Journal of the Brazilian Computer Society, 2026, 32:1, doi: 10.5753/jbcs.2026.5354
This work is licensed under a Creative Commons Attribution 4.0 International License.

A Coding-Efficiency Analysis of HEVC Encoder Embedded in High-End Mobile Chipsets

Vítor Costa [**Federal University of Pelotas** | ***vscosta@inf.ufpel.edu.br***]
Murilo Perleberg [**Federal University of Pelotas** | ***mrperleberg@inf.ufpel.edu.br***]
Luciano Agostini [**Federal University of Pelotas** | ***agostini@inf.ufpel.edu.br***]
Marcelo Porto [**Federal University Pelotas** | ***porto@inf.ufpel.edu.br***]

Video Technology Research Group (ViTech), Universidade Federal de Pelotas, Rua Gomes Carneiro 1, Porto, Pelotas, RS, 96010-610, Brazil.

Received: 17 December 2024 • **Accepted:** 07 July 2025 • **Published:** 22 January 2026

Abstract. High-end mobile devices require dedicated hardware for real-time video encoding and decoding processes. However, the inherent complexity of the video encoding process, combined with the physical limitations imposed by hardware design such as energy consumption, encoding time, memory usage, and heat dissipation, demands the implementation of various constraints and limitations in commercial hardware to simplify and make them feasible for general use. The High Efficiency Video Coding (HEVC) standard is the main targeted video encoder for processing high-resolution videos in high-end chipsets. This paper aims to analyze the HEVC encoder implemented into three commercial chipsets found in high-end smartphones (Apple iPhone 14 Pro, Samsung Galaxy S23 Plus, and Redmi Note 10S) from three major mobile chip manufacturers (Apple, Qualcomm, and MediaTek), considering the impacts of video encoder limitations on encoding efficiency (BD-Rate) and encoding time. The results in this paper may be used as a comparative foundation for hardware designers and future works in the field, as it exposes the encoding efficiency drawbacks and the encoding time gains that commercial chipsets exhibit in their HEVC encoder.

Keywords: dedicated hardware, HEVC, mobile devices, video coding

1 Introduction

Nowadays, the consumption of video is continually evolving. Streaming service platforms, video conferences, and social networks can be listed as the main drivers of the analyzed growth. From the perspective of the users, access to this type of content is increasingly facilitated through mobile devices, owing to the convenience of access and the personalization of suggested content through algorithms. On a global scale, in 2024, the usage of streaming video content in mobile devices reached 65% in IOS-oriented systems [Bitmovin, 2023]. Additionally, mobile devices provided a growth in connectivity, with an estimated 70% of the world's population expected to have internet connectivity through a mobile application at the beginning of 2024 [Cisco, 2020]. This data aligns with the trend towards the intensification of video traffic on web platforms.

Video content demands a high amount of data to be represented and transmitted to the users, thus requiring applying video compression and decompression processes. However, current video encoders are extremely complex and computationally expensive, requiring a significant amount of energy to achieve high-efficiency video compression in real time. Therefore, mobile devices need to incorporate dedicated hardware to handle video encoding and decoding processes. The hardware designers must address the physical limitations imposed by these projects, such as heat dissipation, energy consumption, processing time, memory access, and chip area constraints. Therefore, video encoders and decoders of several day-to-day multimedia devices must apply constraints and hardware-friendly strategies, aiming for the practical usability of the application. Works [David *et al*., 2020; Schierl *et al*., 2007; Sebastiaan *et al*., 2012; Kim *et al*., 2014; Correa *et al*., 2011; Abu and Gunasekara, 2014; Jeong *et al*., 2016] proposes complexity-scalable solutions geared towards mobile devices, while works [de Carvalho *et al*., 2007; Agostini *et al*., 2007], discusses hardware architecture for the Brazilian digital television system.

The High Efficiency Video Coding (HEVC) standard, launched in 2013 by the Joint Collaborative Team on Video Coding (JCT-VC) [ITU-T, 2013], has become the leading for commercial video coding and decoding. According to [Bitmovin, 2023], the HEVC presented a growth of 34% and 28% for encoding live videos and videos on demand, respectively, in 2023. The HEVC is the state-of-the-art encoding standard embedded on high-end multimedia mobile chipsets [Bitmovin, 2023; Scientiamobile, 2018]. Although most chipsets support the HEVC predecessor, the H.264/AVC (Advanced Video Coding) [ITU-T, 2003], it is used as a resource-friendly way of encoding high-resolution videos, with sub-optimal compression capabilities.

Several works in the literature have proposed efficient hardware solutions for different steps of the HEVC encoder. However, due to the complexity-reduction strategies employed in each of these works, coding-efficiency losses, usually measured by the Bjontegaard Delta Rate (BD-Rate) metric [Bjontegaard, 2001], can be observed. The BD-Rate metric quantifies the bitrate variation required to achieve the same level of objective quality, typically measured using Peak Signal-to-Noise Ratio (PSNR), when comparing a baseline to a given

solution. An increase in BD-Rate indicates a degradation in coding efficiency, as more bits are needed to maintain the same objective quality. Conversely, a reduction in BD-Rate reflects an improvement in coding efficiency. BD-Rate is the most widely used metric in the video coding literature for evaluating coding efficiency, as it effectively captures the trade-off between bitrate and objective video quality in a single value.

Taking the dedicated HEVC hardware solutions from reference works [Porto *et al*., 2019; Conceição *et al*., 2015; Leme *et al*., 2019; Perleberg *et al*., 2024; Hu *et al*., 2017; Bubolz *et al*., 2018; Chuen-Ching and Li, 2017; Singhadia *et al*., 2020; Pastuszak and Trochimiuk, 2016; Park *et al*., 2016; Singh and Ahamed, 2018; He *et al*., 2015; Cai *et al*., 2022; Zhang and Lu, 2019; Penny *et al*., 2020; Porto *et al*., 2021; Perleberg *et al*., 2018; Afonso *et al*., 2019] as examples, the observed BD-Rate ranges from 0.3% to 24.16%. There is no predefined tolerability range for BD-Rate increases, making it challenging to assess the overall quality of a work solely based on this metric. Furthermore, the BD-Rate range that widespread mobile chipset video encoders operate is unknown, mainly because manufacturing companies do not express which video encoding constraints were implemented on their dedicated hardware. Thus, the efficiency of commercial smartphone chipset video encoders is not available to serve as a comparative foundation to new improved hardware designs.

In related work [Costa *et al*., 2024], an analysis was conducted, where the efficiency of the HEVC encoder embedded in the Apple A15 Bionic chipset, from the iPhone 13 smartphone, was evaluated. The results in [Costa *et al*., 2024] show an average BD-Rate increase of 19.05%, with an encoding time reduction of 94%, when compared to the reference software of the HEVC. However, as it will be explained later in Section 5, this result can be seen as optimistic, due to a more imprecise analysis in [Costa *et al*., 2024].

This paper presents a coding-efficiency analysis of dedicated hardware implementations of the HEVC standard widely embedded in current high-end smartphone chipsets, describing the video coding constraints employed in each one of the studied chipsets, and presenting the BD-Rate impact provided by those constraints utilizing the reference software of the HEVC, the HEVC Test Model Version 18.0 (HM) [Suehring, 2023]. After obtaining the results, the coding-efficiency and time-savings results showcased by the hardware of the selected smartphones were compared with the golden model implementation of the HEVC standard, available at the HM software.

The high-end smartphones selected in this paper were: Apple iPhone 14 Pro [Apple, 2022], Samsung Galaxy S23 Plus [Samsung, 2023], and Xiaomi Redmi Note 10S [Mi, 2021], featuring the Apple A16 Bionic [Apple, 2022], Qualcomm Snapdragon 8 Gen 2 [Qualcomm, 2022], and MediaTek Helio G95 [MediaTek, 2020] chipsets, respectively. These devices were selected because they belong to the top three manufacturers in the global smartphone market share in 2023 [Canalys, 2023]. Regarding chipset manufacturers for mobile devices, MediaTek leads the market with a 40% share, followed by Qualcomm with 23%, and Apple with 17% by the first quarter of 2024 [Counterpoint, 2024]. Given their significant market shares, the strategies employed by these companies in the HEVC encoder have a greater impact on the current user's daily experience and better reflect the practical commercial reality of an HEVC encoder chip.

The main objective of this paper is to evaluate the impact of the complexity-reduction constraints applied over the HEVC encoder in the previously described chipsets regarding the encoding efficiency and encoding time. This way, a comparative benchmark can be established, generating reference values for future works in the field and providing useful information to hardware designers to assess the BD-Rate increase and the time savings of commercial HEVC encoders, using three high-end chipsets as references. Additionally, this paper also provides the set of constraints employed by each analyzed chipset.

