Mode-Adaptive Subsampling of SAD/SSE Operations for Intra Prediction Cost Reduction

Marcel Corrêa[1,2], Nuno Roma[3], Daniel Palomino[1], Guilherme Corrêa[1], Luciano Agostini[1]

marcelcorrea@ifsul.edu.br, nuno.roma@inesc-id.pt, {dpalomino, gcorrea, agostini}@inf.ufpel.edu.br

[1]Video Technology Research Group (ViTech), Universidade Federal de Pelotas (UFPel), Pelotas, Brasil
[2]Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense (IFSul), Bagé, Brasil
[3]INESC-ID, Instituto Superior Técnico (IST), Universidade de Lisboa (ULisboa), Lisboa, Portugal

***Abstract*—Modern video encoders, such as the recently proposed AV1 and VVC, offer significant encoding gains at the cost of a corresponding increase of the computational effort. This is the case of the adopted intra prediction techniques, comprehending an increased number of prediction modes and range. To mitigate this computational cost, the presented work proposes a new mode-adaptive algorithm that significantly reduces the number of SAD/SSE operations during intra prediction, by generating an optimized subsampling pattern adaptive to each prediction mode. The method can be applied to any video codec and, when applied to AV1, it led to an encoding time reduction and BD-BR impact of 15.36% and 0.6%, respectively, or 7.97% and –0.02%, depending on the selected subsampling parameters. When implemented in hardware, the proposed technique provides an effective reduction as high as 75% of both the area and power on the modified distortion calculation module.**

Keywords—Intra Prediction, Distortion Metric, AV1, Hardware Design

I. Introduction

Web-based video traffic has been pushing the telecommunication infrastructures to their limit, mostly because of the continuous increase in consumption of products that rely on videos, such as social media, streaming services and video conferencing platforms. According to Cisco, this type of traffic is expected to reach 325 Exabytes monthly in 2022, representing 82% of the global internet traffic [1]. However, this number tends to be even higher since this forecast was done prior to the COVID-19 pandemic, which led people to heavily depend on video services for their daily work, educational and social activities.

AOMedia Video 1 (AV1) [2][3] and H.266 Versatile Video Coding (VVC) [4][5] are two of the latest video coding formats made available. AV1 is a royalty-free and open-source project released in 2018 by the Alliance for Open Media (AOM) as the successor of VP9 [6], solving the uncertainty for many service providers when it comes to the licensing and deployment costs of other codec standards. The VVC standard, released in 2020, comes from a long line of successful video coding standards defined by a joint effort of the Moving Picture Experts Group (MPEG) and the Video Coding Experts Group (VCEG) that includes H.265 High Efficiency Video Coding (HEVC) [7], H.264 Advanced Video Coding (AVC) [8], and MPEG-2 [9].

All these modern video coding formats are based on the hybrid video encoder model, and the latest formats bring many novelties to each of the encoding stages when compared to older video formats. Naturally, most of these novelties were introduced to increase the coding efficiency. However, this is achieved at the cost of a significant increase of the required computational effort. As a consequence of this computational effort issue, video encoding is an infeasible task for software-based solutions when real-time processing and high resolutions are required. Hence, the wide adoption of next-generation video codecs highly depends on complexity reduction efforts through the development of fast algorithms suitable for specific hardware designs. Furthermore, even in hardware implementations, such as shown in [10], the distortion metric computation can represent as much as 37% of the gate count of an HEVC intra prediction module targeting Ultra High Definition (UHD) 8K resolution (7680×4320 pixels), which reinforces the need for specific hardware designs for the distortion metric calculation.

Both the intra and inter prediction stages of modern video encoders generate a high number of predicted blocks that must be evaluated locally with a compute-intensive distortion metric, such as the Sum of Absolute Differences (SAD) or the Sum of Squared Errors (SSE). Globally, the best candidates are evaluated by the even more compute-intensive Rate-Distortion Optimization (RDO) algorithm, which also relies on distortion metrics and is considered the bottleneck of an encoder. Therefore, the distortion metric computation significantly penalizes the processing time of the prediction stages on software encoders.

