Motion Vectors Merging: Low Complexity Prediction Unit Decision Heuristic for the Inter-Prediction of HEVC Encoders

Felipe Sampaio, Sergio Bampi
PPGC - PMicro - II
Federal University of Rio Grande do Sul
Porto Alegre/RS, Brazil
{fmsampaio, bampi}@inf.ufrgs.br

Mateus Grellert, Luciano Agostini, Julio Mattos
GACI - PPGC - CDTec
Federal University of Pelotas
Pelotas/RS, Brazil
{mgdsilva, agostini, julius}@inf.ufrgs.br

Abstract— This paper presents the Motion Vectors Merging (MVM) heuristic, which is a method to reduce the HEVC inter-prediction complexity targeting the PU partition size decision. In the HM test model of the emerging HEVC standard, computational complexity is mostly concentrated in the inter-frame prediction step (up to 96% of the total encoder execution time, considering common test conditions). The goal of this work is to avoid several Motion Estimation (ME) calls during the PU inter-prediction decision in order to reduce the execution time in the overall encoding process. The MVM algorithm is based on merging NxN PU partitions in order to compose larger ones. After the best PU partition is decided, ME is called to produce the best possible rate-distortion results for the selected partitions. The proposed method was implemented in the HM test model version 3.4 and provides an execution time reduction of up to 34% with insignificant rate-distortion losses (0.08 dB drop and 1.9% bitrate increase in the worst case). Besides, there is no related work in the literature that proposes PU-level decision optimizations. When compared with works that target CU-level fast decision methods, the MVM shows itself competitive, achieving results as good as those works.

Keywords- High-efficiency Video Coding, Inter-Prediction, Low-Complexity Decision, PU-Level Decision

I. INTRODUCTION

The recent advances in technology enabled many improvements regarding multimedia systems. Along with these advances, the consumer market constantly demands better quality media, such as higher resolution digital videos. When H.264/AVC [1] (the state-of-the-art video-coding standard) was created, only a few specific devices supported 1080p videos. However, the current scenario is different, since 1080p videos are now supported by a plethora of electronic devices and higher resolutions begin to catch the market’s attention. With that in mind, a group composed of video coding experts from ITU-T and ISO/IEC was formed under the name of Joint Collaborative Team on Video Coding (JCT-VC) [2]. The purpose of the JCT-VC was to develop a new video-coding standard with improved compressing tools focusing on high-resolution videos. From this collaboration, the High Efficiency Video Coding (HEVC) standard is emerging, aiming to double the compression rates when compared to H.264/AVC for the same video quality [3]. The final draft of this standard is expected to be released in January 2013.

The HEVC standard is still under development, but its current innovations already bring negative expectations regarding its complexity. The reference software for this standard, called HEVC Model (HM) [4], implements tools that are much more complex than those of H.264/AVC reference software, causing great concern when real-time applications are considered. In addition, this complexity is also a critical drawback for mobile devices, since a great amount of energy will be required to perform video coding/decoding tasks.

The inherent data structures defined in the HEVC standard can be highlighted as one of the main causes of its high complexity [5]. Frames are now divided into Treeblocks, which can be subdivided into Coding Units (CU). Furthermore, CUs can be recursively partitioned, forming a quad-tree structure, which is illustrated in Fig. 1. The CU quad-tree decision is originally performed using the Rate-Distortion Optimization (RDO) decision [6], which evaluates the bitrate and the objective quality (generally expressed by the PSNR, in dB) produced by every possible configuration. In other words, the prediction, residual coding and entropy coding stages are performed for every possible CU partition. During the prediction stage, each CU is once again divided into Prediction Units (PU), introducing a PU decision inside each node of the CU decision tree. Each PU decision also considers the RDO as decision strategy [5].

Figure 1. Example of quad-tree CU partitioning
Several works targeting complexity solutions for video encoders targeting the H.264/AVC can be found in the literature [7-9]. These works proposed new heuristic mode-decision algorithms to reduce the encoder complexity based on encoding information. However, these solutions cannot be directly imported to the HEVC context, due to the new coding structures that provide different data processing procedures when compared to H.264/AVC. There are also some works that propose solutions for the HEVC standard by relying on heuristic decisions in order to reduce the CU encoding time. In [10], encoding execution time is controlled according to a specific target complexity. Then, using historical information of past decisions, the decision core is able to avoid some CU size evaluations, assuming that they would not be chosen in an RDO decision. The work of [11] performs a similar decision: it cuts some CU nodes in the quad-tree based on decisions that where taken inside the current frame. Due to this fact, the author classified its decision as a frame-level decision.