The rest of this paper is organized as follows: Section 2 explains the main stages and tools of the HEVC standard, as well as its reference software. Section 3 covers the entire evaluation method employed in this work, from the video capture with the smartphones to obtaining the BD-Rate for each of the devices. Section 4 discusses the constraints observed in the three chipsets. Section 5 outlines the BD-Rate and temporal reduction results obtained. Finally, Section 6 presents conclusions regarding all the results obtained.

2 HEVC Background

As with other video coding standards, HEVC standardizes the decoder and defines the syntax of the encoded bitstream. This means that the developers of HEVC encoders are not obligated to use all tools defined by this standard and a lot of simplifications can be applied intending to reduce the computational cost of the encoding process and then, reduce the computation, the consumed energy, among others. Even with these simplifications, the encoder can generate a bitstream compliant with the HEVC specifications. In other words, the encoded bitstream can be decoded by any HEVC decoder.

The JCT-VC defined a golden model software implementation of the HEVC encoder and decoder, called HM [Suehring, 2023]. The HM employs the most advanced techniques and tools supported by HEVC standard. Consequently, the HEVC implementation provided by the HM software is highly unrealistic to be embedded in commercial applications due to its very high computational cost. In summary, the HM always seeks the highest compression efficiency, without being concerned about the complexity, real-time processing, energy consumption, or any other factor related to practical applications. Even with these limitations, HM is a good anchor for research experiments, since it can establish the optimal coding-efficiency results of the HEVC.

The HM software is highly configurable, allowing for customization of its behavior by simply modifying the parameters associated with the encoding process [Suehring, 2023]. For this purpose, the HM contains configuration files that implement various predefined temporal profiles, such as Random-Access (RA), Low-Delay-P (LDP), and All-Intra (AI) [Suehring, 2023]. These temporal profiles primarily affect the structure in which video frames are encoded, as well as the tools and heuristics that will be enabled or disabled during the encoding process. In both RA and LDP profiles,

the frames are encoded in a Group of Pictures (GOP). In the RA profile, the frames from a GOP are encoded out of the chronological order of the capture, allowing previous and forward frames (from the one being encoded) to be used as reference frames. On the other hand, in the LDP profile, the frames from each GOP are encoded in the chronological capture order [Suehring, 2023], thus only frames before the frame to be encoded can be used as reference.

2.1 HEVC Block Partitioning

The first step in the HEVC encoding process is the frame block partitioning, where the frame must be fragmented into smaller blocks. In the HEVC standard, the frames are split into squared blocks called Coding Tree Units (CTU), which by default have a maximum size of 64x64 samples. Each CTU can be recursively divided into smaller squared units called Coding Units (CU). Subsequently, each CU is divided into one or more Prediction Units (PU) or Transform Units (TU) [Sze *et al*., 2014]. The PUs are the units processed by the prediction tools in the HEVC, while the TUs are processed by the transform and quantization steps in the HEVC [Sze *et al*., 2014]. The PUs can be classified according to the resulting shape, which can be squared (64x64, 32x32, 16x16, 8x8, and 4x4), symmetrical (64x32, 32x64, 32x16, 16x32, 16x8, 8x16, 8x4, and 4x8), or even asymmetrical through the Asymmetric Motion Partition (AMP) (64x16, 64x48, 16x64, 48x64, 32x8, 32x24, 8x32, 24x32, 16x4, 16x12, 4x16, and 12x16) [Sze *et al*., 2014].

Each PU or TU allowed at the block-partitioning process is processed by other four main steps: predictions (inter and intra-frame), transforms, quantization, and entropy encoding. This section emphasizes, the predictions steps considering that the constraints presented in Section 4 are mostly focused on reducing the complexity of these steps. The PU concept encompasses all color channels within the unit, whereas the Prediction Block (PB), refers to the information of a single color channel individually [Sze *et al*., 2014].

2.2 HEVC Inter-frame prediction

The encoding order of the frames, and the frames that should be used as references in the Inter-frame prediction of the HEVC, are defined by the temporal profile and by the GOP structure. In the LDP, the reference frame is always a frame at the past of the current frame, respecting the chronological capture order as previously discussed [Suehring, 2023]. On the other side, in the RA profile, the already encoded frames in the list to be used as references include frames at the past and at the future of the current frame [Suehring, 2023], since out-of-order processing is supported. Therefore, having a large GOP size and supporting different reference frames implies higher memory consumption (reference frames must be stored to be used as references) and a greater number of evaluations to be performed (more frames must be evaluated).

The main step of the Inter-frame prediction process is the Motion Estimation (ME). During the ME, the HEVC encoder evaluates, for each PU, the most similar block presented in the reference frames, among several Candidate Blocks (CB) [ITU-T, 2013; Sze *et al*., 2014]. This involves searching for another block in the reference frames where the sample values most closely match the PU to be encoded. Once the best CB is found, a Motion Vector (MV) is traced to represent the displacement from the PU being encoded to the best CB found [ITU-T, 2013]. Since the ME process must be executed for all PUs being encoded, the Inter-frame prediction is an extremely resource-intensive and time-consuming task.

The HM software employs many strategies to reduce the complexity of the ME, such as the use of a limited Search Area (SA) within the reference frame, whose size is delimited by the search range parameter, and the use of predictors to guide the beginning of the search [ITU-T, 2003]. However, even with those strategies, an analysis conducted using HM 16.3 showed that 45% of the total encoding time is still spent on the ME process [Grellert *et al*., 2016]. This complexity analysis further implies that real-time HEVC encoders must implement more expressive simplifications to make real-time ME process feasible.

The HEVC also supports quarter-precision in inter-frame prediction. This requires the adoption of a fractional MV, which may indicate an integer position block, a half position block, or a quarter position block. The fractional MVs are evaluated by Fractional Motion Estimation (FME), which is applied after the integer ME [ITU-T, 2013]. So, after the FME, a predefined number of bits from the final MV defines the mantissa of the MV, thus defining the integer and fractional parts of the MV, which are used for reconstructing the PU considering fractional displacements between original and reference frames [Sze *et al*., 2014].

2.3 HEVC Intra-frame prediction

The HEVC standard has 35 Intra-frame prediction modes, namely planar, DC, and 33 directional modes [ITU-T, 2013]. Each one of those modes interpolates the samples from the neighboring block in a different way. The planar mode involves predicting each sample within the PU using a weighted average of neighboring block samples, where the weight used in the average calculation varies according to the position of the sample within the PU. The DC mode is a simple average of the values of all immediately neighboring samples of the PU. In contrast, the 33 angular modes displace the neighboring block samples by different angles, according to the selected angular mode, through an interpolation process.

The HEVC standard supports the Intra-frame prediction to be performed only for squared PUs, with sizes from 4x4 up to 32x32 samples [ITU-T, 2013]. Even though only four PU sizes are supported, the Intra-frame prediction demands considerable computational complexity, consuming 6% of the total encoding time [Grellert *et al*., 2016], since the PU reconstruction requires processing each sample from the PU independently to the neighboring samples. Furthermore, the Intra-frame prediction operates separately for each color channel; consequently, the different channels from the PU must be independently reconstructed, where luma PBs are interpolated separately from chroma PBs.

3 The evaluation method

The evaluation method employed to generate the results in this paper is subdivided into two well-defined stages, as presented in the flowchart in **Figure 1** The first stage, in blue in **Figure 1**, aims to characterize the observed constraints in each of the three selected chipsets. The second stage, represented by the yellow block in **Figure 1**, employs the constraints identified in the first stage in the HM software, to emulate the behavior of mobile encoders. In this second stage, the recommended test sequences from the Common Test Conditions (CTC) [Boyce *et al*., 2018] are used to ensure precise and replicable encoding efficiency and computational complexity analyses. All evaluations performed by the proposed evaluation method were conducted in the HM software version 18.0. The following two subsections explain each one of these two stages.

3.1 Constraints characterization

The primary objective of this first stage, as previously described, is to analyze the constraints and limitations presented in each of the three analyzed chipsets. In other words, the goal is to assess which tools and parameters of the HEVC standard are fully implemented in the encoding chipsets, which ones have been reduced, and which ones have been eliminated, when compared to the default implementation of the HM software.

The Samsung Galaxy S23 Plus and Apple iPhone 14 Pro smartphones, with the respective chipsets Qualcomm Snapdragon 8 Gen 2 and Apple A16 Bionic, are capable of recording videos in UHD 4K (3840x2160 pixels) and FHD (1920x1080 pixels) resolutions at framerates of 30 and 60 frames per second (fps). So, in step (1) presented in **Figure 1**, eight videos were captured and encoded with the chipsets presented in both smartphones, two videos for each possible pair of resolutions and frame rates. The Xiaomi Redmi Note 10S smartphone, equipped with the MediaTek Helio G95 chipset, is unable to record UHD 4K sequences at 60fps. So, only six sequences were obtained, two for each pair of possible resolutions and frame rates (UHD4K@30fps, FHD@60fps, FHD@30fps). A total of 22 video sequences were captured in the step (1) of the first stage. All captured videos have a bit depth of eight bits and have the High Dynamic Range (HDR) feature disabled.

