Versatile Video Coding Using Discrete Wavelet Transforms and Histogram Analysis

Reka Sandaruwan[1] and Anil Fernando[1]
[1]*Department of Computer and Information Sciences, University of Strathclyde, Glasgow, UK*
Email: reka.gallena-watthage@strath.ac.uk

Versatile Video Coding (VVC) introduces advanced techniques such as sophisticated intraprediction mechanisms that significantly improve compression efficiency. However, the increased complexity in intraprediction presents new challenges in computational load and power consumption, essential considerations for efficient video coding. This paper proposes a novel framework that combines discrete wavelet transforms (DWT) and histogram analysis to optimise the prediction directions in VVC. Through selective reduction of prediction directions based on dominant texture orientation and minimising residual energy, the framework achieves enhanced computational efficiency while maintaining video quality. Experimental results confirm its effectiveness, showing minimum rate distortion performance and superior encoding efficiency compared to other methods in the literature.

Introduction: The International Telecommunication Union (ITU) and the Joint Video Experts Team (JVET) recently introduced the Versatile Video Coding (VVC) standard, marking the next generation in video coding [1]. VVC achieves a 50% bitrate reduction over its predecessor, High Efficiency Video Coding (HEVC), without compromising visual quality. This impressive compression efficiency is mainly due to two core techniques: intraprediction and interframe motion compensation. Intra-prediction plays a vital role in achieving high visual quality within VVC by predicting pixel values within a frame based on adjacent pixels, thereby reducing spatial redundancies. VVC introduces a sophisticated partitioning scheme, allowing each 128x128 Coding Tree Unit (CTU) to be divided into smaller blocks to 4x4 pixels using various partitioning modes [2]. These include Binary Tree (BT) splits for vertical and horizontal division, Ternary Tree (TT) splits for three-part division, and the innovative Quad-tree Plus Multi-Type Tree (QTMT) structure [3], which combines quad-tree splits with binary and ternary options for enhanced flexibility. Unlike HEVC, which relies on a fixed prediction unit (PU) concept, VVC dynamically adapts prediction directions based on varying block structures. This adaptability enables for more precise intraprediction of complex textures, enhancing compression efficiency. However, the increased complexity in intra-prediction presents new challenges in computational load and power consumption, essential considerations for efficient video coding.

This paper presents a novel method to optimise intraprediction in VVC by using wavelet transforms and histogram analysis methods to selectively reduce the number of prediction directions while maintaining video quality. By minimising computational load, encoding time, memory use, and energy consumption, our approach aims to achieve a practical balance between compression efficiency and computational resource savings. Three navalities have introduced this paper. The first one is the DWT and histograme-based selective hybrid method to reduce the intra-prediction directions. The second is quality maintenance using the minimum residual energy selection process while the intraprediction direction is selected. The third characteristic is that we have used constant bitrate (CBR) to encode in experimental design. In the literature, no CBR method is found to be used in this kind of research.

Related work: Versatile Video Coding (VVC) introduces advanced coding tools and flexible block partitioning that offer substantial compression efficiency over HEVC. However, this complexity requires optimised approaches to manage the computational demands in the encoding. In VVC, the partitioning of the Coding Tree Unit (CTU) is managed through the Quad Tree with structures based on the Multi-type Tree (QTMT), represented by Partition Hierarchy Maps (PHM), which outline the partition structure at 8x8 unit levels [3]. CTUs are classified according to partition prediction difficulty, with a decision tree used to select optimal split modes according to the PHM. Adaptive networks with varying capacities are employed for different CTU classes, effectively balancing performance on complex CTUs and reducing computation for simpler structures. Most Frequent Mode (MFM) for Intra-Mode Coding: A new intra-mode coding method, MFM, extends the Most Probable Mode (MPM) by encoding the prediction mode based on the frequency of the mode in neighbouring blocks [4]. The methodology described in [5] introduces a deep learning framework to enhance the efficiency of the VVC standard by predicting partition paths during intercoding. To handle screen content with step and asymmetric edges, Geometric Partitioning Mode (GPM) uses an adaptive blending technique with four matrices chosen based on local discontinuities and gradients [6]. VVenC incorporates a hierarchical combination of picture-level and in-picture parallelisation, achieving up to a 4x speedup with four threads [7]. This parallelisation leverages smaller CTU block sizes, wavefront processing, and tile partitioning configurations, allowing efficient encoding across varying video resolutions and presets. To address the vast search space for encoding in VVC, heuristic algorithms [8] with recursive splits of the early termination limit effectively reduce runtime. This approach helps the VVenC encoder maintain high compression efficiency with reduced computational overhead. A classification-prediction framework accelerates CU partitioning by categorising CTUs based on prediction complexity and selecting subnetworks accordingly. This adaptive approach streamlines PHM prediction [9], enhancing coding efficiency by focussing resources on complex partitions. VVC requires a targeted pruning of the encoder search space [10]. Effective strategies are crucial to maintain both encoding performance and Quality of Experience (QoE) given the extensive block partitioning and intra-prediction flexibility in VVC.