The goal of this work is to heuristically accelerate encoding time by acting on a different level in the CU quad-tree decision: the PU level, more specifically in the inter-prediction step. HEVC defines that each PU must be predicted for the three different modes available: skip, intra- and inter-prediction. Some analyses point that the inter-frame prediction consumes from 60% up to 96% of the total encoding time. Such complexity is derived from Motion Estimation (ME), the main tool in the inter-frame prediction, which performs exhaustive searches in order to find the best match between the current block of pixels that is being encoded and an area of pixels from one or more reference frames. These results indicate immense degrees of optimizations, since each CU node in the quad-tree must call ME for each possible PU partition.

This work presents the Motion Vectors Merging (MVM) heuristic, a fast decision method for the inter-prediction PU decision, which aims to reduce the complexity associated with the encoding process. The MVM decides the PU partition by using heuristic approaches instead of calling the Motion Estimation (ME) procedure for every possible configuration. In doing so, the PU decision complexity can be significantly reduced, therefore reducing the complexity inside each node of the CU quad-tree. The MVM proposed heuristic checks all internal borders among the neighboring NxN partitions. Each border is analyzed and a heuristic defines if the partitions can be merged or not. If any merging occurs, the rate-distortion cost is evaluated for the decided PU partition, producing enough information for the CU-level decision.

The MVM fast decision method for inter-prediction PUs was implemented in the HM 3.4 test model [4], and rate-distortion results, as well as speed-up results, were generated. The obtained results show that MVM is capable of achieving a complexity reduction of 34% with insignificant loss in terms of quality (0.08 dB on average) and minor increase in bitrate (1.9% in the worst case). In addition, MVM can be easily integrated with other solutions that reduce encoding complexity targeting different coding structures, such as CUs.

The remainder of this paper is organized as follows: section 2 describes works found in the literature that also target complexity reduction in HEVC encoders; section 3 further explains the complexity issues concerning the coding process, presenting detailed analyses through different perspectives; section 4 is focused on thoroughly explaining the MVM fast PU decision, which is the main contribution of this work; on section 5, PNSR, bitrate and complexity results are shown; and section 6 finally discusses conclusions and future work.

II. RELATED WORK

To the best of our knowledge, there are no published works that present fast decision heuristic-based methods targeting the HEVC PU decision. All related work focus on avoiding the evaluation of every depth in the CU quad-tree decision. These works perform cuts in the quad-tree by keeping the history of decision during the current frame encoding or using the past frames decisions history.

The work presented in [10] proposes a complexity control in the HM test model in order to control the encoder execution time. Since the target of [10] is power-constrained applications, the complexity control checks the battery status and decides by running the encoding process on a specific complexity: higher complexity, higher energy consumption and best rate-distortion results, or lower complexity, lower energy consumption and worst rate-distortion results. This work avoids some CU evaluations based on past frames analyses: (1) one frame is full RDO processed and, afterwards, (2) based on the best rate-distortion results the decision core builds a history map of decisions that (3) will direct the future frame decisions using the already built map of past choices.

The work presented in [11] proposes a fast coding unit decision algorithm for frame-level or CU-level encoding process acceleration. The acceleration is achieved by skipping some CU evaluations based on past frames coding information and neighbor CU coding results. The PU-level decision is not considered in this work.