A preliminary analysis was conducted to observe if the decisions made by the encoder could be affected by the battery level of the smartphone. It was concluded that the battery level of the smartphone does not affect the video encoder constraints in the three evaluated chipsets. Moreover, smartphones have a camera settings window that includes certain options affecting the video encoding process on the device. In the case of the three smartphones focused on this work, there is an option to select the format in which videos will be encoded. Then, all the videos were captured in the "High Efficiency" format, as the "More Compatible" format forces the smartphone to adopt the H.264/AVC encoding standard, which is not the focus of this paper.

In the case of the Samsung Galaxy S23 Plus, an additional configuration option must be selected among "Prioritize video quality", "Prioritize saving space", or "High bitrate videos". No differences were observed in the set of encoding constraints presented in each of those three configurations. The only change is the Quantization Parameter (QP) value used for encoding the video frames. For the sake of analyzing the variations between the three configurations a small test was conducted capturing six UHD 4K videos, two for each configuration, and the average QP value for each of those videos were extracted. In the case of "High bitrate videos" configuration the average QP of the two videos was 18, in the case of "Prioritize video quality" configuration the average QP for the two videos was 22, finally with "Prioritize saving space" configuration the average QP for the last two videos was 24. For the analysis of the present work, the configuration "Prioritize saving space" was selected for capturing the eight sequences, since it is the default configuration of the Samsung Galaxy S23 Plus smartphone.

After the capturing of the video sequences, in step (1), the FFmpeg software [FFmpeg, 2000] was used to extract the video channel from the .mp4 and .mov containers generated by the smartphones as stated by step (2). In step (3) the bitstreams from the videos captured with each smartphone were analyzed by using the Zond 265 tool [Multimedia, 2015] which provides a set of tools for HEVC bitstreams analysis such as CU partitioning statistics, MV graphs, the incidence of intra modes, among many other features that facilitates the identification of the main encoding constraints presented in the mobile HEVC encoders chipsets. At this point, it became possible to observe the top-level constraints contained in the analyzed chipsets, such as GOP size, GOP structure, CTU size, number of reference frames, and intra-frame period.

In step (4), an analysis of in-depth constraints presented in the chipsets encoders was conducted. For this purpose, a modified version of the HM decoder was used. This modified version of the decoder allows the recording (during the decoding process) of the decisions related to the incidence of chosen PU sizes in inter-frame prediction, used intra-frame prediction modes, maximum and minimum MV sizes, and fractional portions of MVs.

So, this first stage of our evaluation method allows for characterizing the constraints from the three chipset encoders, which are discussed in Section 4 and used in the second stage of our evaluation method.

3.2 BD-Rate and time reduction evaluation

A fair direct evaluation of BD-Rate using the videos captured by smartphones is not possible, since they have already been encoded by the smartphone encoder alongside the capturing process. Thus, obtaining the raw original videos (without losses), to be used as a baseline, from this initial encoding is not possible. Therefore, an analysis of BD-Rate based on these sequences is useless. So, aiming to overcome the issue, the second stage of our evaluation method involves encoding the CTC sequences with modified versions of the HM software. That modified version of the HM software emulates the constraints observed in each one of the selected chipsets in the first stage of our method.

In step (5), three distinct implementations of the HEVC reference software were employed. On each one, the default

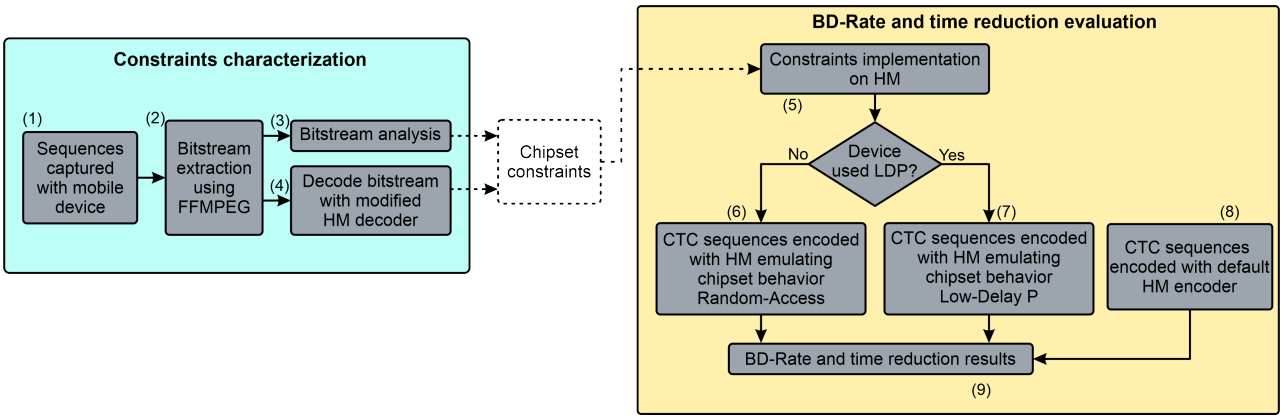

Figure 1. Flowchart of the proposed evaluation method of this work.

source code of the HM encoder was modified to implement the constraints found in each of the three chipset encoders focused on in this paper.

The steps (6) and (7) are responsible for encoding the CTC sequences by using the HM software implementation that emulates the HEVC encoders of each one of the three chipsets. For the simulations of the Samsung Galaxy S23 Plus and the Xiaomi Redmi Note 10S the LDP profile was used, while for the emulation of the Apple iPhone 14 Pro, the RA profile was employed. The reason for these choices is to use a temporal profile that closely resembles the GOP structures of the three chipsets. The step (8) generates the baseline values for comparisons by encoding the CTC sequences using the default implementation of the HM encoder in both LDP (baseline for Qualcomm Snapdragon 8 Gen 2 and MediaTek Helio G95) and RA (baseline for Apple A16 Bionic) profiles.

As input for the simulations in steps (6), (7), and (8), the test video sequences from the HEVC CTC classes A1, A2, B, C, and D were used, being the sequences in classes A1, A2 and B of the same resolutions the analyzed devices are capable of capturing (UHD 4K and FHD). The other classes were included in this investigation to assess the impact of implemented constraints in videos with lower resolutions. Further, each sequence was encoded at the four QP values specified by the CTC (37, 32, 27, 22) [Boyce *et al*., 2018]. All simulations were run in a dedicated server with a Xeon Gold 5120 processor with 14 cores (28 threads), and 64 Gigabytes of RAM. A maximum of 12 encodings were performed in parallel, ensuring that temporal results were not overestimated.

Finally, step (9) is responsible for extracting the BD-Rate and time reduction percentage results of the previous experiments. The BD-Rate metric is calculated based on the bitrate and PSNR values from both baseline encodings and the encodings emulating a chipset. Those values were extracted directly from the HM software log, at the end of the encoding process. The BD-Rate is calculated through interpolating the rate-distortion curves for the four QP values in both baseline and chipset emulation [Bjontegaard, 2001]. After interpolation, the integrals for the two rate-distortion curves are calculated and subtracted, resulting in the BD-Rate value [Bjontegaard, 2001].

The time results are achieved by calculating the average value of four encoding times, one for each recommended QP, in both baseline and chipset emulation contexts. Then, the ratio between the average chipset emulation time and the average baseline time was calculated for each sequence, according to **Equation 1**, generating the time reduction percentage for a given sequence.

$$TR = (1 - \frac{AverageTimeOfChipsetEmulation}{AverageTimeOfBaseline}) * 100 \quad (1)$$

4 Constraints descriptions and analysis

Table 1 depicts all the characteristics and constraints identified in the HEVC encoders from the chipsets, which were identified by applying the first stage of the previously described evaluation method. For comparison purposes, **Table 1** also shows the default characteristics of the RA and LDP profiles of the default HM software version 18.0.

Regarding block partitioning, all analyzed chipsets have a CTU size smaller than the one defined in the HM (64x64 samples). In the Apple A16 Bionic and Qualcomm Snapdragon 8 Gen 2 chips, the CTU size is 32x32, for both UHD 4K and FHD videos. In the case of the MediaTek Helio G95 chipset, for FHD videos the CTU size is also 32x32. However, in UHD 4K sequences, the CTU size is reduced to 16x16. Any reduction in the CTU size directly impacts the complexity and the efficiency of the inter and intra-prediction processes, since it reduces the number of PU sizes evaluated.

As previously mentioned in Section 3.2, the chipsets have different GOP sizes and GOP structures. The Apple A16 Bionic uses a GOP structure closely resembling the RA temporal profile, where frames to be encoded with inter-prediction are encoded in groups of four frames in a non-chronological order, and each frame may use two different reference frames. The Qualcomm Snapdragon 8 Gen 2 and MediaTek Helio G95 chipsets exhibit a behavior closer to the default LDP profile, where frames are encoded in chronological order, and only the frame directly preceding the frame being encoded is used as a reference.

