Alternative Reference Samples to Improve Coding Efficiency for Parallel Intra Prediction Solutions

Iago Storch[1], Nuno Roma[2], Daniel Palomino[3], Sergio Bampi[1]
[1]Graduate Program in Computing, Institute of Informatics, Federal University of Rio Grande do Sul, Porto Alegre - Brazil
[2]INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Lisboa - Portugal
[3]Video Technology Research Group, Graduate Program in Computing, Federal University of Pelotas, Pelotas - Brazil
{icstorch, bampi}@inf.ufrgs.br, nuno.roma@inesc-id.pt, dpalomino@inf.ufpel.edu.br

2024 IEEE 15th Latin America Symposium on Circuits and Systems (LASCAS) | 979-8-3503-8122-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/LASCAS60203.2024.10506142

***Abstract*—Exploring massive parallelism is a common strategy to mitigate the processing time of modern video encoding standards. Nonetheless, the existing data dependencies in some encoding tools pose difficult challenges to exploit such parallelism, especially during intra prediction, where the reconstructed adjacent blocks are used as references. Although some works made use of different reference samples to allow block-level parallelism in intra prediction, their proposals do not consider the variations caused by different bitrates, leading to some degradation in the output sequence. To deal with multiple bitrates more properly, this work proposes the application of image smoothing techniques to generate alternative reference samples that better represent the nuances of different bitrates. Experimental validations demonstrate that these improved references provide coding efficiency gains while still offering an equivalent parallelization opportunity.**

***Index Terms*—intra prediction, parallel video coding, digital image processing, GPU parallelization**

I. Introduction

Video coding algorithms are computationally burdensome, and the tools introduced in new standards increase the processing time [1]. Several works propose to mitigate these difficulties by bypassing some processing stages [2]–[7], yet, the speedup potential is limited. Achieving significant acceleration requires exploring modern parallel computing systems.

Recent video coding standards already incorporate tools to explore parallelism, such as slices, tiles, and wavefront parallel processing [8]–[10]. These tools explore coarse-grained parallelism, which is suitable for multicore CPUs. However, modern computing systems are heterogeneous [11], often equipped with GPUs, which are massively-parallel devices designed for fine-grained parallelism [12]. Therefore, one way to extract the maximum performance out of modern systems is distributing the workload throughout heterogeneous resources, which requires accelerating some tasks in GPU.

A large share of the complexity in modern coding standards comes from their flexible partitioning, as the same frame region is encoded multiple times with different partition combinations, seeking the best rate-distortion result. This way, the simultaneous processing of multiple block possibilities poses significant acceleration potential. Nonetheless, to maximize coding efficiency, video coding algorithms explore spatial and temporal redundancies, leading to data dependencies that create difficult challenges for conceiving block-parallel solutions.

This data-dependency issue is remarkably important in intra prediction. Since the intra prediction of one block depends on the reconstructed samples of adjacent blocks, blocks have an inherent processing order. Some methods [13]–[15] proposed to deal with this limitation by replacing the reconstructed samples of the neighbors with alternative ones that are always available, allowing multiple blocks to be processed in parallel. However, these works always employ the same set of alternative samples, which may not adapt well to different bitrates. Based on this observation, this work proposes a set of alternative reference samples that better represent the nuances of different bitrates, leading to a better coding efficiency with the same parallelization potential of related works.

II. Intra Prediction GPU Parallelization

Modern video coding standards employ a flexible partitioning structure, where a frame is first divided into a regular grid of large blocks, and each of these is recursively partitioned into smaller ones [16]–[18]. Then, a series of intra prediction tools are applied to each one of these blocks. Although there is no dependency between these tools, the prediction of one block depends on the reconstructed samples of adjacent blocks.

Due to the large number of prediction alternatives, video encoders usually start performing a fast and simplified evaluation of some modes, and only conduct the whole encoding process for a block after defining which modes are more likely to be optimal. Based on this framework, the authors of [13]–[15] use alternative reference samples during the mode decision of intra prediction to evaluate all blocks concurrently. After the final mode of each block is defined, the blocks are processed using the regular reconstructed references to produce the bitstream.

