Title: Acceleration of a HEVC intra encoder with a parallel deep learning proposal
Authors: V. Galiano, H. Migallón, M. Martínez-Rach, O. López-Granado, M. P. Malumbres

Abstract

As is well known, each new video coding standard significantly increases in computational complexity with respect to previous standards. This is particularly true for the HEVC video coding standard, and it appears that the same will hold for the new incoming standards. Therefore, the use of techniques for reducing the required complexity without affecting the rate/distortion (R/D) performance is therefore always a topic of intense research interest.

Several works in the literature have tried to reduce the coding time using modern hardware accelerators, moving the most complex parts of the encoder to them. For example, in [6, 16], authors moved the computation of the motion estimation (ME) to a GPU, and in [1, 5, 14], authors proposed the computation of the ME process on an FPGA.

Other works in the literature have used parallel computing strategies to take advantage of the multicore processors available in modern clusters in order to speed up the overall encoding time of a video sequence [9, 10, 12, 13].

There are other works that have focused on source code optimization of specific parts of the HEVC encoder [2, 3, 4, 8]. In [4], authors reported on a fast decision mode based on CABAC rate estimation, while in [8], a fast coding tree unit (CTU) partitioning algorithm was developed that used the CTU texture to prone the CTU quad-tree structure. In [2, 3], a pre-analysis technique was proposed to reduce (a) the size of the search area; (b) the number of reference frames in the inter-frame prediction; (c) the number of prediction modes; and (d) the number of best candidates for the intra-frame prediction process.

Finally, several authors have developed machine learning approaches to reduce the coding time of the HEVC encoder [7, 17]. For example, to reduce the complexity of inter-mode prediction, Zhang et al. [17] proposed a coding unit (CU) depth decision algorithm with a three-level joint classifier based on a support vector machine (SVM) that predicts the splitting of three-sized CUs in the CTU partition. For the intra-mode process, Liu et al. [7] developed a convolutional neural network (CNN) approach that predicts CTU partitioning, and thus, reducing the complexity.

In this paper, we propose a combination of two powerful techniques to significantly reduce the complexity of the HEVC encoding engine: machine learning and parallel computing. In the first place, we use a version of the HEVC encoder that includes a convolutional neural network to speed up the CTU partitioning process and achieve large reductions in coding latency with negligible R/D performance losses. We then speed up this version even further by applying spatial parallelism to the encoder by means of a slice-based approach that exploits the multicore hardware capabilities of current processors.

The HEVC encoder version that includes a CNN is the one proposed in [15]. The main contributions that differentiate this proposal from others relate to the definition of a hierarchical CU partition map (HCPM) to represent the CU partition. Authors in [15] propose a deep CNN structure called an early-terminated hierarchical CNN (ETH-CNN) that can be trained to explore diverse patterns of the CU partition and reduce the complexity of the HEVC intra coding mode.

Using previous HEVC encoder approach, we have introduced a shared memory parallel approach based on Slices similar to the one proposed in [11]. Slices are a way to partition a frame, available in HEVC coding standard, into a set of consecutive CTUs in raster order.

In Figure 1, we show the flowchart of our hybrid HEVC encoder, called DPSA (Deep Parallel Slice Algorithm). In the first step, the master thread reads the configuration parameters, and following this, the HCPM must be computed for all CTUs and all frames. The partition map is stored in a file that will be accessed by all threads when the CTU partitioning tree is computed inside a slice. In this sense, the slice-based parallel algorithm is applied at a higher level. As shown in Figure 1, only the master thread reads the new frame to be encoded, in order to reduce both the number of disk accesses and memory requirements. The frame to be encoded will therefore be stored in the shared memory, and will be accessed only for reading. In fact, each thread will only access those CTUs that are part of the slice to be encoded by it. When coding the set of CTUs for the slice assigned to each thread, we use the prediction for the CU partition obtained from the deep learning approach. When all threads have encoded the slice assigned to them, they write their bit stream into the final bit stream, and this process must be done in the right order.