In the case of the Apple A16 Bionic, a maximum of six frames per encoded frame must be stored (four frames from

the current GOP size plus the last frame from the two previous GOPs that may be used as reference frames). A single frame can use a maximum of two different reference frames. For the Qualcomm Snapdragon 8 Gen 2 and MediaTek Helio G95 chipsets, only two frames must be stored (the frame to be encoded and its reference frame), with only one reference frame. Compared with the HM software in the original RA profile, a maximum of eighteen frames must be stored per encoded frame (sixteen from the current GOP plus two possible reference frames from the two previous GOPs), while each frame can have a maximum of four different reference frames. In the case of the original LDP profile of the HM, twelve frames at maximum are stored per encoded frame (eight from the current GOP plus four reference frames that can be outside the current GOP), with each frame having exactly four reference frames.

Since the Apple A16 Bionic supports two reference frames, the inter bi-prediction of PUs can be supported, meaning that one single PU can have up to two MVs pointing towards two different CBs. The bi-prediction is disabled in Qualcomm Snapdragon 8 Gen 2 and MediaTek Helio G95 implementations, since both only support one reference frame.

So, it is possible to observe that the three chipsets implement memory-efficient approaches toward video coding when compared to the default implementation of the HM software. This is mainly due to the bottleneck that memory systems impose on current high-end hardware for video coding [Perleberg *et al*., 2024].

The intra period, which determines the period (in the number of frames) of which a frame must be encoded solely with intra-frame prediction tools, is equal to the video framerate being encoded on the Qualcomm and MediaTek chipsets. In the case of the default LDP profile, only the first frame from the video is intra coded. However, in the Apple chipset, the intra-period is 32 for 30fps videos and 60 for 60fps videos. This slight change in the case of 30fps videos may be justified by the fact that the intra period needs to be a multiple of the GOP size.

Regarding the limitations of PU sizes, all chipsets exhibit a lack of support for squared or rectangular PUs of size 64. The absence of support for asymmetrical PUs (AMP tool) was also detected, since no asymmetric PUs were detected for all captured video sequences. In the case of the Apple A16 Bionic and Qualcomm Snapdragon 8 Gen 2 chipsets, the supported PU sizes in inter-frame prediction are the same, they both support squared and rectangular blocks with sizes lower or equal to 32, with a lack of support for rectangular PUs of size 8 (8x4 and 4x8). For the MediaTek Helio G95 chipset, the support for certain PU sizes depends on the resolution of the encoded video. In both FHD and UHD 4K sequences, squared and rectangular PUs of size 16 are supported, however, the chipset encoder provides additional support for squared PUs of 32x32 samples in FHD videos. All other PU sizes are not supported by this chipset.

Concerning the search range parameter, several experiments were conducted to determine an approximation of the search range that generates MV ranges close to those observed in each of the three chipset encoders. This was done because this type of information is not presented in the video bitstream and, then, the search range must be inferred considering the MV sizes. The default values of the search range parameter for the HM RA and LDP profiles are 384 and 64, respectively. However, the observed MV range was not restricted by the search range value. This is justified by the inter-frame algorithm choosing a new predictor MV after the ME found a similar block in the reference frame, so that a larger MV should be used to indicate the displacement between the most similar CB regarding the new predictor MV. Further, the default parameters in HM generate much larger MVs than those observed in the supported vertical and horizontal MV intervals of the chipsets, as presented in **Table 1**.

Given this discrepancy, tests were conducted to collect the MV ranges obtained by encoding the CTC test sequences in the modified versions of HM software (emulating each chipset encoder) with different search range values. **Table 2** illustrates the different MV ranges of those simulations. It is worth mentioning that these simulations were only performed for obtaining the best search range approximation of each chipset, and were not performed for obtaining the actual coding efficiency and time reduction results.

The results in **Table 2** show that the HM software emulating the Apple A16 Bionic chipset with a search range of eight, resulted in the MV range of [-428, 296] vertically and [-463, 309] horizontally. The upper bound of the horizontal MV interval is already close to those listed in **Table 1**, regarding the maximum MV interval found in the videos captured with this chipset. On the other hand, by increasing the search range parameter to 16, or even 32, only further distanced the MV ranges from the target, as also denoted by **Table 2**. Thus, the approximation with a search range of eight was considered the best approximation and used for generating the BD-Rate result related to the Apple A16 Bionic in Section 5.

Table 2 also shows the MV ranges from the emulations of the Qualcomm Snapdragon 8 Gen 2 chipset. With a search range of 15, an MV range of [-433, 127] vertically and [-318, 180] horizontally is displayed, which is already close to the MV intervals supported by the chipset itself (considering the upper horizontal limit). When a search range of 14 is used, an MV range of [-339, 149] horizontally is obtained, which provides horizontal MV values smaller than those presented by the device. Therefore, the approximation with search range 15 was considered when generating the BD-Rate related to the Qualcomm Snapdragon 8 Gen 2 chipset in Section 5.

Finally, the HM software emulating the MediaTek Helio G95 chipset, with a search range of six, resulted in MV intervals of [-249, 80] vertically and [-200, 159] horizontally, values higher than those supported by the chipset, as pointed by **Table 2**. Hence, the approximation with search range six was chosen for calculating the BD-Rate for the MediaTek Helio G95 in Section 5.

Note that although the MV range of the Apple A16 Bionic chipset is higher than that of the Qualcomm Snapdragon 8 Gen 2 chipset, a higher search range approximation was used for Snapdragon (15) than for the Apple chipset (8). It is hard to determine the cause of this behavior, since it is impossible to know which ME algorithm and parameters were used by each chipset to determine the CBs for evaluation. Although, one possible cause is the GOP structure employed in each of these chipsets, where the LDP GOP structure will generate smaller vectors than the RA structure, even when a higher

Table 1. Observed constraints of the HEVC encoder of the Apple A16 Bionic (iPhone 14 Pro), Qualcomm Snapdragon 8 Gen 2 (Galaxy S23 Plus), and MediaTek Helio G95 (Redmi Note 10S) chipsets.

Tool/Parameters	HM-18.0 (default Random-Access)	HM-18.0 (default Low-Delay-P)	Apple iPhone 14 Pro	Samsung Galaxy S23 Plus	Xiaomi Redmi Note 10S
Chipset	N/A	N/A	Apple A16 Bionic	Qualcomm Snapdragon 8 Gen 2	MediaTek Helio G95
CTU size	64x64	64x64	32x32	32x32	16x16 (UHD videos) 32x32 (FHD videos)
GOP size	16	8	4	1	1
GOP structure	Random-Access	Low-Delay-P	Random-Access	Low-Delay-P	Low-Delay-P
Reference Frames	Up to 4	Up to 4	Up to 2	1 (Previous chronological frame)	1 (Previous chronological frame)
Bi-prediction support	Enabled	Enabled	Enabled	Disabled	Disabled
Intra Period	32	-1 (First frame only)	60 (60fps videos) 32 (30fps videos)	Same as video framerate	Same as video framerate
Asymmetrical PUs	Yes	Yes	No	No	No
Inter PU limitations – UHD 4K videos	N/A	N/A	No 4x8 and 8x4 PU support	No 4x8 and 8x4 PU support	Only 8x16, 16x8, and 16x16 PUs
Inter PU limitations – Other resolutions	N/A	N/A	No 4x8 and 8x4 PU support	No 4x8 and 8x4 PU support	Only 8x16, 16x8, 16x16 and 32x32
Vertical MV interval	[-2055, 2045]	[-1533, 1968]	[-188, 187]	[-77, 60]	[-45, 44]
Horizontal MV interval	[-1745, 2139]	[-2176, 2138]	[-316, 315]	[-189, 188]	[-77, 76]
Search Range (approximation)	384	64	8	15	6
FME bits	2 [.0, .25, .5, .75]	2 [.0, .25, .5, .75]	2 [.0, .25, .5, .75]	2 [.0, .25, .5, .75]	2 [.0, .25, .5, .75]
Intra PU limitations	N/A	N/A	No 4x4 luma PB support in UHD 4K@60fps	No 4x4 luma PB support in UHD 4K@60fps	Only 16x16 PUs

Table 2. MV ranges for the different chipsets emulations considering variations on the search range parameter.