III. HEVC QUAD-TREE BASED DECISION

Differently from H.264/AVC encoders, which divide the frame in coding units of fixed 16x16 size blocks (called macroblocks), the first proposals for HEVC encoders already proposed more flexible coding units. Therefore, the fixed macroblocks are now treated as Coding Units (CU), which can assume dimensions larger than 16x16 pixels [5]. Each CU can be recursively sub-divided into blocks of 64x64, 32x32, 16x16 and 8x8 pixels. During prediction, each CU is once again divided into PUs. There are four possible PU partitions, 2Nx2N, 2NxN, Nx2N and NxN, where 2N is equivalent to the corresponding CU dimension. The PU partitions for a 64x64 CU and their corresponding sizes are illustrated in Fig. 2.
All these possible configurations are supported in order to enable patterns more suitable to different picture regions (homogeneous or heterogeneous parts). The HEVC first test model implements a CU size decision technique that persisted through all the consecutive test models: the quad-tree based approach [5]. The HM test model implements all possibilities by distributing them in a tree-based data structure. The decision algorithm evaluates all possibilities by performing the depth first search (DFS) in a recursive manner. Inside each tree node, the RD cost is calculated for all possible configurations: (a) the three prediction modes (skip, intra and inter modes) and (b) the four possible partition sizes (2Nx2N, 2NxN, Nx2N and NxN). In fact, the latest HM 3.4 test model [4], which is used in this work, restricts the skip modes to assume only 2Nx2N partition size and the intra modes to be partitioned in 2Nx2N and NxN [5].

Considering predictive (P) or bi-predictive (B) slices, the HM test model analyzes the skip, inter and intra modes for each CU node in the quad-tree. The proposals for the intra-prediction in HEVC considerably increase the computational complexity of this process when compared to H.264/AVC encoders (in the worst case, 35 prediction modes must be analyzed, against 9 in the worst case of H.264/AVC intra-prediction) [12]. The skip prediction mode defines a CU as a 2Nx2N PU that has no relevant information regarding transform coefficients, and all motion data, i.e., motion vectors, reference frames indices among others, are inferred from the motion merge technique [5]. Skip-predicted CUs bring no relevant complexity overhead. Lastly, the inter-prediction is still responsible for most of the computational complexity among all the other coding tools.

A. HM Complexity Analysis

Fig. 3 presents some analyses regarding the execution time spent in the ME compared to the remaining encoding tools. The HM execution profile shows that Motion Estimation, the main tool of the inter-prediction, spends more than 96% of the required encoding time when the Full Search algorithm is used. In addition, even when applying TZ Search, a fast algorithm that achieves a speed up of 23 times when compared to Full Search with small quality losses [4], ME takes more than 70% of the total encoding time.

B. RDO Decisions Experimental Analysis

The HM test model works with two possible complexity settings: Low Complexity (LC) or High Efficiency (HE). Additionally, there are other three configuration sets that define the coding/access parameters in the coding process: Intra Only, Random Access and Low Delay [13]. The combination of these two sets of specifications can generate six different test conditions parameters. In this work, since the focus is to propose a fast algorithm for the PU partition size decision in the inter-prediction, the Intra Only parameter set will not be considered in the analysis.

In order to evaluate the RDO decisions, four different video sequences were evaluated. Three of them are HD 1080p (1920x1080 pixels) test sequences: BasketballDrive (few details, medium amount of colors per frame, high motion), BQTerrace (medium details, few amount of colors per frame, medium motion) and Cactus (large amount of details and colors per frame, high motion) [13]. Fig. 4 shows some results related to the prediction modes that are chosen by the default RDO decision in the HM test model.

Considering the three video sequences, the most used prediction mode is the skip mode. On average, 71% of the samples are predicted using skip mode when the LD configuration is set and almost 80% are predicted as skip mode for the RA parameters. The inter-prediction generates the best results in 26% and 17% of the cases, for LD and RA respectively. The intra-prediction does not surpass 10% in the Basketball Drive. On average, only 3.6% of all CUs are intra-coded.

The complexity analysis shows that the inter-prediction spends from 63% up to 96% of the overall HM execution time. On the other hand, considering the best case in Fig. 4, where the inter-prediction is the best option in only 36%, the
highest complexity module is responsible for not the most part of the final codification modes. Based on these results, it is possible to conclude that heuristic-based approaches for the inter-prediction can significantly reduce the encoder complexity, and consequently the execution time, besides it can provide good results in the rate-distortion optimization.

C. Inter-Prediction PU Decisions

This analysis aims to statistically map the inter-prediction decision considering each PU size. Considering all CU dimensions (64, 32, 16 and 8), every partition size decision was counted. It is important to have a trace that shows the most used partition size when the HM is running with the Full RDO decision. With that in mind, Fig. 5 shows the PU partition size decision considering three video sequences.