This work presents a mode-adaptive distortion metric subsampling to reduce the cost of the SAD/SSE operations of the intra prediction stage, allowing for faster encoding times on software encoders and for lower area and lower power dissipation on hardware encoders.

II. Motivational Analysis

Before the prediction process begins, a frame is firstly divided into several blocks of pixels. Both AV1 and VVC support square and rectangular block sizes in a range of 128×128 and 4×4 samples, although each format has its own exclusive partitioning structure.

In intra-frame prediction, each block is predicted from the reconstructed samples of previously encoded spatial neighbor blocks from the same frame, as shown in Fig. 1. The figure illustrates an AV1 example of a 4×4 block, which needs a total of 17 reference samples from previously encoded adjacent blocks to be predicted. The blue arrows illustrate the projection of predicted samples to indexes on the reference array following a 73-degree prediction angle.

To explore spatial redundancies in directional textures, AV1 and VVC support 56 and 65 different directional predictors, respectively. Moreover, to explore more homogeneous and non-directional patterns, both formats also support a variety of new prediction modes and enhanced versions of modes from their predecessors.

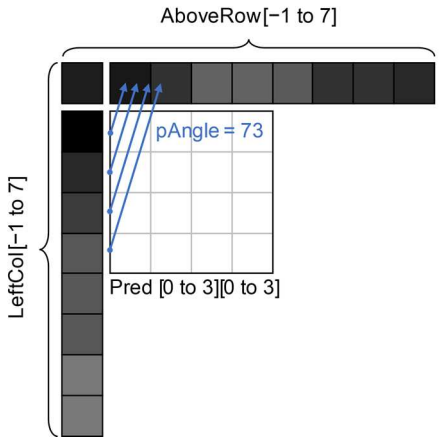

Fig. 1. An AV1 example of a 4×4 block *Pred* to be predicted using the reference arrays *LeftCol* and *AboveRow*.

For instance, in AV1, a single 128×128 superblock (SB) can be partitioned into 1,869 different subblocks according to a 10-way partition tree. Each block can be intra predicted by up to 66 different modes, which leads to a very high number of predicted blocks that must be compared to the original blocks using the SSE distortion metric to achieve the best coding for a single SB. The SSE between an original block o and a predicted block p of size $w \times h$ is the sum of the squared differences between every original and predicted sample, as defined in (1).

$$SSE_{o,p} = \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} \left(o_{i,j} - p_{i,j}\right)^2 \qquad (1)$$

However, since intra prediction only considers reference samples adjacent to the left and above the block to be predicted, more accurate predictions are expected to be obtained for positions spatially closer to the reference samples, and less accurate predictions as the spatial distance from the references increases. Indeed, this hypothesis was confirmed by experiments performed in the AV1 reference software *libaom* 2.0.0 [11], using the Class BCDE test sequences recommended in [12]. Fig. 2 illustrates the average error obtained for the AV1 intra prediction modes. In particular, it shows, the error for the Paeth, Directional 135, Directional 90 (Vertical), and Directional 120 (Horizontal) modes, with green areas representing small errors and red areas representing larger errors. It is worth noticing that those modes that access both reference samples, such as the Paeth and Directional 135, obtain accurate prediction samples along both the top and left edges, whereas modes that rely more on a single reference sample array, such as the Vertical and Horizontal modes, show more accurate predicted samples along one of the block edges.