Proposed framework: This paper proposes a novel approach that combines wavelet transforms and histogram analysis to optimise the prediction directions in VVC. This integrated method harnesses the strengths of both techniques, utilising wavelet analysis for multiscale texture orientation and histogram analysis for detailed directional consistency. This synergy facilitates a highly selective, efficient, and precise intraprediction process. The proposed model employs a sequential, step-by-step procedure to determine the optimal number of prediction directions for each coding unit (CU). Fig. 1 illustrates the high-level operations of the proposed method. The initial stage of the VVC codec's encoding process involves block partitioning, where the video frame is segmented into CTUs. VVC supports a maximum CTU size of 128x128 pixels, with the flexibility to partition blocks to a minimum size of 4x4 pixels, enabling fine-grained adaptation to varying content characteristics within the frame. The CTUs in the VVC codec are recursively partitioned into smaller blocks using the QTMT structure. These partitioned blocks are then processed along two parallel paths. In the first path, the blocks are sent to a DWT converter, which decomposes each CU block into different frequency subbands Low-Low (LL), Low-High (LH), High-Low (HL), HL, High-High (HH) to capture various directional components. The energy of each subband is calculated by adding the squared coefficients, providing an initial estimate of the dominant orientation within the block. In the second path, the CU blocks are analysed through the generation of histograms. For each block, the gradient magnitudes and orientations of individual pixels are computed using Canny operators. These gradients are used to construct an orientation histogram that records the number of pixels corresponding to specific orientation ranges up to 16 bins (0°-22.5°, 22.5°-45°, 45°-67.5°,... 337.5°-360°). The output of both paths is then used to determine the optimal intraprediction directions. The residual energy is calculated using Equation 1. The residual block R(x,y) is computed by taking the difference between the original pixel values O(x,y) and the predicted pixel values P(x,y) for each pixel location (x,y) within a block. The squared values are then summed over all pixels in the block to obtain the residual energy E, as denoted by Equation 2. Lower residual energy indicates better prediction quality. Finally, the method that offers the lowest residual energy between the DWT and histogram-based approaches is selected for optimal intracoding.

DWT base prediction optimization: This process enables the identification of directional textures within video content. This is an asthmatic