Although [13]–[15] propose different GPU parallelization techniques, they have one thing in common: using the original samples instead of the reconstructed ones as references to allow blocks to be independently processed. Usually, this strategy slightly degrades coding efficiency due to existing mismatches between the references used to choose the prediction mode and those used in the actual encoding. Naturally,

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) - Finance Code 001, FAPERGS, CNPq, and Fundação para a Ciência e a Tecnologia (FCT) under project UIDB/50021/2020.

this mismatch is aggravated with smaller bitrates as the reconstructed blocks are more affected by distortion. Conversely, the original samples are better references in larger bitrates. If this dependency on the bitrate could be modeled, then existing parallelization approaches (such as [13]–[15]) would improve their coding efficiency with the same acceleration potential.

Therefore, this work proposes a set of alternative reference samples that decouples adjacent blocks and accounts for the outgoing bitrate to represent the reconstructed samples more faithfully. When compared to the original frame samples, this set of alternative samples promotes the same parallelization opportunities while improving the resulting coding efficiency.

III. Proposed Alternative References in Intra Prediction

The quantization stage in video coding causes the most information loss, notably at low bitrates. High-frequency coefficients are more attenuated than low-frequency ones [19], such that the main difference between the original and reconstructed block lies in the attenuation of high-frequency information.

This observation raises the possibility of modeling the reconstructed samples with a smoothed version of the original ones, with the smoothing intensity controlled by the target bitrate (i.e., the quantization level). One way to achieve this is the classic low-pass convolution kernel [20] that allows a scalable blurring by adjusting the contribution of each sample. Another method is using the encoded samples of the previous frame since they are subject to the same quantization intensity.

Under this premise, this work evaluates a set of low-pass filtering kernels (convolution matrices) to assess their capability of replacing the regular references at different quantization levels. These kernels are depicted in Fig. 1 with a label. The convolution result is divided by the sum of the coefficients, that is, after convolving with 3×3_v0 the result is divided by 9 and rounded. Fig. 1 omits the division for clarity.

The smoothing intensity is mostly controlled by two factors: kernel dimension and weight distribution. For uniform weights, larger kernels lead to increased smoothing. For a fixed dimension, assigning larger weights to the central coefficient leads to less smoothing. Although this image filtering can be expensive in CPUs, this work targets computing systems equipped with GPUs, and image filtering is a classic example of a task where GPUs achieve great performance [21].

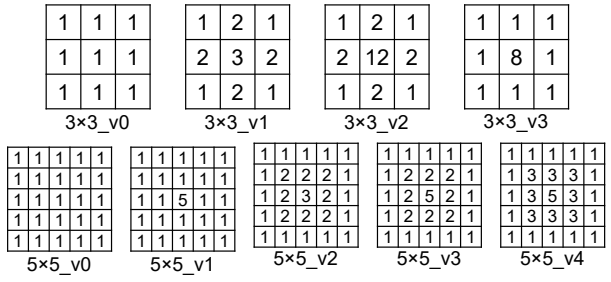

Fig. 1. Considered low-pass filtering kernels.

In addition to the low-pass filtered predictors, this work also compares with the original frame samples, and the reconstructed and filtered samples of the previous frame. The original samples serve as a performance baseline as they are widely used in the literature [13]–[15]. The reconstructed and filtered samples refer to the samples before and after the in-loop filtering stage, respectively. The previous frame samples are subject to a similar quantization and are easy to obtain.

IV. Experimental Evaluation

Considering that parallelization is required the most in larger video resolutions, this work considered the set of five 1080p sequences and six 4k videos from the VVC Common Test Conditions (CTCs) [22] (see Table I). The evaluations are conducted on the first 32 frames of each video.

To assess the suitability of the alternative reference samples in a real encoding scenario, the VVC standard was selected [17], and its VTM Software Encoder [23] implementation was used to perform the encoding using the *all_intra* configuration [22]. The intra prediction of VTM uses the regular reconstructed samples (i.e., before in-loop filtering) of adjacent blocks as references. To obtain results at a wide range of bitrates, quantization parameters (QPs) between 22 and 57 (in steps of 5) are used. The suitability of the alternative references is measured in terms of correlation with the regular references and coding efficiency, as discussed in the following sections.