It is important to notice that the 2Nx2N partition is the preferred partition choice, specially for smaller CU sizes. On average, 2Nx2N partitions are chosen in 85% of the PU partition decisions. The rectangular partitions, 2NxN and Nx2N, are the best option in 7% and 7.3% respectively. The NxN PU partition is used only for 8x8 CU with an inexpressive part of the total decisions, corresponding only to 0.2%.

The rate-distortion decision results show that the 2Nx2N PU partitions for four CUs of depth d+1 take priority when compared to NxN PU decisions for a d-depth CU. This fact is also considered for the proposed low complexity PU decision algorithm, i.e., 2Nx2N PUs take preference as much as possible. Besides, the 2NxN and Nx2N PU partitions have approximately the same occurrence, which is much lower than the 2Nx2N occurrence. This indicates that the proposed algorithm must prioritize the 2Nx2N choice when possible. In addition, 2NxN and Nx2N partitions should have with the same priority, and NxN PUs are used only in 8x8 CUs and should have the lowest priority of all.

IV. MVM FAST PU DECISION METHOD

As previously explained, ME is called three to four times during the inter-prediction inside each CU node in the quadtree. This characteristic is extremely unwanted considering real time and energy constraints. Therefore, solutions that aim to reduce the number of ME calls are very important to make real time HEVC encoding a feasible task. The Motion Vectors Merging algorithm described in this work was designed with such goal in mind. In order to accomplish this, a heuristic is applied to decide all PU sizes above NxN. When certain conditions are met, NxN partitions are merged into 2NxN, Nx2N or 2Nx2N partitions without performing ME operations for each one of the larger candidate PU sizes. If the conditions are not met, then ME takes place and the regular procedure holds for all possible partitions. It is important to emphasize that 2Nx2N PUs must take priority in the event of a tie, since these partitions will produce a lower bitrate. This is explained by the fact that greater PUs requires less motion information per pixel, such as motion vectors and reference frame index.

Fig. 6 and 7 show the MVM algorithm and the heuristic decision that was employed respectively. For each CU, the ME is initially performed for all four NxN partitions. With the vectors produced from ME, the heuristic function is applied to decide which partitions can be merged. The conditions that drive the merge decision are based on the vector similarity among the neighboring PUs. The PU decision tree is evaluated in a top-down manner. If all four PUs produced the same motion vectors, then the 2Nx2N is considered the best PU size and the ME is performed for that size only. When 2Nx2N does not meet the restrictions, the rectangular sizes are evaluated following the same steps, i.e., if two neighboring PUs produced the same motion vectors, then the corresponding PU size will be used and ME is applied to produce the best motion vector for that PU size. It is important to notice that ME must be applied for the PUs that are selected via heuristic decision in order to achieve better RD results. In doing so, motion vectors that are different from the ones that were used to decide whether PUs are merged can be produced.

Borders Merging Heuristic
1. function mergeBorders (A, B, C, D):
2. if ((A + B) + (C + D)) then
3. partSize ← 2Nx2N
4. else if ((A + D) then
5. partSize ← 2NxN
6. else if ((B + C) then
7. partSize ← Nx2N
8. else
9. partSize ← NONE
10. return partSize

Figure 6. Border merging algorithm.

Proposed Fast Inter PU Decision Algorithm
A ← motion vector block 0
B ← border between blocks 0 and 2 of a CU
C ← border between blocks 1 and 3 of a CU
D ← border between blocks 2 and 3 of a CU
1. function checkInterBestPU (cu):
2. //four motion vectors for each block in NxN PU partition
3. bestMV ← motionEstimation(cu, NxN)
4. A ← (bestMV[0] ← bestMV[1])
5. B ← (bestMV[0] ← bestMV[2])
8. bestPartSize ← mergeBorders(A, B, C, D)
9. resultsME ← motionEstimation(cu, bestPartSize)
10. return calcRDCost(resultsME)

Figure 7. Fast Inter PU decision algorithm
The MVM was integrated in the HM 3.4 test model. All rate-distortion evaluations were performed under the following settings: (a) low complexity configuration, (b) IPPP prediction structure, (c) search window dimension [±64] and (d) two search algorithms (Full Search and TZ Search). Fig. 8 presented the HM test model execution time reduction comparison between the full RDO based decision available in the HM test model and the proposed MVM fast decision method.