This concept was already explored in other different parts of the AV1 encoder, namely: (i) the Recursive-based-filtering prediction mode, which breaks the predicted mode into 4×2 smaller blocks and use already predicted samples as reference for the next 4×2 blocks instead of using the reference arrays as the spatial distance increases [3]; (ii) the Inter-intra compound mode, which combines an intra-predicted block and an inter-predicted mode using a bilinear filter, but gives a higher weight to the intra-predicted block in regions closer to the reference samples [3]; and (iii) the Asymmetrical Discrete Sine Transform (ADST) [13], which was added to the AV1 specification because intra prediction residues are likely to be smaller near the reference arrays from where they are

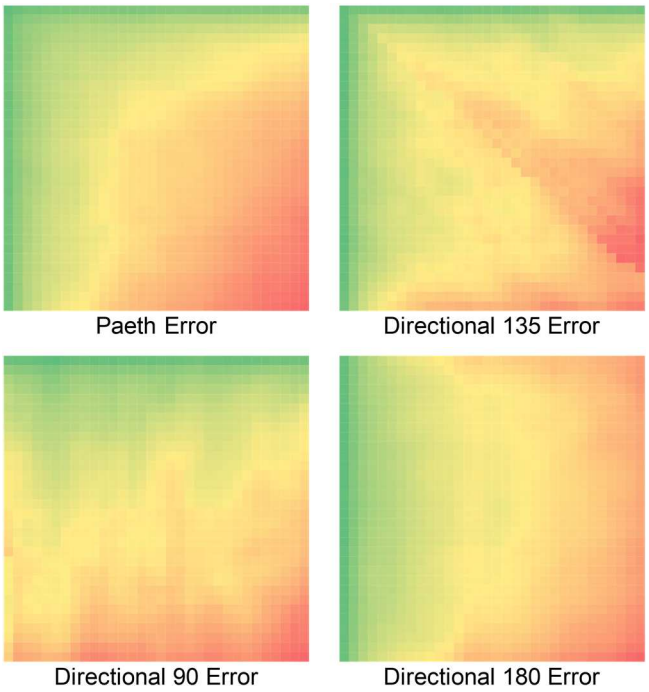

Fig. 2. Average error shown in form of heat maps for 32×32 blocks and the AV1 intra prediction modes Paeth, Directional 135, Vertical and Horizontal.

predicted. However, up until now this idea was never explored to optimize the distortion metric calculation and this is the main novelty of this work.

III. Proposed Mode-Adaptive Subsampling

The proposed mode-adaptive distortion metric subsampling algorithm reduces the overall cost of the SAD/SSE computation by applying a non-uniform subsampling technique, prioritizing the higher error areas generated by each predicted mode (warm colors in Fig. 2) over lower error areas. For each predicted mode and each block size, a subsampling mask is generated offline by the algorithm. The resulting masks can then be applied by a software or hardware encoder with a reduced number of operations.

To define a subsampling pattern, preliminary experiments were run in *libaom* 2.0.0 [11], showing promising results for subsampling masks that considers as low as 25% of the predicted samples during SSE computation, i.e., effectively cutting the number of operations by 75%. However, despite the fact that the higher error areas showed to be more relevant for discarding bad prediction candidates, it was observed that the rest of the predicted block should not be ignored. As a consequence, a portion of the lower error area is also considered in the mask generation method.

Hence, for a given prediction mode and block size, the proposed algorithm comprehends two stages, with the associated average error data (heatmaps) as input and a subsampling mask as output, where the error data must be obtained by exporting the prediction error of the unmodified encoder. Then, the first stage includes the positions of the predicted samples with the highest average error in the mask, by considering a ratio parameter *H_Area*. The second stage consists of a loop that iterates over the positions identified during stage 1. According to a ratio parameter L_Area, it unselects the same number of the previously marked positions, replacing them by other positions located in the

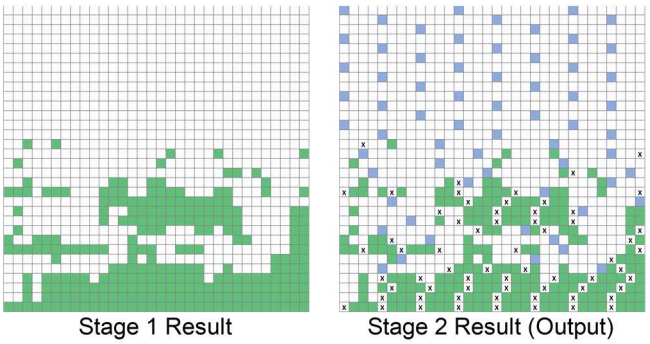

Fig. 3. Example of a 32×32 subsampling mask generation for the AV1 Vertical mode.

empty area identified during stage 1, by adopting a new uniform distribution pattern.