Chipset	Search Range	Vertical MV Range	Horizontal MV Range	Targeted Vertical MV Range	Targeted Horizontal MV Range
Apple A16 Bionic	32	[-754, 500]	[-774, 726]	[-188, 187]	[-316, 315]
	16	[-497, 527]	[-895, 328]		
	8	[-428, 296]	[-463, 309]		
Qualcomm Snapdragon 8 Gen 2	16	[-382, 155]	[-361, 204]	[-77, 60]	[-189, 188]
	15	[-433, 127]	[-318,180]		
	14	[-247, 108]	[-339, 149]		
MediaTek Helio G95	16	[-674, 153]	[-491, 271]	[-45, 44]	[-77, 76]
	8	[-318, 94]	[-354, 130]		
	6	[-249, 80]	[-200, 159]		

search range is used.

The RA profile, employed in the Apple A16 Bionic chipset, encodes the frames in a different order than the chronological order, so that the only reference frame available may not be temporally close to the frame being encoded. In this case, information that could be found with a smaller MV in the chronological preceding frame is found with a higher MV in distant reference frames. Conversely, the Qualcomm Snapdragon 8 Gen 2 chipset employs the LDP profile and has only one reference frame, chronologically preceding the frame to be encoded. So, it is possible to assume that, in the LDP, the difference between the frame to be predicted and its reference frame should be minimal for most of the encoded frames.

The FME bits in **Table 1** refer to the precision with which the encoders can represent fractional values in their MVs. A higher number of bits in the fractional part results in greater representability and precision in the FME process. All chipsets evaluated, and the default HM software with RA and LDP profiles, showed the same number of fractional bits, as indicated in **Table 1**.

Finally, all three chipset encoders showed limitations in the number of supported intra-frame prediction PU sizes. The Apple A16 Bionic and Qualcomm Snapdragon 8 Gen 2 chipsets do not support the processing of 4x4 luma PBs in UHD4K@60fps videos. In this case, 4x4 PBs are only supported for chrominance samples, due to the color subsampling typically used to represent videos. The MediaTek Helio G95 chipset encoder showed a higher restriction at the supported intra-frame prediction PU sizes, supporting only the 16x16 PU size, regardless of resolution.

Based on the constraints and characteristics observed in each of the chipsets, it is possible to state that the Apple A16 Bionic chipset has the HEVC implementation with the smallest number of restrictions when compared with the HM implementation, followed by the Qualcomm Snapdragon 8 Gen 2, and finally the MediaTek Helio G95.

It is worth mentioning that additional constraints could be present in these encoders which cannot be found utilizing the proposed evaluation method. Some aspects of the video encoding process such as the search algorithm used in the ME process, the number of PUs partitions evaluated in the intra-frame prediction, and many others cannot be found in the bitstream header nor via the decoding process with the HM.

5 BD-Rate and time reduction analysis

This section discusses the BD-Rate and time reduction results for each of the three analyzed chipsets, obtained through the methodology in Section 3. As previously stated, the results were obtained using modified versions of the HM software, each version emulating the set of constraints presented in **Table 1**, for each of the three analyzed chipsets. As input, the

CTC test sequences, from classes A1, A2, B, C, and D were considered, in the four recommended QP values (22, 27, 32, 37) [Boyce *et al*., 2018].

Table 3 details the absolute time results, in minutes, used as baseline for calculating the time reduction obtained from the chipsets. As mentioned in Section 3, **Table 3** time results are the average for the four recommended QPs of each given sequence. As also stated in Section 3, the default HM software encoding time in the RA profile was used as a baseline for the time reduction results of the Apple A16 Bionic chipset, while the default HM software encoding time in the LDP profile was used as a beseline for the time reduction results of both Qualcomm Snapdragon 8 Gen 2 and MediaTek Helio G95.

It is important to note that absolute encoding times are influenced by the underlying server specifications in which the simulations were conducted. Thus, comparisons and analyses should be based on the relative percentage time reductions observed across the sequences.

Table 3. Absolute baseline times, in minutes, for each video.

Class	Video	HM-18.0 (RA) Time (m)	HM-18.0 (LDP) Time (m)
A1	Tango2	754	576
	FoodMarket4	747	586
	Campfire	811	696
	Average A1	**771**	**620**
A2	CatRobot	613	526
	DaylightRoad2	651	562
	ParkRunning3	905	745
	Average A2	**723**	**611**
B	MarketPlace	354	306
	RitualDance	400	315
	Cactus	262	239
	BasketballDrive	326	270
	BQTerrace	280	296
	Average B	**324**	**285**
C	BasketballDrill	58	52
	BQMall	62	60
	PartyScene	58	61
	RaceHorsesC	48	42
	Average C	**56**	**54**
D	BasketballPass	15	14
	BQSquare	14	16
	BlowingBubbles	12	14
	RaceHorses	10	10
	Average D	**13**	**13**
Average A1 and A2		**747**	**615**
Average B		**324**	**285**
Total Average		**336**	**283**

Table 4 presents the results for each video contained in classes A1, A2, B, C, and D, as well as the average results for each class and an overall average of all sequences. The results in **Table 4** encompass both the impact on BD-Rate and the reduction in encoding time of the evaluated constraints.

The results obtained for classes A1, A2, and B, which contain video sequences of UHD 4K (A1 and A2) and FHD (B) resolutions, are the most important in this investigation as they are the same resolutions supported by the smartphones analyzed by this work. Analyzing the class A1 and A2, which contain only UHD 4K (3840x2160 pixels) sequences, it can be observed a better coding efficiency, denoted by an average BD-Rate of 20.94%, in the Qualcomm Snapdragon 8 Gen 2 chipset.

Considering the constraints of the HEVC encoder in each chipset, it was expected that the Apple A16 Bionic would exhibit better coding efficiency than the Qualcomm Snapdragon 8 Gen 2, which was not reached by the results of classes that contain UHD 4K sequences (A1 and A2). The justification for this unexpected behavior may be related to the value of the search range considered in the emulation of each chipset, being that the search range of the Snapdragon 8 Gen 2 chipset is higher (15) than the one employed during the emulation of the Apple A16 Bionic chipset (8) (see **Table 1**), which greatly affects the encoding efficiency in the case of UHD 4K videos.

Also, the video “FoodMarket4” is primarily responsible for the increase in the average BD-Rate in class A1 for the Apple A16 Bionic emulation, with a BD-Rate of 54.29%, while the Qualcomm Snapdragon 8 Gen 2 emulation achieved a BD-Rate increase of 21.30% for the same sequence. Since “FoodMarket4” is an UHD 4K video, it naturally demands longer MV values for the ME process to achieve the best CB. So, a reduction in the search range reduces the SA, compromising the ME process, especially on UHD 4K videos. Since the search range used during the emulation of the Qualcomm Snapdragon 8 Gen 2 chipset (16) is double the one employed during the emulation of the Apple A16 Bionic chipset (8), the significant increase in BD-Rate for the sequences in class A1 may be justified by the encoder’s inability to find the best CBs due to the reduced SA, especially on the ”FoodMarket4” sequence.

A separate emulation of the Apple A16 Bionic chipset with a search range of 15, the same employed for the Qualcomm Snapdragon 8 Gen 2 chipset, was done. The results were significantly improved with an average BD-Rate of 23.85% for class A1 and 21.29% for class A2. The first value is still slightly higher than the one achieved by the Qualcomm chipset; however, this separate emulation demonstrates that an increase in the search range can drastically impact the BD-Rate of specific videos. The most notable case is the “FoodMarket4” sequence, which reduces the BD-Rate from 54.29% to 29.19% in this scenario, softening the average increase in BD-Rate for the A1 class in the context of the Apple A16 Bionic chipset.

Furthermore, regarding class A1, the MediaTek Helio G95 chipset presents the worst BD-Rate values for all sequences, a consistent result considering its higher number of constraints, as presented in **Table 1**. Overall, the average BD-Rate of videos from class A1 resides at 31.70% for the Apple A16 Bionic, 17.64% for Qualcomm Snapdragon 8 Gen 2, and 59.28% for MediaTek Helio G95.

Regarding the results for Class A2, both Apple A16 Bionic and Qualcomm Snapdragon 8 Gen 2 exhibit similar average coding-efficiency results. However, analyzing each video behavior in this class at the two chipsets, one can observe that the BD-Rate varies significantly according to the sequence and the emulated chipset. The “CatRobot” sequence achieves a smaller BD-Rate of 19.78% on the Apple A16 Bionic emulation, whereas at Qualcomm Snapdragon 8 Gen 2 it achieves

Table 4. BD-Rate and time reduction percentage results for the analyzed constraints emulated at the HM Software Version 18.0 over the CTC test video sequences.