The results show that the proposed MVM fast inter PU decision method accelerates the encoding time by 35%. This optimization can be even higher when more complexity heuristics are inserted in the inter-prediction step. These results are expected, since most part of the encoding effort is spent on the inter-prediction, more specifically in the ME step, and these are the modules that the proposed algorithm targets to optimize. Besides, this encoding time reduction is achieved with minimum rate-distortion losses.

Fig. 9 presents the sets of rate-distortion evaluations for the two video sequences. The QP parameters were assigned to 22, 27, 32 and 37, as specified in the common test conditions document. For all cases, the PNSR drop and the bitrate increase are negligible. The average results when using the Full Search algorithm show a PSNR drop of only 0.08 dB with a bitrate increase of 1.37% on average. When the TZ Search is adopted, similar behavior is detected, by increasing only in 1.91% the bitrate and with 0.06 dB of PSNR drop. The rate-distortion results show that the proposed MVM fast decision method is highly efficient when compared with the full RDO implementation. For all cases, the RD graphic of the MVM is almost equal to the best corner case implementation, as presented in Fig. 9.

Due to the fact that the proposed algorithm acts in the PU level, it provides much more implementation flexibility. In other words, it is possible to combine the proposed inter PU-level decision with any decision that works in the above levels. This flexibility makes the MVM easily applicable in any HEVC encoder solution targeting low complexity. The low complexity that is provided by the use of the MVM is interesting when power-constrained devices are targeted. As in [3], where the complexity is adjusted accordingly with what the target application needs, the proposed PU decision algorithm could be selected when battery life is at a critical level, providing almost the same RD coding efficiency with significant complexity reduction.

Tab. 2 presents the comparison with related works. There are no published works that perform fast decision in PU level. All works assume that the full RDO is applied for all PU partition sizes. This way, the comparison will assume the CU-level heuristic algorithms that aggressively avoid some CU depths processing in order to save computation. The work [10] performs several complexity target analyses. The comparison in Tab. 2 considers only the 60% target, which is the nearest possible scenario that can provide a fair comparison.
Figure 9. Rate-distortion analyses on the proposed inter PU decision

<table>
<thead>
<tr>
<th>TABLE II. RELATED WORKS COMPARISON</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spec.</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>ΔPSNR (average)</td>
</tr>
<tr>
<td>ΔBitrate (average)</td>
</tr>
<tr>
<td>ΔExecution Time</td>
</tr>
</tbody>
</table>

One of the main important advantages of the MVM proposed method is that its implementation in high efficiency encoders is somewhat easy. Since the \( N \times N \) partitions are processed, the proposed algorithm just compares all motion vectors among them and, according to a Boolean expression, decides the best inter-prediction partition for the target PU. This way, this simple complexity provides encoding results that are equivalent to the CU-level fast decisions that inserts much more control flow in the encoding process.

Besides, MVM PU-level decision for the inter-prediction process can be extended for integration with other decisions that act on different coding structures levels. It is important because the ME is the most expensive step in high efficiency encoders and it is executed many times during the CU tree-based RD decision. By optimizing this bottleneck, the proposed decision can achieve an execution time reduction almost equal to the ones achieved by aggressive CU-level decisions.

VI. CONCLUSIONS

This paper proposed the Motion Vectors Merge algorithm, a novel fast heuristic decision for the inter-prediction PU trees of the emerging HEVC standard. MVM relies on saving the number of ME operations performed in each PU tree by merging \( N \times N \) PUs into larger ones without applying the RDO approach to make this decision. When PUs are merged through the heuristic decision, ME is applied for the merged PUs in order to achieve better RD costs.

Experimental results point an execution time reduction of 34% on average by just acting in the inter-prediction in PU level. This speed-up is achieved with insignificant losses in the rate-distortion encoding results.

As future work, different heuristics will be elaborated intending to find better complexity reduction at the same video quality. In addition, hardware implementations of the MVM will be designed to estimate the resources overhead introduced by this solution in HEVC hardware encoders.

REFERENCES