Fig. 3 illustrates the 32×32 subsampling mask obtained for the AV1 Vertical mode by the proposed algorithm with parameters *H_Area*=25% and *L_Area*=25%. In the first stage, the 256 positions from the heatmap with the highest error (see Fig. 2) are selected (marked in green). In the second stage, a loop removes 25% of the green positions, one at every four, for a total of 64 positions removed (marked with an X), assigning them to new positions in the originally empty area, one at every twelve positions (marked in blue). With the generated mask, every time the encoder evaluates the Vertical mode of a 32×32 predicted block, it will only use the positions marked in green and blue for the distortion metric computation.

This same algorithm can be used to generate subsampling masks for all intra prediction modes and block sizes of any modern video encoder, as long as the associated average error data (heatmaps) are generated to be used as input. However, different combinations of *H_Area* and *L_Area* should be tested to achieve the best results for each encoder.

IV. Results and Discussion

The following two subsections present, respectively, an analysis of how different parameter combinations can improve an AV1 software encoder, and how different subsampling degrees can reduce power and area in SAD/SSE hardware implementations.

A. Rate-Distortion and Encoding Time Results

A variety of combinations of the *H_Area* and *L_Area* parameters were tested on the *libaom* 2.0.0 encoder [11] with inter prediction disabled and using the test sequences recommended in [12].

Table I shows the resulting coding quality in terms of Bjøntegaard Delta Bitrate (BD-BR) for each color space (*Y* for luminance, *U* and *V* for chrominance) and total encoding time. Fig. 4 shows the quality results in terms of the combined BD-BR YUV metric, by following the $(6Y+U+V)\div 8$ relation.

As it can be observed in both Table I and Fig. 4, the uniform distribution of positions in the lower error area given by the *L_Area* parameter reduces the BD-BR impact notably. The *H_Area*=25% and *L_Area*=50% setup offers a significant encoding time reduction of 11.76% while increasing the BD-BR YUV by 0.71%. The *H_Area*=50% and *L_Area*=50% setup, on the other hand, offers slightly less than half of the encoding time reduction, but increases the BD-BR YUV by a negligible amount of 0.05%, making it an ideal choice for time saving without compromising the encoding quality.

TABLE I
BD-BR AND ENCODING TIMING RESULTS FOR DIFFERENT PARAMETER COMBINATIONS

H_area (%)	L_area (%)	BD-BR (%)			Enc. Time (%)
		Y	U	V	
25	0	2.08	1.26	1.07	87.99
	25	1.27	0.63	0.43	
	33	1.10	0.41	0.56	
	50	0.95	0.22	0.14	
50	0	0.72	0.42	0.41	96.25
	25	0.39	-0.25	-0.25	
	33	0.18	0.05	-0.21	
	50	0.13	-0.11	-0.26	

Class BCDE sequences only

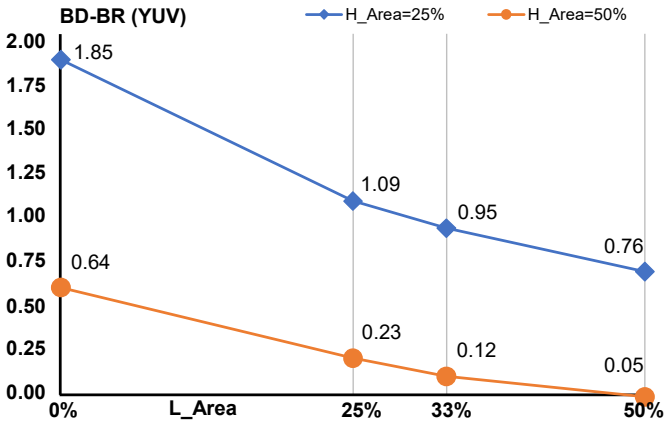

Fig. 4. BD-BR YUV curves for different parameter combinations for Class BCDE sequences.

