ISSN: 3049-1479 (Online)

Dynamic Mode Parameter Control for VVC: An Approach to Balancing Compression Efficiency and Computational Complexity

Devendrakumar Mangroliya[1], Rahul N. Gonawala[2]

MTech Student, Electronics and Communication Engineering Department,
Bhagwan Mahavir College of Engineering and Technology, Surat, Gujarat, India[1]
Assistant Professor, Electronics and Communication Engineering Department,
Bhagwan Mahavir College of Engineering and Technology, Surat, Gujarat, India[2]

mangroliyadev92@gmail.com[1], rg.surat@gmail.com[2]

Abstract: Video compression, crucial for efficient multimedia storage and transmission, benefits from the Versatile Video Codec (VVC), excelling in compression efficiency. However, rising resolutions and complexities strain computational resources. This paper delves into dynamic mode parameter control within VVC to optimize compression and complexity balance. By adapting VVC's mode parameters based on content characteristics, network conditions, and quality requirements using machine learning and scripting algorithms, we optimize encoding parameters in real-time. Through extensive experimentation, we demonstrate that this approach significantly enhances VVC's flexibility for high-resolution streaming and real-time communication, while also shedding light on the potential of integrating machine learning into video coding pipelines for adaptive compression. Overall, this research not only advances understanding of VVC but also offers a promising solution for addressing the evolving demands of multimedia applications where balancing compression efficiency and computational complexity is crucial.

Key Words:VVC, Compression, Efficiency, Computational, Complexity.

I. INTRODUCTION

In the realm of multimedia content storage and transmission, efficient video compression stands as a cornerstone. The Versatile Video Codec (VVC) emerges as a cutting-edge solution, promising substantial gains in compression efficiency over its predecessors [1]. However, as video resolutions and complexities surge, so do the computational demands, potentially posing barriers to widespread adoption. This paper explores a novel approach– Dynamic Mode Parameter Control – within the VVC framework, aimed at striking an optimal balance between compression efficiency and computational complexity. By dynamically adjusting VVC's mode parameters based on various factors including content characteristics, network conditions, and quality requisites, we aim to optimize encoding parameters in real-time. This introduction sets the stage for our investigation into the feasibility and efficacy of this adaptive strategy in enhancing VVC's flexibility for diverse multimedia applications, from high-resolution streaming to real-time communication, while also considering the integration of machine learning techniques to further refine the compression process.

A. Mode Parameters of VVC

VVC, the Versatile Video Codec, encompasses a comprehensive set of mode parameters crucial for efficient video compression. These parameters include intra prediction modes, determining how pixels within a coding block are predicted spatially; inter prediction modes, exploiting temporal redundancy by referencing previously coded frames; transform coding methods like DCT or DWT, converting pixel values into frequency-domain coefficients; quantization techniques, adjusting coefficient precision to balance compression and image quality; and entropy coding schemes such as arithmetic coding or CABAC, further compressing data by assigning shorter codewords to more probable symbols. Together, these mode parameters govern the delicate balance

https://doi.org/10.63766/spujstmr.24.000021

between compression efficiency and computational complexity in VVC encoding, with dynamic adjustment offering adaptability to varying content characteristics and encoding conditions, ensuring optimal performance across diverse multimedia applications.

The partitioning step divides a video frame into non-overlapping blocks in order to prepare it for the different encoding decisions, such as prediction, transformation, and quantization. In video coding standards such as HEVC [3].

B. Compression Efficiency and Computational Complexity

Video compression efficiency is paramount in digital multimedia technology, aiming to reduce data while preserving perceptual quality. Key metrics like Peak Signal-to-Noise Ratio (PSNR) gauge compression quality, with higher PSNR indicating better efficiency. However, PSNR has limitations, prompting the use of metrics like Structural Similarity Index (SSI). Computational complexity, referring to resources needed for compression, is crucial for real-time applications. Encoding/Decoding Time and Algorithmic Complexity are key metrics. Efficient memory usage and algorithm scalability also impact computational complexity. Balancing compression efficiency with computational feasibility ensures the development of practical, widely applicable video compression technologies, meeting evolving digital media demands.