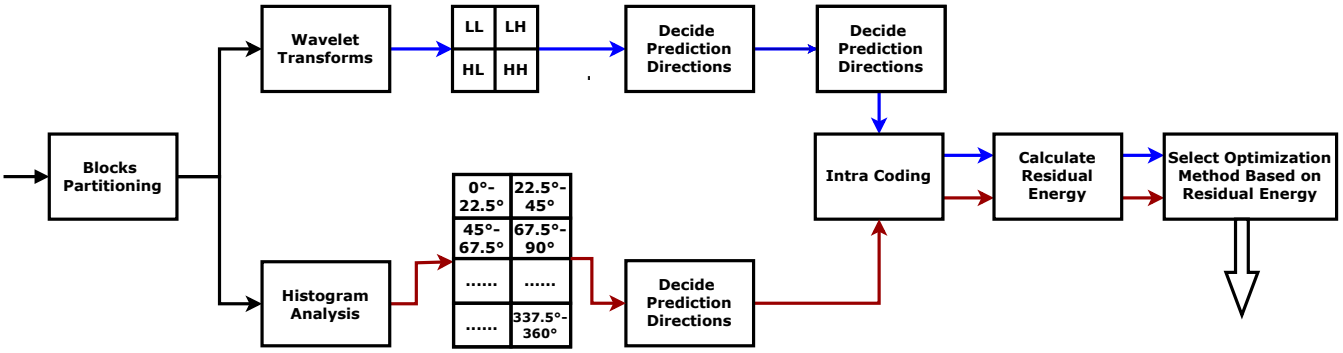


Fig 1 *High-level architecture of the proposed framework.*

explanation of the proposed method. For a given block B of size NxN, the DWT decomposes B into four sub-bands denoted by Equation 3. Each sub-band S contains coefficients C(xy) for pixel (x,y) in that sub-band as Equation 4. Calculate the energy of each subband by adding the squared wavelet coefficients within each subband. The energy distribution across the subbands indicates the primary texture direction in the block, where the energy of a subband E_S is calculated as Equation 5. Based on the energy analysis, prioritise only those prediction directions that align with the dominant orientations. If E_{LH} is the largest, horizontal details dominate. If E_{HL} is the largest, vertical details dominate. If E_{HH} is the largest, diagonal details dominate. This step significantly reduces the number of prediction directions by limiting them to only the directions that align with the dominant characteristics of the block. Using the energy values of E_{LH}, E_{HL}, E_{HH} the most relevant prediction directions ($D_{selected}$) can be selected. Apply a filtering technique to further smooth the prediction, especially if multiple directions are retained. Mode-dependent intra-smoothing can be employed to avoid visual artefacts in cases where multiple predictions are used. For blocks where multiple directions have similar energies, an adaptive filtering process is applied, as shown in Equation 6, where P_d: Prediction based on direction d and w_d: Weight proportional to the energy E_d of the corresponding subband. If $|D|$ is the total number of prediction modes in standard VVC and $|D_{selected}|$ is the reduced ($|D_{selected}| < |D|$). This reduces complexity, where computational cost C is proportional to the number of modes as in Equation 7.

$$R^2(x,y) = (O(x,y) - P(x,y))^2 \tag{1}$$

$$E = \sum_{x,y} R^2(x,y) \tag{2}$$

$$B \xrightarrow{DWT} \{\mathrm{LL}, \mathrm{LH}, \mathrm{HL}, \mathrm{HH}\} \tag{3}$$

$$S = \{C_{xy}\}, \quad x, y \in [1, N/2] \tag{4}$$

$$E_S = \sum_{x=1}^{N/2} \sum_{y=1}^{N/2} C_{xy}^2 \tag{5}$$

$$P_{final} = \sum_{d \in D_{selected}} w_d P_d \tag{6}$$

$$C_{reduced} \approx \frac{|D|}{|D_{selected}|} C_{standard} \tag{7}$$

Histogram base prediction optimisation: This approach uses gradient-based histograms to guide the intra-prediction process to the most relevant directions, minimising computational load. Gradients quantify the intensity changes across neighbouring pixels, while histograms aggregate orientation data to identify predominant texture directions. Analyse the orientation histogram to find the bin(s) with the highest counts, representing the predominant orientation(s) in the block. Using the Canny edge detection operator[11], compute the gradient magnitudes G(x,y) by Equation 8 and orientations θ(x,y) for each pixel (x,y) by equation 9 in a block, Where O represents the original intensities of the pixels in the block and $\frac{\partial O}{\partial x}$ and $\frac{\partial O}{\partial y}$ are the horizontal and vertical gradients, respectively. For example, if the 45°-67.5° bins have the highest count, it suggests a strong 45°-67.5° structure in the block. If a single orientation is dominant, focus on prediction directions that align with this orientation. If multiple orientations have similar counts, select the corresponding prediction directions, favouring those with higher histogram counts. Bin the gradient orientations θ(x,y) into specific ranges. Each bin's count H_k represents the number of pixels that fall within its range. Equation 10 denoted orientation histogram formation where, w(x,y) is a weight based on gradient magnitude G(x,y) and $\delta(k, \mathrm{bin})$ is 1 if the orientation falls into the bin k, otherwise 0. Based on the identified dominant orientation(s), choose a subset of prediction directions that best match the content. Find the bin(s) k_{max} with the highest counts H_k, representing the dominant texture orientation (s) as Equation 11. When multiple directions are selected, apply a mode-dependent intra-smoothing technique to refine the prediction result, reducing the likelihood of visual artefacts. Equation 12 denotes this intra-smoothing technique where, $P_{d(x,y)}$ is the prediction from direction d and w_d α H_k, the histogram count for the corresponding orientation. Focussing on dominant orientations ensures a lower residual energy E_{res} of equation 13.

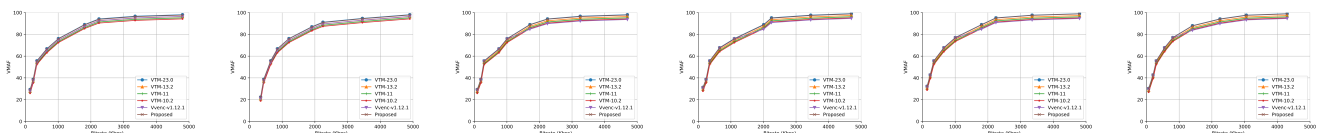
$$G(x,y) = \sqrt{\left(\frac{\partial O}{\partial x}\right)^2 + \left(\frac{\partial O}{\partial y}\right)^2} \tag{8}$$

