exceed three zettabytes by 2021," June 2017.

[11] X. Qian, Y. Zeng, W. Wang, and Q. Zhang, "Co-saliency detection guided by group weakly supervised learning," *IEEE Transactions on Multimedia*, vol. 1, p. 1, 2022.

[12] A. Segall, V. Baroncini, J. Boyce, J. Chen, and T. Suzuki, "Joint call for proposals on video compression with capability beyond hevc," Tech. Rep. JVET-H1002, Joint Video Exploration Team (JVET), October 2017.

[13] International Telecommunication Union (ITU), "Joint Video Exploration Team (JVET) - ITU-T Study Group 16," 2020.

[14] "Versatile video coding, recommendation itu-t h.266 and iso/iec 23090-3 (vvc)," tech. rep., ITU-T and ISO/IEC JTC 1, July 2020.

[15] C.-Y. Hsu, C.-M. Chen, T.-C. Chuang, H.-Y. Huang, S.-Y. Hsiang, C.-Y. Chen, M.-C. Chiang, C.-H. Lai, C.-M. Tsai, Y.-P. Su, Z. Lin, Y.-W. Hsiao, J. Klopp, I.-C. Wang, Y.-C. Huang, and S.-Y. Lei, "Description of sdr video coding technology proposal by mediatek," Tech. Rep. JVET-J0018, Joint Video Exploration Team (JVET), April 2018.

[16] Y. Ye, J. M. Boyce, and P. Hanhart, "Omnidirectional 360° video coding technology in responses to the joint call for proposals on video compression with capability beyond hevc," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 5, pp. 1241–1252, 2020.

[17] "Merriam-webster dictionary." [Accessed 2024-04-19].

[18] R. Steinmetz and K. Nahrstedt, *Multimedia Fundamentals: Media Coding and Content Processing*. Springer, 2004.

[19] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the versatile video coding (vvc) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 3896–3918, 2020.

[20] L. L. Prince, "Roundhay garden scene," 1888. [Accessed 2024-04-19].

[21] C. Davenport, "Youtube video sizes chart." Online image, 2018. Accessed on April 19, 2024.

[22] ITU-T, "ITU-T Recommendation BT.500-11," ITU-T Recommendation BT.500-11, ITU-T, 2002. Methodology for the subjective assessment of the quality of television pictures.

[23] W. Menasri and A. Skoudarli, "Performance comparison of throughput between avc, hevc, and vvc hardware cabac decoder," *Journal of Real-Time Image Processing*, vol. 20, no. 26, 2023.

[24] D. Zhang and Q. Li, "An efficient cu partition algorithm for vvc intra coding," *Journal of Physics: Conference Series*, vol. 1815, p. 012006, feb 2021.

[25] H. Zhao, S. Zhao, X. Shang, and G. Wang, "A fast algorithm for vvc intra coding based on the most probable partition pattern list," *Applied Sciences*, vol. 13, no. 18, 2023.

[26] Y.-H. Tsai, C.-R. Lu, M.-J. Chen, M.-C. Hsieh, C.-M. Yang, and C.-H. Yeh, "Visual perception based intra coding algorithm for h.266/vvc," *Electronics*, vol. 12, no. 9, 2023.

[27] S. Zhao, X. Shang, G. Wang, and H. Zhao, "A fast algorithm for intra-frame versatile video coding based on edge features," *Sensors*, vol. 23, no. 13, 2023.

[28] N. Zouidi, A. Kessentini, W. Hamidouche, N. Masmoudi, and D. Menard, "Multitask learning based intra-mode decision framework for versatile video coding," *Electronics*, vol. 11, no. 23, p. 4001, 2022.

[29] M. Li, Z. Wang, and Q. Zhang, "Fast cu size decision and intra-prediction mode decision method for H.266/VVC," *EURASIP Journal on Image and Video Processing*, vol. 2024, no. 1, pp. 1–21, 2024.

[30] Y. Li, Z. He, and Q. Zhang, "Fast decision-tree-based series partitioning and mode prediction termination algorithm for h.266/vvc," *Electronics*, vol. 13, no. 7, 2024.

[31] D. Luo, S. Xiong, X. He, and et al., "Block-correlation-based intra prediction for VVC," *Multimedia Tools and Applications*, vol. 82, pp. 23635–23653, 2023.

[32] F. Belghith, B. Abdallah, S. Ben Jdidia, and et al., "CNN-based ternary tree partition approach for VVC intra-QTMT coding," *Signal, Image and Video Processing*, vol. 18, pp. 3587–3594, 2024.

[33] H. Li, P. Zhang, B. Jin, and Q. Zhang, "Fast cu decision algorithm based on cnn and decision trees for vvc," *Electronics*, vol. 12, no. 14, 2023.

[34] Y. Liu, M. Riviere, T. Guionnet, A. Roumy, and C. Guillemot, "Cnn-based prediction of partition path for vvc fast inter partitioning using motion fields," 2023.

[35] W. Song, C. LI, and Q. Zhang, "Fast cu partitioning algorithm for vvc based on cnn and fsvm," *IEEE Access*, vol. 12, pp. 52150–52160, 2024.

[36] T. Mallikarachchi, A. Fernando, and H. K. Arachchi, "Efficient coding unit size selection based on texture analysis for HEVC intra prediction," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, (Chengdu, China), pp. 1–6, July 2014.

[37] G. Bjøntegaard, "Calculation of average psnr differences between rd-curves," 2001.

[38] F. Chen, Y. Ren, Z. Peng, *et al.*, "A fast CU size decision algorithm for VVC intra prediction based on support vector machine," *Multimed Tools Appl*, vol. 79, pp. 27923–27939, 2020.

[39] Joint Video Exploration Team (JVET) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, "JVET common test conditions and software reference configurations." 7th Meeting, Torino, Italy, 13–21 July 2017.