II. RELATED WORKS

In recent research on VVC codec, multiple papers have been proposed to enhance video compression efficiency and reduce computational complexity. Notable contributions from Papers [2], [4], [5], and [6] shed light on distinct aspects of this landscape.

In "Rate Control Technology for Next Generation Video Coding Overview and Future Perspective" [2], the effectiveness of rate control was demonstrated through tests on video sequences of various resolutions using reference software such as JM19.0, HM16.26, and VTM18.0. The experiments, utilizing YUV video sequences from official recommended sources, set the target bit rate to 100,000, with 50 input video frames and encoding configurations including LDB, LDP, and RA. Notably, only JM software used default configuration and lacked support for higher resolution videos. The chosen SDR video sequences ranged from RaceHorses (416 × 240) to Traffic (2560 × 1600). Results indicated significantly higher average PSNR values for HM and VTM compared to JM. The research transition from HEVC to VVC is evident, with HM reference software still widely used. However, VVC reference software research is in its infancy, typically focusing on hierarchical structure and various algorithms to achieve higher negative BD-rate values and lower RBE values. Notably, experiments often compare HM and VTM, with frame and CTU levels being more studied than GOP level. While some experiments achieve notable performance improvements within acceptable bit rate accuracy ranges, bit allocation accuracy in RA configurations varies considerably.

In the paper "Adaptive intra-refresh for low-delay error-resilient video coding" [4], we introduce an efficient model for selecting intra-refresh cycle sizes based on network packet loss rates. Our linear model adapts cycle sizes without relying on sequence-specific parameters, enhancing flexibility. Experimental results validate the algorithm's effectiveness. Currently, one cycle size applies to entire video sequences regardless of scene changes, prompting future work on frequent parameter updates for adaptive scene-based selection. Additionally, we aim to extend this approach to the HEVC standard.

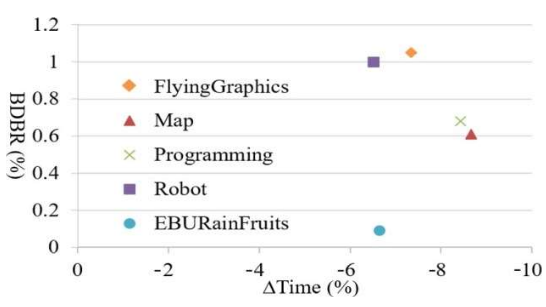

Fig.1:BDBR and Time of the proposed algorithm under LD configuration

In [5], Yu and Jung introduce an adaptive perceptual quantizer (APQ) for high dynamic range (HDR) video compression with HEVC Main 10 Profile. The proposed technique adjusts the transfer function (TF) based on HDR content, mapping luminance to luma adaptively. By extracting maximum and minimum luminance values, a scaling factor ratio is obtained to adjust the PQ-TF, resulting in APQ-TF. This ratio, adaptive to HDR content, is updated per-frame and encoded as metadata. In decoding, the ratio is used to adjust the inverse PQ-TF, yielding inverse APQ-TF. Compared to PQ-TF, APQ-TF covers a larger range of luma values, reducing quantization distortion, and maintains better perceptual uniformity, enhancing color retention. Flickering artifacts from adaptive quantization are mitigated using a low-pass filter. Experimental results show that HDR video coding with APQ-TF saves more bitrate while preserving perceptual uniformity. Future work includes exploring adjustments to other parameters in PQ-TF and investigating the relationship between APQ and DRA, with consideration for combining APQ-TF and DRA.

The paper [6] proposes a machine learning-based framework for fast intra mode decision in HEVC Screen Content Coding (SCC). Unlike traditional methods, which check IBC and PLT modes for SCBs, the framework utilizes decision trees with dynamic features to make mode decisions separately. This sequential arrangement reduces computational complexity by 47.62% on average, with a minimal increase in BDBR (1.42%). Future work may explore fast SCC encoding algorithms based on CNNs, despite their high computational complexity. Strategies like increasing stride size and designing multi-output CNNs could mitigate this drawback, with this paper serving as a baseline for future CNN approaches.