Class	Video	Apple A16 Bionic		Qualcomm Snapdragon 8 Gen 2		MediaTek Helio G95	
		BD-Rate (%)	Time Reduction (%)	BD-Rate (%)	Time Reduction (%)	BD-Rate (%)	Time Reduction (%)
A1	Tango2	31.836	62.8	20.029	66.7	68.979	83.4
	FoodMarket4	54.295	61.6	21.309	65.2	74.313	82.9
	Campfire	8.97	51.1	11.607	55.3	34.567	81.6
	Average A1	**31.701**	**58.5**	**17.648**	**62.4**	**59.286**	**82.7**
A2	CatRobot	19.783	58.3	29.192	64.0	67.848	82.3
	DaylightRoad2	36.824	56.9	33.156	64.7	81.357	82.8
	ParkRunning3	17.187	57.8	10.351	59.5	31.260	82.7
	Average A2	**24.598**	**57.7**	**24.233**	**62.7**	**60.155**	**82.6**
B	MarketPlace	35.457	55.7	29.101	62.4	35.765	77.5
	RitualDance	20.519	60.0	17.443	61.7	28.127	77.4
	Cactus	17.862	50.8	38.117	59.2	52.366	75.9
	BasketballDrive	21.982	57.1	20.890	61.6	31.479	77.1
	BQTerrace	34.119	39.9	75.573	57.9	92.727	75.8
	Average B	**25.988**	**52.7**	**36.225**	**60.6**	**48.093**	**76.7**
C	BasketballDrill	26.542	51.5	43.038	58.2	61.570	75.9
	BQMall	22.818	51.7	28.878	59.8	44.596	76.1
	PartyScene	21.261	44.6	64.411	54.8	80.350	75.1
	RaceHorsesC	21.487	54.1	20.934	59.9	34.374	77.2
	Average C	**23.026**	**50.5**	**39.315**	**58.2**	**55.222**	**76.1**
D	BasketballPass	13.201	49.7	14.436	58.3	27.602	76.0
	BQSquare	23.822	36.1	121.300	51.4	135.643	73.2
	BlowingBubbles	23.992	40.5	46.456	55.2	60.173	74.9
	RaceHorses	16.029	49.2	18.850	58.3	33.406	76.1
	Average D	**19.261**	**43.9**	**50.260**	**55.8**	**64.206**	**75.1**
Average A1 and A2		**28.150**	**58.1**	**20.941**	**62.6**	**59.721**	**82.6**
Average B		**25.988**	**52.7**	**36.225**	**60.6**	**48.093**	**76.7**
Total Average		**24.631**	**52.1**	**35.004**	**59.7**	**56.658**	**78.1**

29.19%. The contrary happens at the "ParkRunning3" sequence, where the Qualcomm Snapdragon 8 Gen 2 achieves a better coding efficiency, with a BD-Rate of 10.35%, while the Apple chipset achieves a BD-Rate of 17.19%.

Analyzing the results from Class B, which contains five FHD (1920x1080 pixels) video sequences, the BD-Rate values are aligned with the expected for each of the chipsets, with the Apple A16 Bionic achieving the best encoding efficiency. This corroborates the justification provided in the previous paragraph, where for FHD videos the impact of reducing the search range is not as significant as in UHD 4K videos. Another noteworthy point is the sequence "BQTerrace", which shows significantly high BD-Rate values for all three emulated chipsets, reaching 75.57% in the Qualcomm Snapdragon 8 Gen 2 and 92.72% in the MediaTek Helio G95 emulations. Those exceedingly high values may be attributed to the GOP structure employed by the two chipsets, since in the case of the Apple A16 Bionic, with a GOP structure resembling the RA temporal configuration, the BD-Rate result is considerably lower, reaching 34.11%. The average BD-Rate of Class B in Apple A16 Bionic chipset is 25.98%, Qualcomm Snapdragon 8 Gen 2 reached 36.22% and MediaTek Helio G95 achieved 48.09%.

In Classes C and D, which contain videos at resolutions of 832x480 and 416x216 pixels, respectively, there is a noticeable degradation in the average coding efficiency of the Qualcomm Snapdragon 8 Gen 2. It illustrates that the constraints applied to this chipset are geared towards processing high-resolution videos, such as the ones available in class A1, which is also reinforced by its best performance in the A1 class. On the other hand, the MediaTek Helio G95 results demonstrate exceedingly high BD-Rate values for low-resolution sequences, as can be seen in the "BQSquare" and "PartyScene" videos that reach BD-Rate values of 135.64% and 80.35%, respectively. Finally, the Apple A16 Bionic chipset exhibits homogeneous BD-Rate values across all sequences in classes C and D.

The only related work in the literature that performs a similar coding-efficiency analysis on the HEVC encoder embedded in mobile chipsets is [Costa *et al*., 2024]. In that study, the authors analyzed the set of constraints and the BD-rate degradation present in the HEVC encoder integrated into the Apple A15 Bionic chipset from the iPhone 13 smartphone. Based solely on the constraints examined in [Costa *et al*., 2024], it can be stated that both the Apple A15 Bionic and Apple A16 Bionic feature a very similar set of limitations, differing only in the intra prediction modes supported for UHD 4K videos. However, the results in [Costa *et al*., 2024] report a BD-Rate increase of 19.05% for classes A1, B, C and D, which is 5.58% lower than the average BD-Rate obtained in this study for the Apple A16 Bionic, for the same classes (24.64%).

The reason for this discrepancy lies in the search range approximation of 32 adopted in [Costa *et al*., 2024], which

is four times greater than the search range of 8 used in the present study for the Apple A16 Bionic. A search range of 32 is considerably less precise than the approximation with 8, as demonstrated by **Table 2**, especially given that both chipsets generate the same MV ranges ([-316, 315]) [Costa *et al*., 2024]. Therefore, it can be stated that the BD-Rate results in [Costa *et al*., 2024] are optimistic, whereas the results presented in this paper more accurately reflect the actual search range of the chipset. To confirm that the observed BD-Rate discrepancy is indeed due to the search range approximation, an emulation of the Apple A16 Bionic using HM with a search range of 32 was performed with the same sequences used in [Costa *et al*., 2024], resulting in a BD-Rate of 19.04%, which supports the hypothesis.

In summary, the average BD-Rate values for all classes are 24.63%, 35.00%, and 56.65% for the Apple A16 Bionic, Qualcomm Snapdragon 8 Gen 2, and MediaTek Helio G95 chipsets, respectively. Those results showed a high degradation in the coding efficiency of the HEVC encoder. The HEVC was developed to double the coding efficiency in relation to its predecessor, the AVC. This means that the HEVC encoder implemented at the MediaTek Helio G95 chipset, for example, has an average coding efficiency similar to an AVC encoder.

The results in this paper also demonstrated that by implementing the constraints observed in each of the smartphone encoders into HM, it is possible to achieve a reduction in encoding time of 52.1% for the Apple A16 Bionic, 59.7% for the Qualcomm Snapdragon 8 Gen 2, and 78.1% for the MediaTek Helio G95. These time reduction percentages are directly related to the level of the constraints employed at the emulated HM encoder. This also demonstrates how the high-end mobile chipsets handle the computational effort of the HEVC standard, and what are the most common constraints employed to achieve real-time video encoding of high-resolution videos in these devices.

Another important conclusion from the observed results is that all commercial chipsets were designed to reduce HEVC encoding time by at least 50%. In fact, the results in **Table 4** showed a reduction in computational time ranging from 52.1% to 78.1% when considering the average results for all evaluated resolutions and all chipsets.

The HEVC dedicated hardware design papers in the literature such as [Porto *et al*., 2019; Conceição *et al*., 2015; Leme *et al*., 2019; Perleberg *et al*., 2024; Hu *et al*., 2017; Bubolz *et al*., 2018; Chuen-Ching and Li, 2017; Singhadia *et al*., 2020; Pastuszak and Trochimiuk, 2016; Park *et al*., 2016; Singh and Ahamed, 2018; He *et al*., 2015; Cai *et al*., 2022; Zhang and Lu, 2019; Penny *et al*., 2020; Porto *et al*., 2021; Perleberg *et al*., 2018; Afonso *et al*., 2019] show BD-Rate increases ranging from 0.3% to 24.16% from several video coding stages, as detailed by **Table 5**. The BD-Rate results for the chipsets presented in this section show significantly higher values, ranging from 24.63% (best average result) to 56.65% (worst average result), indicating that the impact of these literature solutions is in line with the commercial reality of HEVC encoders available at high-end mobile chipsets.

However, it should be stated that these works report BD-Rate increases regarding the application of their proposed solution onto a specific video coding stage, while the results provided by this paper for the chipsets refer to the entire HEVC encoding process. Therefore, solutions with an already high BD-Rate impact, such as [Penny *et al*., 2020; Porto *et al*., 2021; Perleberg *et al*., 2018; Afonso *et al*., 2019], can introduce even more BD-Rate impact when integrated into a commercial HEVC encoder hardware implementation.

The results also revealed an interesting observation regarding the relation between encoding time gains and coding-efficiency losses. When considering all results together, the chipsets reduced the encoding time by 1.7% for each 1% increase in BD-Rate. The Apple A16 Bionic chipset reached the best results, with an average reduction of 2.1% in encoding time for each 1% increase in BD-Rate. The Qualcomm Snapdragon 8 Gen 2 ranked second, with a reduction of 1.7%, followed by the MediaTek Helio G95 at 1.4%. The results of this relation could serve as a valuable metric to guide the development of future works in the area since hardware designs that achieve a better ratio than those presented here could significantly contribute to current multimedia electronics applications.