$$\theta(x,y) = \arctan\left(\frac{\frac{\partial O}{\partial y}}{\frac{\partial O}{\partial x}}\right) \tag{9}$$

$$H_k = \sum_{(x,y)} w(x,y) \cdot \delta\big(k, \mathrm{bin}(\theta(x,y))\big) \tag{10}$$

$$k_{max} = \arg\max_k H_k \tag{11}$$

$$P_{final}(x,y) = \sum_{d \in D} w_d \cdot P_d(x,y) \tag{12}$$



Class A1 ***Class A2*** ***Class B*** ***Class C*** ***Class D*** ***Class E***

Fig 2 *RD curves for video classes.*

Table 1. Comparison of the Performance Metrics for Latest Research and the Proposed Approach.

Class	Tissier et al. [2], VTM 10.2		Proposed, VTM 10.2		Peng et al. [3], VTM 13.2		Proposed, VTM 13.2		Liu et al. [5], VTM 11		Proposed, VTM 11	
	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)
A1	1.81	51.1	1.28	52.4	2.01	48.1	1.26	53.0	1.47	26.1	1.27	51.5
A2	1.86	44.6	1.29	42.2	1.71	47.2	1.27	48.3	—-	—-	1.27	46.3
B	2.21	46.5	1.72	42.3	2.17	47.9	1.71	49.2	1.06	23.3	1.68	45.2
C	3.20	43.1	1.89	47.4	1.98	44.2	1.88	48.1	0.29	17.3	1.88	44.6
D	3.02	36.8	2.23	46.4	2.09	38.5	2.22	44.5	0.15	08.8	2.21	40.0
E	1.45	38.7	1.05	47.1	1.66	40.8	1.05	44.2	0.80	14.2	1.05	40.1
Average	**2.26**	**43.5**	**1.57**	**46.3**	**1.94**	**44.5**	**1.56**	**47.88**	**0.83**	**19.34**	**1.56**	**45.6**

$$E_{\text{res}} = \sum_{(x,y)} [O(x,y) - P_{\text{final}}(x,y)]^2 \tag{13}$$

$$\Delta T = \frac{T_{\text{baseline}} - T_{\text{optimized}}}{T_{\text{baseline}}} \times 100\% \tag{14}$$

Experimental Setup: The experiment uses a widely recognised classified video dataset that is frequently cited in the literature [2, 3, 5]. The study is conducted using VTM reference software, which has been commonly used in the literature for performance comparison. It is compared with the same version used in the literature. A Constant Bitrate (CBR) encoding method is used to ensure a consistent data rate for the experiment and optimise live video streaming. The testing environment is set up on Ubuntu 22.04.3 LTS, running Linux Kernel 6.5 on a 64-bit Intel Core i7 architecture with hyperthreading, 16 GB RAM, and GPU-enabled acceleration. Intel Media SDK is used to enhance video processing performance. The evaluation employs PSNR and Video Multimethod Assessment Fusion (VMAF) metrics to compute rate distortion performance, with the Bjntegaard Metric (BD-Rate) used for rate-distortion comparison. The encoded time gain (ΔT) is analysed as a measure of the reduction of relative complexity, quantifying the performance improvement achieved by optimisations. The complexity reduction is calculated by comparing the encoding times with and without the applied technique, as described in Equation 14 where Where T_{baseline} Encoding/decoding time of the unoptimized system and $T_{\text{optimized}}$ Encoding/decoding time of the optimised system.

Table 2. Comparison of Performance Metrics for State-of-the-Art CODECs and the Proposed Approach.

Class	Proposed, Vvenc 1.12.1		Proposed, VTM 23.0	
	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)
A1	-18.21	-28.2	1.21	54.2
A2	-16.53	-19.1	1.22	50.1
B	-14.17	-26.8	1.67	53.6
C	-16.98	-24.2	1.98	49.2
D	-18.12	-12.5	2.11	47.7
E	-15.66	-29.4	1.02	48.3
Average	**-16.61**	**-23.34**	**1.53**	**50.5**

Experimental Results: The performance of the proposed method was thoroughly evaluated and analysed to assess its effectiveness in terms of quality distortion and performance improvement. The analysis focused on two key metrics: the reduction in quality distortion and the computational efficiency of the method. A lower degree of quality distortion, as demonstrated by metrics such as VMAF, indicates the significant advantage of the proposed approach in preserving visual fidelity while achieving compression. This highlights its ability to maintain high video quality, even with reduced computational demands. Furthermore, the reduction in complexity achieved by the proposed method was quantified by evaluating the savings in coding time. The reduction in encoding time, represented by metrics such as (ΔT), underscores the efficiency of the method in minimising computational overhead.