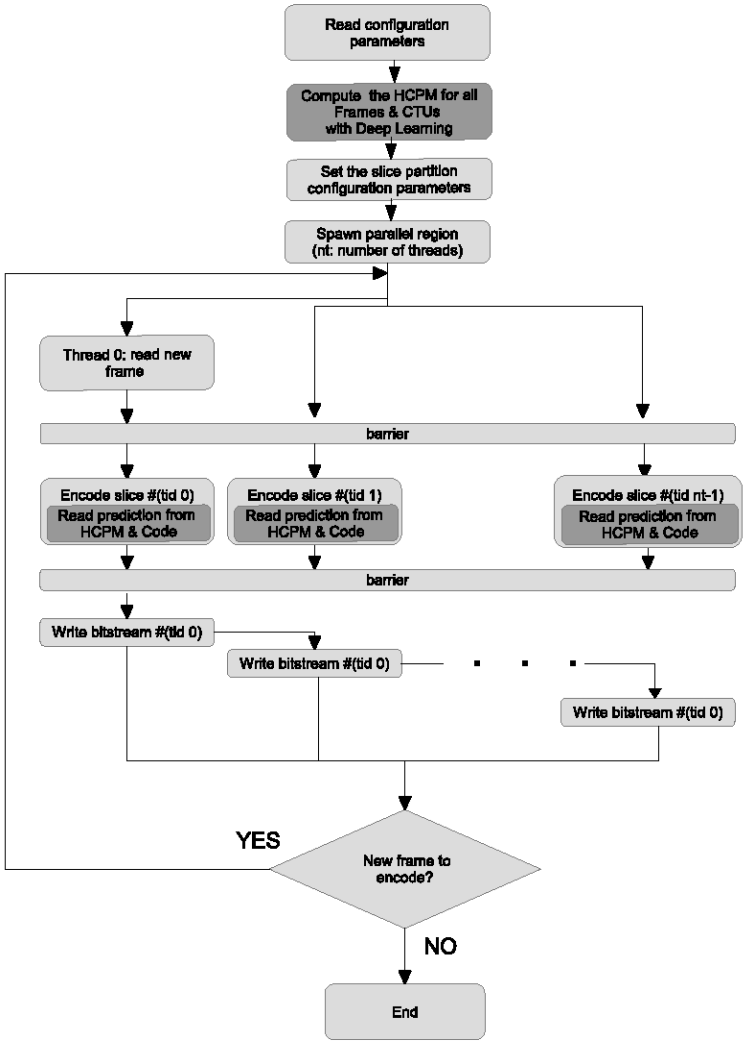

Figure 1. Hybrid HEVC encoder flowchart. DPSA algorithm is represented using light grey boxes, while the dark grey ones represent the contribution from deep learning.

After implementing and validating our hybrid HEVC encoder, we have carried out some experimental tests to measure the combined speed-up and the R/D penalty introduced by the machine learning and parallel computing techniques.

The results show that the use of machine learning alone is able to achieve speed-ups of up to 13x, and when the slice-based parallel approach is added, the overall speed-up reaches 35x. The R/D penalty in terms of the BD-rate metric depends on the video content and the number of threads used, and varies between 10% and 0.35% for the tested video sequences.

References

1. Alcocer, E., Gutierrez, R., Lopez-Granado, O., Malumbres, M.: Design and implementation of an efficient hardware integer motion estimator for an HEVC video encoder. J. Real-Time Image Processing 16, 547557 (2019). https://doi.org/10.1007/s11554-016-0572-4

2. Cebrian-Marquez, G., Martinez, J.L., Cuenca, P.: Adaptive inter CU partitioning based on a look-ahead stage for HEVC. Signal Processing: Image Communication 76, 97{108 (2019). https://doi.org/10.1016/j.image.2019.04.019

3. Cebrian-Marquez, G., Martinez, J.L., Cuenca, P.: Inter and intra preanalysis algorithm for HEVC. Journal of Supercomputing 73, 414432 (2019). https://doi.org/10.1007/s11227-016-1882-9