A. Correlation Between Regular and Alternative References

Since there are many blocking possibilities and each prediction mode can use different samples, this evaluation considers all frame samples. First, the videos are encoded with the considered QPs and the reconstructed samples are extracted – these represent the outcome of the best encoding decisions according to VTM. The reconstructed and filtered samples of the previous frame are also extracted. The original samples are always available, and the smoothed samples are obtained by applying the convolution kernels from Fig. 1 to the original samples. The correlation between the current reconstructed frame and each of the alternative reference possibilities indicates the suitability of the alternative references in each QP.

The obtained correlation results are presented in Fig. 2, where the results for all videos and frames are averaged for simplicity. "*Original*" represents the original frame samples, "*PRec*" and "*PFil*" represent the previous frame reconstructed and filtered samples, and the remaining labels reference the convolution kernels from Fig. 1. A panned-in version of most alternative references between QPs 22 and 42 is also depicted.

Fig. 2 shows that the original samples present the best correlation for QP 22, and this correlation decreases for larger QPs. This is expected since larger QPs will introduce more degradation in the predicted blocks. Furthermore, the alternatives based on the previous frame present the smallest correlation, and this correlation increases for larger QPs. This occurs for a similar reason: smaller QPs produce reconstructed videos similar to the original, where the motion in the scene makes two successive frames less correlated; on the contrary,

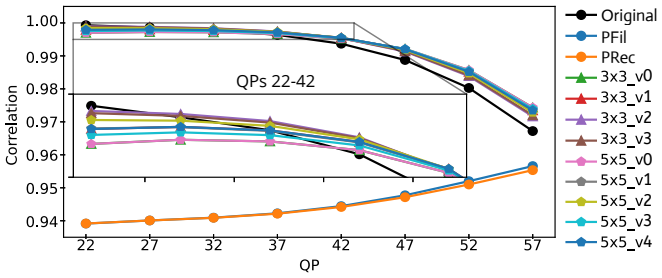

Fig. 2. Correlation between regular and alternative references.

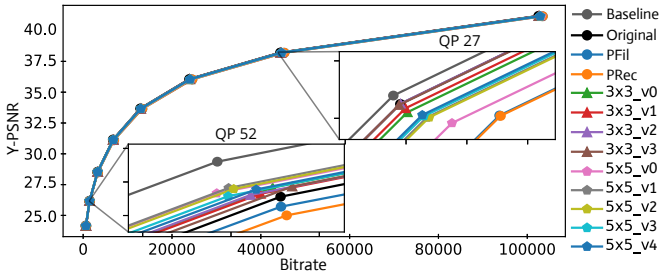

Fig. 3. Rate-distortion plot of *Cactus* video sequence.

larger QPs produce blurred reconstructed videos where the motion in the scene is less perceptible, and successive frames are more similar. The correlation of smoothing-based alternative references also varies with the QP. For smaller QPs (from 22 to 37), the samples produced by 3×3_v2 kernel present the largest correlation, closely followed by 3×3_v3 – note that these kernels produce the least smoothing due to their weight distribution. In QP 42, the largest correlation is obtained with kernel 5×5_v4. Finally, in QP 57 (not in the zoomed region), kernel 5×5_v0 provides the largest correlation – this kernel produces the most smoothing. These correlation results reinforce the idea that smoothed versions of the original frame can be better references for larger QPs than the original samples, which can improve the coding efficiency of parallel intra prediction methods such as [13]–[15].

B. Coding Efficiency of Alternative References

The VTM encoder breaks the intra mode decision into three main stages [23]. First, it uses Rough Mode Decision (RMD) to test most of the modes in a simplified manner, and a list of the best modes is created. Then, the Most Probable Modes heuristic is used to add some modes to this list. Finally, Rate-Distortion Optimization (RDO) thoroughly evaluates the modes in the list to decide on the best one. This section evaluates the coding efficiency of alternative references by swapping the regular references with alternative ones during RMD, when most modes are tested. Using alternative references changes the prediction signal and may lead RMD to create a distinct list of modes to be forwarded to RDO. Nonetheless, RDO uses the regular references to evaluate the modes' list thoroughly.