III. METHODOLOGY

Dynamic control of mode parameters in VVC (Versatile Video Coding) encoding is a crucial technique used to optimize compression efficiency while adapting to varying content characteristics and network conditions.

A. Content Analysis

Content analysis in the context of video encoding, such as VVC (Versatile Video Coding), involves analyzing the visual and temporal characteristics of the video content to inform encoding decisions.

i. Algorithm of Motion Detect

The key part of the logic involves computing the absolute difference between pixel values of corresponding pixels in consecutive frames, converting the difference to grayscale, summing the intensities, and comparing the total difference to a motion threshold.A simplified mathematical expression to represent the core logic of the motion detection could be expressed as follows:

Let I_t be the intensity matrix of the current frame,$I_{(t-1)}$ be the intensity matrix of the previous frame and

ΔI_t is the absolute intensity difference matrix:$\Delta I_t = | I_t - I_{(t-1)} |$

Let ΔI_{gray} be the grayscale representation of ΔI_t ,and Total Different be the sum of pixel intensities in ΔI_{gray}:

$\Delta I_{gray} = \text{ConvertToGray}(\Delta I_t)$

$\text{TotalDiff} = \sum(i,j) \Delta I_{gray} (i,j)$

Finally, the motion detection decision is based on comparing TotalDiff with a threshold:

MotionDetected = TotalDiff > motionThreshold

ii. Motion Analysis

Motion Complexity: Measuring the complexity of motion helps decide whether to use different block sizes, prediction modes, and reference frames.

Motion Estimation: Motion analysis identifies areas of motion within a frame. Algorithms detect moving objects or regions, which helps determine where motion compensation may be applied.

B. Encoder Parameters affected by Content Analysis

i. Quantization Parameters (QPs)**:** Content analysis helps determine the complexity of different regions in a video frame. In more complex regions with fine details, lower quantization levels (smaller QP values) may be chosen to preserve quality. In less complex regions, higher quantization levels may be used to achieve more significant compression. Higher motion complexity, high texture detail, or intricate patterns may lead to lower QP values.

ii. Prediction Modes (Intra/Inter)**:** The analysis of motion complexity and scene changes informs the choice between intra-frame and inter-frame prediction modes. Intra prediction is often used for keyframes or scenes with minimal motion, while inter prediction is efficient for scenes with motion. Scenes with high motion complexity may benefit from inter prediction, while static scenes may use intra prediction.

iii. GOP (Group of Pictures) Structure*:* The duration of shots, scene changes, and temporal characteristics influence the choice of GOP structure. Shorter GOPs may be used for fast-changing scenes, while longer GOPs can be efficient for steady scenes.

C. Network-Aware Adaptation

Dynamic control methods may also consider the network conditions during video transmission. For example, in streaming applications, the system could dynamically adjust parameters based on available bandwidth, upload speed, latency, or other network characteristics to optimize video quality while maintaining a consistent streaming experience. The available bandwidth of the network is a critical factor in video streaming. Network-aware adaptation involves monitoring the current bandwidth and dynamically adjusting video encoding parameters, such as bitrates, to match the available network capacity. This helps prevent issues like buffering or low video quality due to insufficient bandwidth.

Bandwidth is commonly calculated as the amount of data that can be transmitted through a network connection in a given amount of time. The formula for calculating bandwidth is:

Bandwidth = Amount of Data/Time

i. Encoder Parameters affected by Network Bandwidth

1.1 Bitrate Control Parameters**:** The available network bandwidth influences the choice of target bit rate for video encoding. Bitrate control parameters determine the amount of data allocated to represent each second of video. In situations of limited bandwidth, lower target bit rates may be chosen to prevent buffering and ensure smooth playback. In low-bandwidth conditions, the encoder may use a lower target bit rate to generate a more bandwidth-friendly video stream.