4. Chen, W., Wang, X.: Fast entropy-based CABAC rate estimation for mode decision in HEVC. SpringerPlus 756 (2016). https://doi.org/10.1186/s40064-016-2377-0

5. Haddar, R., Chaari, A., Kibeya, H., Ben Ayed, M.A., Masmoudi, N.: FPGA-based implementation of TZsearch algorithm for h.265/HEVC standard. In: 2017 18th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA). pp. 605-610 (Dec 2017). https://doi.org/10.1109/STA.2017.8314939

6. Lee, D., Sim, D., Cho, K.: Fast motion estimation for HEVC on graphics processing unit (GPU). J Real-Time Image Processing 12, 549562 (2016). https://doi.org/10.1007/s11554-015-0522-6

7. Liu, Z., Yu, X., Gao, Y., Chen, S., Ji, X., Wang, D.: Cu partition mode decision for HEVC hardwired intra encoder using convolution neural network. IEEE Transactions on Image Processing 25(11), 5088{5103 (Nov 2016). https://doi.org/10.1109/TIP.2016.2601264

8. Maazouz, M., Batel, N., Bahri, N., Masmoudi, N.: Homogeneity-based fast CU partitioning algorithm for HEVC intra coding. Engineering Science and Technology, an International Journal 22 (3), 706-714 (2019). https://doi.org/10.1016/j.jestch.2018.12.016

9. Migallon, H., Galiano, V., Piñol, P., Lopez-Granado, O., Malumbres, M.P.: Distributed Memory Parallel Approaches for HEVC Encoder. The Journal of Supercomputing pp. 1-12 (2016). https://doi.org/10.1007/s11227-016-1666-2

10. Migallon, H., Lopez-Granado, O., Galiano, V., Piñol, P., Malumbres, M.P.: Shared Memory Tile-Based vs Hybrid Memory GOP-Based Parallel Algorithms for HEVC Encoder, pp. 521-528. Springer International Publishing, Cham (2016). https://doi.org/10.1007/978-3-319-49583-5_40

11. Piñol, P., Migallon, H., Lopez-Granado, O., Malumbres, M.P.: Slice-based parallel approach for HEVC encoder. The Journal of Supercomputing 71(5), 1882-1892 (May 2015). https://doi.org/10.1007/s11227-014-1371-y

12. Piñol, P., Migallon, H., Lopez-Granado, O., Malumbres, M.: Slice-based parallel approach for HEVC encoder. J Supercomputing 71, 1882-1892 (2015). https://doi.org/10.1007/s11227-014-1371-y

13. Storch, I., Palomino, D., Zatt, B.: Speedup evaluation of HEVC parallel video coding using tiles. J Real-Time Image Processing (2019). https://doi.org/10.1007/s11554-019-00900-y

14. Vidyalekshmi V.G., Yagain, D., Ganesh Rao K: Motion estimation block for HEVC encoder on FPGA. In: International Conference on Recent Advances and Innovations in

Engineering (ICRAIE-2014). pp. 1-5 (May 2014). https://doi.org/10.1109/ICRAIE.2014.6909136
15. Xu, M., Li, T., Wang, Z., Deng, X., Yang, R., Guan, Z.: Reducing complexity of HEVC: A deep learning approach. IEEE Transactions on Image Processing 27(10), 5044-5059 (Oct 2018). https://doi.org/10.1109/TIP.2018.2847035
16. gang Xue, Y., you Su, H., Ren, J., Wen, M., yuan Zhang, C., quan Xiao, L.: A highly parallel and scalable motion estimation algorithm with GPU for HEVC. scientific programming 2017, 1-15 (2017). https://doi.org/10.1155/2017/1431574
17. Zhang, Y., Kwong, S., Wang, X., Yuan, H., Pan, Z., Xu, L.: Machine learning-based coding unit depth decisions for flexible complexity allocation in high efficiency video coding. IEEE Transactions on Image Processing 24(7), 2225-2238 (July 2015). https://doi.org/10.1109/TIP.2015.2417498