TABLE I
BEST ALTERNATIVE REFERENCE SAMPLE FOR EACH QP

Video	**22**	**27**	**32**	**37**	**42**	**47**	**52**	**57**
BasketballDrive	5v1	3v2	3v2	3v1	5v1	5v3	5v2	5v0
BQTerrace	Orig	Orig	Orig	3v2	3v2	3v0	5v2	5v0
Cactus	Orig	3v3	3v2	3v2	3v3	3v1	5v0	PFil
MarketPlace	3v3	3v2	3v1	3v3	5v1	3v3	3v0	5v1
RitualDance	Orig	Orig	Orig	3v3	5v3	5v0	5v2	5v1
Campfire	Orig	Orig	3v2	3v3	3v0	5v2	5v2	3v0
CatRobot	Orig	3v3	3v1	3v2	3v1	5v0	5v0	PRec
DaylighRoad	Orig	3v1	3v0	5v1	5v0	5v1	3v0	5v0
FoodMarket	3v3	5v3	5v1	5v2	5v1	5v2	5v0	5v3
ParkRunning	Orig	Orig	Orig	3v2	5v1	3v0	5v1	5v2
Tango	3v1	5v2	5v1	5v1	5v0	5v0	3v0	5v0
Most Common (Mode)	Orig	Orig	Orig 3v2	3v2	5v0 5v1 5v1	5v0	5v2	5v0

The resulting bitrate and distortion (in terms of Y-PSNR) results of each encoding were traced. As an example, the rate-distortion plot for video sequence *Cactus* is depicted in Fig. 3, where the horizontal and vertical axis represent the bitrate and Y-PSNR, respectively. The zoomed-in areas correspond to QPs 27 and 52. The performance of the unmodified encoder using the regular reference samples is represented by "Baseline" – this result is presented only as a reference, and the following discussion focuses solely on the alternative references.

Fig. 3 shows that different alternative reference samples present the best rate-distortion performance depending on the considered QP. In QP 27, for instance, the original samples and the alternatives 3×3_v2 and 3×3_v3 present the best performance. Note that the samples from the previous frame present a significantly worse performance, as foreseen by the evaluations from Section IV-A. In QP 52, on the contrary, the best rate-distortion performance is achieved by alternatives 5×5_v0, 5×5_v1, and 5×5_v2. Here, the original samples are considerably worse than most alternatives and only outperform the alternatives based on the previous frame. Similarly to the obtained conclusions from Fig. 2, the performance of alternatives based on the previous frame and with greater smoothing is better for larger QPs (lower bitrates). In summary, although the original samples are efficient in replacing the regular references at higher bitrates, the smoothing kernels proposed in this work prove themselves more efficient in replacing the regular reference samples at lower bitrates.

A summary of which alternative reference presents the best rate-distortion performance for each video and QP is presented in Table I, where 3v*i* and 5v*i* represent the *i-th* variation of a 3×3 or 5×5 kernel, respectively. Since many points are close together, the best reference for each QP is estimated dividing the Y-PSNR by the bitrate and selecting the alternative with the maximum ratio in each QP. The line "**Most Common**" represents the reference that was the most frequently tagged as the best in the respective QP (i.e., the statistical mode).