1.2. Quantization Parameters (QPs):The quantization process controls the trade-off between video quality and compression efficiency. In low-bandwidth scenarios, higher quantization levels (larger QP values) are typically used to reduce the amount of data transmitted over the network. Higher QP values may be employed to achieve more aggressive compression and lower bit rates.

ii. System Resource Availability

System resource availability plays a crucial role in the implementation of Dynamic Mode Parameter Control (DMPC) in video compression. DMPC involves adapting encoding parameters in real-time based on changing conditions, and the availability of system resources can significantly influence the effectiveness of dynamic adjustments.

iii. Encoder Parameters affected by System Resource

1.1 Parallel Processing**:** Leverage multi-core processors and parallel processing capabilities to distribute the encoding workload across multiple threads or cores. This can significantly improve encoding speed and efficiency, especially on modern hardware architectures.

ISSN: 3049-1479 (Online)

***1.2 Thread Management*:** Increasing the number of threads can lead to faster encoding but may also increase system resource utilization.

1.3 Buffer Management: Optimize buffer sizes and configurations to match the available system memory. Proper buffer management helps prevent memory-related bottlenecks during the encoding process.

IV. EXPERIMENTAL SETUP

In the quest for an algorithm that can discern the authenticity of video content analysis, the experimental setup forms the backbone of our endeavor.

A. Hardware and Software Configuration

***i. The simulations are performed using*:**

The encoding experiments were conducted on a system equipped with an Intel® Core™ i7-8650U CPU running at 1.90 GHz. processing speed with maximum clock speed around 2112 MHz and using Microsoft windows Machine operate on x64 bit architecture systems with RAM (Random Access Memory) capacity of 16GB.OpenCV (Open-Source Computer Vision Library) is a comprehensive open-source computer vision and machine learning software library that has gained widespread popularity for its versatility and extensive functionality.Emgu.CV serves as a .NET wrapper for OpenCV, providing seamless integration and enabling developers to utilize OpenCV functionalities directly in C# projects.

ii. Test Sequences

Four HD video sequences in 8-bit YCbCr 4:2:0 format, obtained from [7], with a resolution of 1280×720, are selected as the test dataset. Each sequence differs in duration and total frame count. Table I provides an overview of the sequences, including their names, and parameters like resolutions, frame rates, number of frames, and bit depth.

TABLE I

DATA SET FOR TESTING ON CONTENT ANALYSIS

Sequence Name	Frames	Resolution	Bit Rate
FVDO_Freeway	232	1280x720	3.73 Mbps
FVDO_Plane	298	1280x720	1.53 Mbps
FVDO_Shore	684	1280x720	3.73 Mbps
FVDO_Golf	311	1280x720	1.15 Mbps

iii. Encoder Configuration

VVenC is employed for VVC encoding tasks, with configurations set for random access mode using varying quantization parameters (QP). A group of pictures (GOP) structure of 32 frames is applied, utilizing a hierarchical approach. Between the intra and inter keyframes, frames are encoded using bi-directional prediction within this hierarchy. Given the high computational demands of VVC encoding, enabling multithreading via command line options can significantly improve CPU performance. Further specifics about the command configurations are detailed in the following sections.

VVenC- 1.8.0:Each video sequence is encoded using a fixed quantization parameter (QP) value of 32. The corresponding command line examples for performing the encoding on a Windows operating system are provided below.

Encode**:** vvencapp.exe -i<input raw file> -s <frame size> -f <frames to be encoded> -fr<frame rate> -c <file type> -- internal-bitdepth<bit depth size> -o <bit stream file>

V. RESULTS & SUMMARY

Content analysis is conducted in this paper with the aim of assessing the content complexity of a given video and identifying frames with distinct static and motion characteristics. This involved analyzing spatial and temporal features of the video frames. For content complexity, spatial complexity was measured by examining textures, colors, and object layouts within frames. Temporal complexity was assessed by analyzing the rate of change between consecutive frames. Higher spatial and temporal complexities indicate scenes with more intricate visual information and dynamic content. The results revealed varying levels of content complexity throughout the video. Scenes with intricate details, rapid motion, or dynamic changes were identified as high-complexity segments, while static or slow-changing scenes were classified as low-complexity.

To determine quantization parameters based on the motion complexity of the given content, these parameters are used in the VVC encoder to encode video for better compression efficiency and reduced computational complexity. When motion complexity is high, the quantization parameter (QP) is set low; conversely, for lower motion complexity, the quantization parameter (QP) value is set high as shown in Table II.