At this point, it is important to emphasize that the BD-Rate and time reduction percentages were derived from an emulation of the HEVC constraints observed in the referenced chipsets encoders, applied over a software environment which is the HEVC reference software. The actual coding-efficiency losses achieved by the hardware of these devices could potentially be even higher due to factors such as memory bandwidth limitations, targeted bitrate, and real-time operation. Nevertheless, the video encoding constraints outlined in this paper are undoubtedly present in the HEVC encoders of these chipsets and play a crucial role in guiding future HEVC hardware encoder designs targeting mobile devices.

Moreover, all reference works cited above assess the BD-Rate degradation of their solutions through simulations performed using the HM software. Therefore, having the BD-Rate results obtained in the same way via HM simulation validates the presented methodology and is essential to al-

Table 5. BD-Rate results from different dedicated HEVC hardware solutions from the literature.

Work	Targeted Video Coding Stage	BD-Rate (%)
[Porto *et al*., 2019]	SAD Tree	0.30
[Conceição *et al*., 2015]	Transform	0.43
[Leme *et al*., 2019]	Transform	0.57
[Perleberg *et al*., 2024]	Motion Estimation	0.63
[Hu *et al*., 2017]	Motion Estimation	0.77
[Bubolz *et al*., 2018]	Video Transrating	0.81
[Chuen-Ching and Li, 2017]	Motion Estimation	0.99
[Singhadia *et al*., 2020]	Transform	1.44
[Pastuszak and Trochimiuk, 2016]	Motion Estimation	1.64
[Park *et al*., 2016]	Motion Estimation	1.70
[Singh and Ahamed, 2018]	Motion Estimation	2.00
[He *et al*., 2015]	Motion Estimation	2.07
[Cai *et al*., 2022]	Entropy Coding	2.22
[Zhang and Lu, 2019]	Intra Prediction	3.69
[Penny *et al*., 2020]	Motion Estimation	10.54
[Porto *et al*., 2021]	Motion Estimation	16.17
[Perleberg *et al*., 2018]	Motion Estimation	18.38
[Afonso *et al*., 2019]	Motion Estimation	24.16
Apple A16 Bionic	Entire encoder	24.63
Qualcomm Snapdragon 8 Gen 2	Entire encoder	35.00
MediaTek Helio G95	Entire encoder	56.65

low a fair comparison with previous and future works in the literature.

6 Conclusions

This paper presented a method to evaluate the coding efficiency of the HEVC encoders available at high-end smartphone chipsets, reproducing the results utilizing the HEVC reference software. Three smartphone chipsets from the world's leading manufacturers were evaluated: Apple A16 Bionic (iPhone 14 Pro), Qualcomm Snapdragon 8 Gen 2 (Samsung Galaxy S23 Plus), and MediaTek Helio G95 (Xiaomi Redmi Note 10S). Therefore, the paper first provided the set of constraints of each chipset, then, the HEVC reference software was used to emulate each set of constraints of the three analyzed chipsets. Finally, the coding-efficiency losses and encoding time results were extracted and compared with the results of the default HEVC reference implementation.

The coding-efficiency results show that the three analyzed chipsets operate within a wide range of average BD-Rate, ranging from 24% to 56% when compared to the default HEVC implementation, with significant variability according to the constraints from the chipset encoder, video content, resolution, and frame rate. The time reduction results range from 52% to 78% in the software-emulated versions of the chipsets encoders, representing a great reduction in the complexity of these mobile encoders compared to the HM software

The results provided by this paper establish a comparative benchmark for hardware designers and researchers, that can use these results as a comparison to assess the behavior of commercial HEVC video encoders in high-end smartphones. Moreover, the proposed method of evaluation can be replicated for any other mobile chipset encoder.

Declarations

Acknowledgements

The authors gratefully acknowledge the support provided by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), the Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul (FAPERGS), the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), the Universidade Federal de Pelotas (UFPel), and the Video Technology Research Group (ViTech) during the development of this research.

Funding

This research was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. It was also financed in part by the Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul – Brasil (FAPERGS), and by the Conselho Nacional de Desenvolvimento Científico e Tecnológico – Brasil (CNPq).

Authors' Contributions

Vítor Costa is the first author of this work. He contributed with data curation, formal analysis, methodology, project administration, software implementation and writing.

Murilo Perleberg is the second author of this work. He contributed with software implementation, validation, visualization, data curation, and writing.

Luciano Agostini is the third author of this work and co-advisor of both the first and second authors. He contributed with manuscript writing and revisions.

Marcelo Porto is the fourth author of this work and lead advisor of both the first and second authors. He contributed to conceptualization, investigation, validation, manuscript writing, and revisions.

Competing interests

The authors declare that they have no competing interests regarding the publication of this work.

Availability of data and materials

The captured video sequences and the HM software versions emulating each mobile device generated during the current study are made available at https://drive.google.com/drive/folders/1IH4ecRZ75H1vCyw5A-2HTZmX7YN8LkNW?usp=sharing

References

Abu, A. and Gunasekara, G. H. (2014). Power and performance analysis of multimedia applications running on low-power devices by cache modeling. *Multimedia Tools and Applications*, 72:207–230. DOI: 10.1007/s11042-012-1350-3.

Afonso, V., Conceição, R., Saldanha, M., Braatz, L., Perleberg, M., Corrêa, G., Porto, M., Agostini, L., Zatt, B., and Susin, A. (2019). Energy-Aware Motion and Disparity Estimation System for 3D-HEVC With Run-Time Adaptive Memory Hierarchy. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(6):1878–1892. DOI: 10.1109/TCSVT.2018.2847633.

Agostini, L., Filho, A., Staehler, W., Rosa, V., Zatt, B., Pinto, A., Porto, R., Bampi, S., and Susin, A. (2007). Design and FPGA prototyping of a H.264/AVC main profile decoder for HDTV. *Journal of the Brazilian Computer Society*, 13:25–36. DOI: doi.org/10.1007/BF03192399.

Apple (2022). iPhone 14 Pro technical specifications. Retrieved Feb. 22, 2024. Avaialble at:`https://support.apple.com/kb/SP875?viewlocale=en_US`.

Bitmovin (2023). The annual bitmovin video developer report. Retrieved May 6, 2024. Avaialble at:`https://bitmovin.com/video-developer-report`.

Bjontegaard, G. (2001). Calculation of Average PSNR Differences between RD curves. In *ITU-T SG16/Q6 VCEG 13th meeting, Austin*. Retrieved Jul. 21, 2025: `https://www.itu.int/wftp3/av-arch/video-site/0104_Aus/VCEG-M33.doc`.

Boyce, J., Suehring, K., Li, X., and Seregin, V. (2018). *JVET-J1010: JVET common test conditions and software reference configurations*. Available at: `https://www.researchgate.net/publication/326506581_JVET-J1010_JVET_common_test_conditions_and_software_reference_configurations`.

Bubolz, T., Conceição, R., Grellert, M., Zatt, B., Agostini, L., and Correa, G. (2018). Fast and energy-efficient HEVC transrating based on frame partitioning inheritance. *2018 IEEE 9th Latin American Symposium on Circuits and Systems (LASCAS)*, pages 1–4. DOI: 10.1109/LASCAS.2018.8399905.

Cai, Y., Fan, Y., Huang, L., Zeng, X., Yin, H., and Zeng, B. (2022). A Fast CABAC Hardware Design for Accelerating the Rate Estimation in HEVC. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(4):2385–2395. DOI: 10.1109/TCSVT.2021.3093579.

Canalys (2023). Global smartphone market drops just 1demand. Retrieved Feb. 22, 2024. Available at: `https://canalys.com/newsroom/worldwide-smartphone-market-Q3-2023`.

Chuen-Ching, W. and Li, G. L. (2017). Hardware-friendly advanced motion vector prediction method and its architecture design for high efficiency video coding. *Multimedia Tools and Applications*, 76(26):25285–25296. DOI: 10.1007/s11042-017-4500-9.

Cisco (2020). Cisco annual internet report (2018-2023) white paper. Retrieved Feb. 22, 2024. Available at: `https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html`.

Conceição, R., Araújo, A., Porto, M., Zatt, B., and Agostini, L. (2015). Hardware design of fast HEVC 2-D IDCT targeting real-time UHD 4K applications. *2015 IEEE 10th Latin American Symposium on Circuits and Systems (LASCAS)*, pages 1–4. DOI: 10.1109/LASCAS.2015.7250473.