Although the results from Fig. 3 and Table I depict which alternatives are best suitable for each QP, they do not represent their overall coding efficiency. The BD-BR [24] metric is used to measure such relation, which represents the bitrate increase to maintain the same quality when encoding a video using the alternative references compared to using the regular references. The BD-BR results were computed twice: using QPs from 22 to 37, as recommended by the CTCs [22] (*conventional QPs*), and with QPs from 22 to 57 (*extended QPs*). Due to the many

TABLE II
CODING EFFICIENCY RESULTS AT DIFFERENT SCENARIOS AND QP RANGES

Video	BD-BR [%] for conventional QPs (range 22-37)							BD-BR [%] for extended QPs (range 22-57)						
	Orig	PRec	3v2	5v0	Hyb-0*	Hyb-1*	Best	Orig	PRec	3v2	5v0	Hyb-0	Hyb-1	Best
BasketballDrive	0.815	6.789	0.474	1.444	0.598	0.598	0.555	1.206	7.514	0.770	1.327	0.713	0.855	0.684
BQTerrace	0.449	2.075	0.543	2.915	0.436	0.436	0.432	1.046	2.427	0.972	2.962	0.921	1.194	0.837
Cactus	0.407	2.823	0.392	1.911	0.376	0.376	0.378	1.002	3.141	0.849	1.860	0.786	0.878	0.691
MarketPlace	0.284	3.082	0.221	0.636	0.236	0.236	0.241	0.485	3.237	0.410	0.633	0.425	0.429	0.411
RitualDance	0.474	7.812	0.464	1.726	0.453	0.453	0.452	0.787	7.132	0.746	1.300	0.670	0.709	0.650
Campfire	0.535	4.623	0.559	1.860	0.504	0.504	0.505	0.807	5.674	0.807	1.615	0.683	0.730	0.665
CatRobot	0.625	2.810	0.463	1.082	0.574	0.574	0.452	1.250	2.853	1.040	1.114	0.970	0.872	0.833
DaylightRoad	0.573	3.815	0.293	0.986	0.891	0.891	0.235	1.436	5.077	1.024	1.122	0.912	0.840	0.748
FoodMarket	0.413	3.864	0.242	0.184	0.270	0.270	0.132	0.589	3.510	0.353	0.301	0.373	0.323	0.221
ParkRunning	0.133	1.985	0.150	0.376	0.133	0.133	0.127	0.505	2.640	0.463	0.553	0.425	0.403	0.384
Tango	0.667	5.506	0.479	0.516	0.604	0.604	0.390	1.253	6.582	0.974	0.793	0.880	0.810	0.768
Average	**0.489**	**4.108**	**0.389**	**1.240**	**0.461**	**0.461**	**0.354**	**0.942**	**4.526**	**0.764**	**1.235**	**0.705**	**0.731**	**0.627**

* The coding efficiency for Hyb-0 and Hyb-1 in QPs 22-37 are identical because the same alternative references are used in these QPs.

alternatives used in this work, Table II presents the results for a subset of representative scenarios described in sequence: **Orig** uses the original samples in all QPs, as in [13]–[15]; **PRec** uses the reconstructed samples of the previous frame in all QPs; **3v2** and **5v0** use the samples obtained with the corresponding kernels in all QPs, since these are the best alternatives in most QPs; **Best** uses the best alternative reference (derived from Table I) for each QP in each video[1]; and **Hyb-0** and **Hyb-1** use a predefined set of alternative references for each QP, where **Hyb-0** is constrained to using only 3×3 kernels and **Hyb-1** also allows 5×5 kernels. In **Hyb-0**, the references for QPs in the range [22, 57] are [Orig, Orig, 3v2, 3v2, 3v0, 3v0, 3v0, 3v0]. For **Hyb-1**, the references for QPs in the range [22, 57] are [Orig, Orig, 3v2, 3v2, 5v1, 5v0, 5v0, 5v0]. These proposals are based on the results from Table I and on the premise that a greater smoothing models larger QPs more efficiently. Note that the proposals **Hyb-0** and **Hyb-1** share the same set of alternative references for QPs in the range [22, 37].

The results presented in Table II show that for the conventional range of QPs, the use of original frame samples as references results in a coding efficiency loss of 0.489% BD-BR. In this QP range, the proposed alternatives obtain a varying coding efficiency depending on the encoded video: when considering only "Orig", "3v2" and "5v0", each alternative is the best for at least one video. It also shows that if the curves of Fig. 3 were plotted for all videos, the lines would cross each other at different points. It is noteworthy that using the reconstructed samples from the previous frame ("PRec") leads to a significant coding efficiency loss of 4.108% BD-BR, while using the best alternative (among those herein proposed, "Best") leads to a coding efficiency of 0.354% BD-BR. Nonetheless, when a predefined set of alternatives is used ("3v2", "Hyb-0" and "Hyb-1" columns), a coding efficiency between 0.389% and 0.461% BD-BR is achieved. This represents a remarkable improvement over the adoption of the original samples commonly used in literature.