TABLE II

RESULTS OF CONTENT ANALYSIS OF VIDEO SEQUENCES

Sequence Name	Motion Frames	Motion Complexity	QP Value
FVDO_Freeway	231	5628.55	28
FVDO_Plane	297	1388.51	54
FVDO_Shore	683	2122.50	50
FVDO_Golf	310	7178.95	18

Here, Table III presents compression efficiency in YUV-PSNR values, while Table IV compares encoding times between the default QP value method and the proposed QP value method using the OpenCV library.

TABLE III

RESULTS OF COMPRESSION EFFICIENCY YUV-PSNR VALUE OF VIDEO SEQUENCES IN BOTH METHODS

Sequence Name	Frames	Purposed (%)	Default(%)
FVDO_Freeway	232	40.1293	37.8095
FVDO_Plane	298	32.4533	43.1862
FVDO_Shore	684	36.3305	44.1994
FVDO_Golf	311	52.3971	43.1975

TABLE IV

RESULTS OF TIME COMPARISON VIDEO SEQUENCES IN BOTH METHODS

Sequence Name	Frames	Purposed (s)	Default (s)
FVDO_Freeway	232	6126.020	4742.400
FVDO_Plane	298	2848.430	4127.567
FVDO_Shore	684	6450.000	10283.213
FVDO_Golf	311	13552.432	4863.841

In less complex regions, higher quantization levels may be used to achieve more significant compression. Higher motion complexity, high texture detail, or intricate patterns may lead to lower quantization to decrease encoder time.

VI. CONCLUSION

In conclusion, our study on Dynamic Mode Parameter Control (DMPC) for Versatile Video Coding (VVC) highlights its potential in balancing compression efficiency and computational complexity. By dynamically adjusting quantization parameters based on content complexity, we achieved significant improvements in compression efficiency while reducing computational overhead. Through content complexity analysis and the implementation of a dynamic quantization parameter adjustment approach, we observed better compression ratios without compromising visual quality. This method not only enhances encoding performance but also reduces computational complexity, leading to faster encoding times and more efficient resource utilization. While our findings underscore the promise of DMPC for VVC, further research is warranted to refine complexity analysis methods and explore advanced techniques for dynamic parameter adjustment, ultimately advancing the efficiency and scalability of VVC encoding systems for diverse multimedia applications.

REFERENCES

[1] B. Bross, et al, "*Versatile Video Coding (Draft 2)*", ISO/IEC doc. JVET-K1001, Ljubljana, SI, 10–18 July 2018.
[2] Zeng, H.; Xu, J.; He, S.; Deng, Z.; Shi, C. *Rate Control Technology for Next Generation Video Coding Overview and Future Perspective.* Electronics 2022, 11, 4052.
[3] JCT-VC, *High Efficiency Video Coding (HEVC) Test Model 9 Encoder Description.Joint Collaborative Team on Video Coding (JCT-VC)* of ITU-T SG16 WP3 andISO/IEC JTC1/SC29/WG11, 11th Meeting: Shanghai, CN, 10–19 October 2012.
[4] Xu, J., Wang, B. & Peng, Q. *Adaptive intra-refresh for low-delay error-resilient video coding*. J Real-Time Image Proc 20, 70 (2023).
[5] Shengtao Yu, Cheolkon Jung " *Adaptive perceptual quantizer for high dynamic range video compression* ", School of Electronic Engineering, Xidian University, Xian, Shaanxi 710071, China,Article Nov 2018.
[6] W. Kuang, Y. -L. Chan, S. -H. Tsang and W. -C. Siu, "*Machine Learning-Based Fast Intra Mode Decision for HEVC Screen Content Coding via Decision Trees*" in IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 5, pp. 1481-1496, May 2020, doi: 10.1109/TCSVT.2019.2903547.
[7] Video sample from *https://www.fastvdo.com/H.264.html*
[8] VVC Encoder reference*https://www.hhi.fraunhofer.de/fileadmin/Departments/VCA/MC/VVC/vvenc-v0.2.1-v1.pdf*