Fig 2 illustrates the performance of the proposed method, demonstrating minimal quality distortion compared to standard VTM and optimised VVenc encoders (please zoom in for an improved viewing experience). The results indicate that the proposed method achieves slightly lower quality distortion than VTM while delivering better quality preservation than VVenc, highlighting its effectiveness in maintaining video fidelity. Table 1 presents the reduction in complexity and performance improvement achieved by the proposed method, quantified using (ΔT). The proposed method demonstrated lower quality distortion and reduced encoding complexity compared to [2] and [3]. This indicates that the proposed method provides significant improvements in both quality and complexity compared to the aforementioned literature. However, the proposed method exhibited slightly higher quality distortion and a greater reduction in encoding complexity compared to [5]. This suggests that while the proposed method achieves a better reduction in coding complexity, [5] performs better in terms of quality. Table 2 compared to the optimised VVenc encoder, the proposed method ensured better quality, while VVenc excelled in reducing the complexity of coding. Using VTM 23.0 as the reference model encoder, the proposed method showed a minimal difference in BD-Rate from VTM while delivering exceptional performance in reducing coding complexity. This performance enhancement not only accelerates the encoding process, but also demonstrates the feasibility of the proposed method for real-time and resource-constrained applications.

Conclusion: The proposed framework leveraging wavelet-based multiscale texture orientation and histogram-driven directional consistency, the approach reduces computational complexity while maintaining high video quality. Experimental results confirm its effectiveness, showing superior rate distortion performance and encoding efficiency compared to standard VVC and optimised VVenc encoders and a savings of coding complexity over the VTM reference encoder and other methods in the literature. The framework achieves excellent balance of reduced quality distortion and increased performance, as demonstrated by quality metrics and encoding performance metrics, and significantly decreases encoding time, highlighting its suitability for modern video compression demands.

References

1. VVC Recomondation - T-REC-H.266-202309-I!!PDF-E.pdf
2. Tissier, A., et al.: Machine Learning Based Efficient QT-MTT Partitioning Scheme for VVC Intra Encoders. IEEE Transactions on Circuits and Systems for Video Technology 33(8), 4279–4293 (2023). doi:10.1109/TCSVT.2022.3232385. https://ieeexplore.ieee.org/document/10004946/
3. Peng, Z., Shen, L.: A classificationâĂŘprediction joint framework to accelerate QTMTâĂŘbased CU partition of interâĂŘmode VVC. Electronics Letters 59(7), e12770 (2023). doi:10.1049/ell2.12770. https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/ell2.12770
4. Yoon, Y.â., et al.: Most frequent mode for intraâĂŘmode coding in video coding. Electronics Letters 55(4), 188–190 (2019). doi:10.1049/el.2018.7452. https://onlinelibrary.wiley.com/doi/10.1049/el.2018.7452

5. Liu, Y., et al.: Light-Weight CNN-Based VVC Inter Partitioning Acceleration. In: 2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP), pp. 1–5. Nafplio, Greece: IEEE (2022). https://ieeexplore.ieee.org/document/9816276/
6. Lee, M., Oh, S., Sim, D.: An efficient geometric partitioning mode by adaptive blending method for screen content coding. Electronics Letters 59(19), e12961 (2023). doi:10.1049/ell2.12961. https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/ell2.12961
7. George, V., et al.: Efficient VVC Encoding Using Hierarchical Parallelization: A Comprehensive Analysis. International Journal of Semantic Computing 18(02), 175–204 (2024). doi:10.1142/S1793351X2450003X. https://www.worldscientific.com/doi/10.1142/S1793351X2450003X
8. Wieckowski, A., et al.: VVC Search Space Analysis Including an Open, Optimized Implementation. IEEE Transactions on Consumer Electronics 68(2), 127–138 (2022). doi:10.1109/TCE.2022.3148813. https://ieeexplore.ieee.org/document/9702526/
9. Wang, Z., et al.: A Fast Transform Algorithm for VVC Intra Coding. In: 2022 11th International Conference on Communications, Circuits and Systems (ICCCAS), pp. 237–240. Singapore, Singapore: IEEE (2022). https://ieeexplore.ieee.org/document/9825469/
10. Bossen, F., et al.: VVC Complexity and Software Implementation Analysis. IEEE Transactions on Circuits and Systems for Video Technology 31(10), 3765–3778 (2021). doi:10.1109/TCSVT.2021.3072204. https://ieeexplore.ieee.org/document/9399488/
11. Canny, J.: A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-8(6), 679–698 (1986). doi:10.1109/TPAMI.1986.4767851. https://ieeexplore.ieee.org/document/4767851