[1]The method used to estimate the best variation is an approximation. It might be possible to obtain a slightly better coding efficiency with different combinations of alternative references.

In the extended range of QPs, using the original samples as references leads to a coding efficiency of 0.942% BD-BR, while the best set of alternatives for each video results in a coding efficiency of 0.627% BD-BR. As in the conventional QP range, the hybrid approach leads to better coding efficiency than the original samples – "Hyb-0" and "Hyb-0" achieve coding efficiencies of 0.705% and 0.731% BD-BR, respectively.

It is worth noting that for the extended range of QPs, "Hyb-1" leads to a worse coding efficiency than "Hyb-0" for most videos. Likewise, for both ranges of QPs, "5v0" is worse than "3v2" and the hybrid approaches. This happens because "5v0" is the best alternative for some videos and QPs by a rather small margin, but it is significantly worse than other alternatives in some cases. Therefore, using it in non-optimal cases leads to large coding efficiency losses. Finally, the BD-BR results for the extended range of QPs are worse than those of the conventional range because of the natural difficulty of obtaining a good approximations in a wide range of RD points.

V. CONCLUSION

This work proposed a set of alternative reference samples to allow performing the intra mode decision of multiple blocks in parallel. While all works proposing GPU parallelization of intra prediction use the original frame samples as references during intra mode decision, the proposed references are based on a low-pass filtering of the original samples. In particular, it was shown that by adjusting the filter coefficients it is possible to obtain a better modeling of the reconstructed samples at lower bitrates. When compared to using the original samples, the proposed alternative references can reduce the coding efficiency penalty from 0.489% down to 0.354% BD-BR, or from 0.942% down to 0.627% BD-BR, depending on the range of bitrates. Although image filtering is usually expensive in CPUs, computing systems equipped with GPUs (the target of this work) can perform it very efficiently exploiting their massive parallelism with a very small timing overhead. This shows that the proposed alternative reference samples can improve coding efficiency while keeping the acceleration potential of techniques based on GPU parallelization of intra prediction.

REFERENCES

[1] F. Bossen, K. Sühring, A. Wieckowski, and S. Liu, "VVC Complexity and Software Implementation Analysis," IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 10, pp. 3765–3778, 2021, doi: 10.1109/TCSVT.2021.3072204.

[2] K. Wang, H. Liang, S. Zhang and F. Yang, "Fast CU Partition Method Based on Extra Trees for VVC Intra Coding," 2022 IEEE International Conference on Visual Communications and Image Processing (VCIP), Suzhou, China, 2022, pp. 1-5, doi: 10.1109/VCIP56404.2022.10008800.

[3] S. De-Luxán-Hernández, G. Venugopal, V. George, H. Schwarz, D. Marpe and T. Wiegand, "A Fast Lossless Implementation Of The Intra Subpartition Mode For VVC," 2020 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 2020, pp. 1118-1122, doi: 10.1109/ICIP40778.2020.9191103.

[4] A. Gou, H. Sun, J. Katto, T. Li, X. Zeng and Y. Fan, "Fast Intra Mode Decision for VVC Based on Histogram of Oriented Gradient," 2022 IEEE International Symposium on Circuits and Systems (ISCAS), Austin, TX, USA, 2022, pp. 3028-3032, doi: 10.1109/ISCAS48785.2022.9937635.

[5] G. Wu, Y. Huang, C. Zhu, L. Song and W. Zhang, "SVM Based Fast CU Partitioning Algorithm for VVC Intra Coding," 2021 IEEE International Symposium on Circuits and Systems (ISCAS), Daegu, Korea, 2021, pp. 1-5, doi: 10.1109/ISCAS51556.2021.9401614.