Correa, G., Assuncao, P., Agostini, L., and da Silva Cruz, L. A. (2011). Complexity control of high efficiency video encoders for power-constrained devices. *IEEE Transactions on Consumer Electronics*, 57(4):1866–1874. DOI: 10.1109/TCE.2011.6131165.

Costa, V., Perleberg, M., Agostini, L., and Porto, M. (2024). Coding Efficiency and Time Evaluation of Apple A15 Bionic Chipset HEVC Encoder. *2024 IEEE 15th Latin America Symposium on Circuits and Systems (LASCAS)*, pages 1–5. DOI: 10.1109/LASCAS60203.2024.10506151.

Counterpoint (2024). Global Smartphone AP-SoC Market Share: Quarterly. Retrieved Feb. 22, 2024. Available at: `https://www.counterpointresearch.com/insight/global-smartphone-apsoc-market-share-quarterly`.

David, G., Cebrián-Márquez, G., and Cuenca, P. (2020). Rate-distortion/complexity analysis of HEVC, VVC and AV1 video codecs. *Multimedia Tools and Applications*, 79(39):29621–29638. DOI: 10.1007/s11042-020-09453-w.

de Carvalho, E., de Barros, G., de Paula Costa, L., Faria, R., Nunes, R., Lopes, R., and Zuffo, M. (2007). The Brazilian digital television system access device architecture. *Journal of the Brazilian Computer Society*, 13:95–113. DOI: 10.1007/BF03192404.

FFmpeg (2000). Fast foward moving pictures experts group. Retrieved Feb. 22, 2024. Available at: `https://ffmpeg.org/`.

Grellert, M., Bampi, S., and Zatt, B. (2016). Complexity-scalable HEVC encoding. *2016 Picture Coding Symposium*, pages 1–5. DOI: 10.1109/PCS.2016.7906356.

He, G., Zhou, D., Li, Y., Chen, Z., Zhang, T., and Goto, S. (2015). High-Throughput Power-Efficient VLSI Architecture of Fractional Motion Estimation for Ultra-HD HEVC Video Encoding. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 23(12):3138–3142. DOI: 10.1109/TVLSI.2014.2386897.

Hu, L., Gu, J., He, G., and He, W. (2017). A hardware-friendly hierarchical HEVC motion estimation algorithm for UHD applications. *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–4. DOI: 10.1109/ISCAS.2017.8050322.

ITU-T (2003). H.264: Advanced Video Coding. Retrieved Dec. 4, 2024. Avaialble at: `https://www.itu.int/itu-t/recommendations/rec.aspx?rec=6312&lang=en`.

ITU-T (2013). H.265: High Efficiency Video Coding. Retrieved Mar. 20, 2024. Avaialble at: `https://www.itu.int/rec/T-REC-H.265`.

Jeong, H., Yang, J., and Song, M. (2016). Video quality adaptation to limit energy usage in mobile systems. *IEEE Transactions on Consumer Electronics*, 62(3):301–309. DOI: 10.1109/TCE.2016.7613197.

Kim, E., Jeong, H., Yang, J., and Song, M. (2014). Balancing energy use against video quality in mobile devices. *IEEE Transactions on Consumer Electronics*, 60(3):517–524. DOI: 10.1109/TCE.2014.6937338.

Leme, M., Braatz, L., Palomino, D., Agostini, L., and Porto, M. (2019). Low-Power and High-Throughput Approximate 4×4 DCT Hardware Architecture. *2019 IEEE 10th Latin American Symposium on Circuits and Systems (LASCAS)*, pages 245–248. DOI: 10.1109/LASCAS.2019.8667581.

MediaTek (2020). MediaTek Helio G95. Retrieved Feb. 22, 2024. Avaialble at: `https://www.mediatek.com/products/smartphones-2/mediatek-helio-g95`.

Mi (2021). Redmi Note 10s technical specifications. Retrieved Feb. 22, 2024. Avaialble at: `https://www.mi.com/global/product/redmi-note-10s/specs/`.

Multimedia, S. (2015). Zond 265 - HEVC Analyzer. Retrieved Feb. 22, 2024. Avaialble at: `https://www.solveigmm.com/en/products/zond/`.

Park, S., Choi, B. G., Lim, I. G., Park, H. I., and Kang, S. W. (2016). An efficient motion estimation hardware architecture using Modified Reference Data Access (MRDAS) skip algorithm for High Efficiency Video Coding (HEVC) encoder. *2016 IEEE 6th International Conference on Consumer Electronics*, pages 85–89. DOI: 10.1109/ICCE-Berlin.2016.7684724.

Pastuszak, G. and Trochimiuk, M. (2016). Algorithm and architecture design of the motion estimation for the H.265/HEVC 4K-UHD encoder. *Journal of Real-Time Image Processing*, 12:517–529. DOI: 10.1007/s11554-015-0516-4.

Penny, W., Corrêa, G., Agostini, L., Palomino, D., Porto, M., Nazar, G., and Zatt, B. (2020). Low-Power and Memory-Aware Approximate Hardware Architecture for Fractional Motion Estimation Interpolation on HEVC. *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5. DOI: 10.1109/ISCAS45731.2020.9180976.

Perleberg, M., Afonso, V., Agostini, L., Zatt, B., and Porto, M. (2024). Memory-Centered Motion Estimation System With CTB-Based Full-Splitting Algorithm. *IEEE Transactions on Consumer Electronics*, 70(2):5007–5019. DOI: 10.1109/TCE.2024.3399123.

Perleberg, M., Afonso, V., Conceição, R., Susin, A., Agostini, L., Porto, M., and Zatt, B. (2018). Energy and rate-aware design for HEVC motion estimation based on Pareto efficiency. *Journal of Integrated Circuits and Systems*, 13(1):1–12. DOI: 10.29292/jics.v13i1.18.

Porto, R., Agostini, L., Zatt, B., Roma, N., and Porto, M. (2019). Power-Efficient Approximate SAD Architecture with LOA Imprecise Adders. *IEEE 10th Latin American Symposium on Circuits and Systems (LASCAS)*, pages 65–68. DOI: 10.1109/LASCAS.2019.8667554.

Porto, R., Perleberg, M., Afonso, V., Zatt, B., Roma, N., Agostini, L., and Porto, M. (2021). Fast and energy-efficient approximate motion estimation architecture for real-time 4K UHD processing. *Journal of Real-Time Image Processing*, 18:723–737. DOI: 10.1007/s11554- 020-01014-6.

Qualcomm (2022). Snapdragon 8 Gen 2 Mobile Platform. Retrieved Feb. 22, 2024. Avaialble at: `https://www.qualcomm.com/products/mobile/snapdragon/smartphones/snapdragon-8-series-mobile-platforms`.

Samsung (2023). Samsung Galaxy S23 & S23+ technical specifications. Retrieved Feb. 22, 2024. Avaialble at: `https://www.samsung.com/us/smartphones/galaxy-s23/specs/`.

Schierl, T., Stockhammer, T., and Wiegand, T. (2007). Mobile Video Transmission Using Scalable Video Coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9):1204–1217. DOI: 10.1109/TCSVT.2007.905528.

Scientiamobile (2018). Growing Support of HEVC or H.265 Video on Mobile Devices. Retrieved Jun. 5, 2024. Avaialble at: `https://www.scientiamobile.com/growing-support-of-hevc-or-h-265-video-on-mobile-devices/`.

Sebastiaan, V., Wallendael, G., Wolf, K., Cock, J., Lambert, P., and de Walle, R. (2012). An enhanced fast mode decision model for spatial enhancement layers in scalable video coding. *Multimedia Tools and Applications*, 58:215–237. DOI: 10.1007/s11042-010-0716-7.

Singh, K. and Ahamed, S. R. (2018). Low Power Motion Estimation Algorithm and Architecture of HEVC/H.265 for Consumer Applications. *IEEE Transactions on Consumer Electronics*, 64(3):267–275. DOI: 10.1109/TCE.2018.2867823.

Singhadia, A., Mamillapalli, M., and Chakrabarti, I. (2020). Hardware-Efficient 2D-DCT/IDCT Architecture for Portable HEVC-Compliant Devices. *IEEE Transactions on Consumer Electronics*, 66(3):203–212. DOI: 10.1109/TCE.2020.3006213.

Suehring, K. (2023). High Efficiency Video Coding (HEVC) Test Model (HM) - Version 18.0. Retrieved Dec. 4, 2024. Avaialble at: `https://vcgit.hhi.fraunhofer.de/jvet/HM`.

Sze, V., Budagavi, M., and Sullivan, G. (2014). *High Efficiency Video Coding (HEVC): Algorithms and Architectures*. Springer. DOI: 10.1007/978-3-319-06895-4.

Zhang, Y. and Lu, C. (2019). Efficient Algorithm Adaptations and Fully Parallel Hardware Architecture of H.265/HEVC Intra Encoder. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(11):3415–3429. DOI: 10.1109/TCSVT.2018.2878399.