[6] Q. Zhang, Y. Wang, L. Huang and B. Jiang, "Fast CU Partition and Intra Mode Decision Method for H.266/VVC," in IEEE Access, vol. 8, pp. 117539-117550, 2020, doi: 10.1109/ACCESS.2020.3004580.

[7] F. Sagrilo, M. Loose, R. Viana, G. Sanchez, G. Corrêa and L. Agostini, "Learning-Based Fast VVC Affine Motion Estimation," 2023 IEEE International Symposium on Circuits and Systems (ISCAS), Monterey, CA, USA, 2023, pp. 1-5, doi: 10.1109/ISCAS46773.2023.10181659.

[8] C. C. Chi et al., "Parallel Scalability and Efficiency of HEVC Parallelization Approaches," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1827-1838, Dec. 2012, doi: 10.1109/TCSVT.2012.2223056.

[9] Y. -K. Wang et al., "The High-Level Syntax of the Versatile Video Coding (VVC) Standard," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 10, pp. 3779-3800, Oct. 2021, doi: 10.1109/TCSVT.2021.3070860.

[10] P. K. Papadopoulos et al., "On the Evaluation of Coarse Grained Parallelism in AV1 Video Coding," 2018 13th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP), Zaragoza, Spain, 2018, pp. 55-59, doi: 10.1109/SMAP.2018.8501888.

[11] S. Singh, "Computing without Processors: Heterogeneous Systems Allow Us to Target Our Programming to the Appropriate Environment.," Queue, vol. 9, no. 6, pp. 50–63, Jun. 2011, doi: 10.1145/1989748.2000516.

[12] M. Garland and D. B. Kirk, "Understanding throughput-oriented architectures," Communications of the ACM, vol. 53, no. 11, pp. 58–66, 2010.

[13] V. Galiano et al., "GPU-based HEVC intra-prediction module," in Journal of Supercomputing, vol. 73, pp. 455-468, 2017.

[14] S. Radicke et al, "A Parallel HEVC Intra Prediction Algorithm for Heterogeneous CPU+GPU Platforms," in IEEE Transactions on Broadcasting, vol. 62, no. 1, pp. 103-119, 2016.

[15] I. Storch, N. Roma, D. Palomino, S. Bampi, "GPU Acceleration of MIP Intra Prediction in VVC," European Signal Processing Conference (EUSIPCO), Helsinki, Finland, 2023, pp. 600-604, doi: 10.23919/EUSIPCO58844.2023.10290037.

[16] G. J. Sullivan, J. -R. Ohm, W. -J. Han and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1649-1668, Dec. 2012, doi: 10.1109/TCSVT.2012.2221191.

[17] B. Bross et al., "Overview of the Versatile Video Coding (VVC) Standard and its Applications," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 10, pp. 3736-3764, Oct. 2021, doi: 10.1109/TCSVT.2021.3101953.

[18] Y. Chen et al., "An Overview of Core Coding Tools in the AV1 Video Codec," 2018 Picture Coding Symposium (PCS), San Francisco, CA, USA, 2018, pp. 41-45.

[19] J. Pfaff et al., "Intra Prediction and Mode Coding in VVC," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 10, pp. 3834-3847, Oct. 2021, doi: 10.1109/TCSVT.2021.3072430.

[20] R. Gonzalez and R. Woods, "Digital Image Processing," 3rd edition, Pearson, 2007.

[21] R. Ansorge, "Programming in Parallel with CUDA: A Practical Guide," Cambridge University Press, 2022.

[22] F. Bossen, X. Li, V. Seregin, K. Sharman, K. Suhring "VTM and HM common test conditions and software reference configurations for SDR 4:2:0 10 bit video," JVET Document Y-2010, 2022.

[23] A. Browne, Y. Ye and S. Kim, "Algorithm description for Versatile Video Coding and Test Model 18 (VTM 18)," JVET Document AB-2002, 2022.

[24] G. Bjøntegaard, "VCEG-M33: Calculation of average PSNR differences between RD-curves,"3200/60 Document, Mar. 2001.