TABLE II
BD-BR YUV PER VIDEO SEQUENCE FOR TWO SELECTED PARAMETER COMBINATIONS

Class	Sequence	BD-BR YUV (%)	
		H_Area=25% L_Area=50%	H_Area=50% L_Area=50%
A1 (UHD 4K)	*Tango2*	1.39	0.34
	FoodMarket4	0.46	0.02
	Campfire	0.27	−0.11
	A1 Average	**0.71**	**0.09**
A2 (UHD 4K)	*CatRobot*	0.40	0.00
	DaylightRoad2	0.51	0.08
	ParkRunning3	0.42	0.09
	A2 Average	**0.44**	**0.06**
B (1080p)	*MarketPlace*	1.14	0.12
	RitualDance	0.23	−0.19
	Cactus	0.98	0.16
	BasketballDrive	−0.18	−0.18
	BQTerrace	0.24	0.04
	B Average	**0.48**	**−0.01**
C (480p)	*BasketballDrill*	0.84	−0.35
	BQMall	0.45	0.09
	PartyScene	0.67	0.07
	RaceHorses	0.70	−0.03
	C Average	**0.66**	**−0.05**
E (720p)	*FourPeople*	0.66	−0.35
	Johnny	0.61	−0.22
	KristenAndSara	0.93	−0.04
	E Average	**0.73**	**−0.20**
ABCE Average		**0.60**	**−0.02**

TABLE III
SYNTHESIS RESULTS FOR DIFFERENT SAD TREE CONFIGURATIONS

	Full [10]	H_Area=50%	H_Area=25%
Required Operators	256 8-bit subs 128 8-bit adds 64 9-bit adds 32 10-bit adds 16 11-bit adds 8 12-bit adds 4 13-bit adds 2 14-bit adds 1 15-bit add	128 8-bit subs 64 8-bit adds 32 9-bit adds 16 10-bit adds 8 11-bit adds 4 12-bit adds 2 13-bit adds 1 14-bit add	64 8-bit subs 32 8-bit adds 16 9-bit adds 8 10-bit adds 4 11-bit adds 2 12-bit adds 1 13-bit add
Area (KGates)	49.8	27.9	12.4
Power (mW)	38.0	21.4	10.9

SAD Tree (TSMC 40nm, 529 MHZ)

Table II details the BD-BR YUV results for two selected parameter combinations when applied to several video sequences to ensure a performance evaluation of the algorithm on modern media content. This test set removes the low resolution (416×240 pixels) Class D sequences and includes the more demanding Class A1 and A2 video sequences [12], which have a resolution of UHD 4K (3840×2160 pixels), 10 bits per channel and a higher frame rate.

As it can be observed in Table II, the obtained average performance of the proposed algorithm increased when tested against the higher resolution content of the Class ABCE test set. The BD-BR YUV impact for the *H_Area*=25% and *L_Area*=50% setup was reduced from 0.76% to 0.60%, and the impact for the *H_Area*=50% and *L_Area*=50% setup from 0.05% became an improvement of −0.02%. Furthermore, the encoding time for the same parameter combinations also improved from 87.99% to 84.64%, and from 96.25% to 92.03%, respectively.

B. Hardware Implementation Results

In this subsection, the proposed subsampling masks applied to the intra prediction architectures proposed in [10][14] is discussed. The considered architectures are highly parallelized and target UHD resolutions. Synthesis and timing results of the SAD/SSE modules of these works are compared against the modified versions optimized by masks generated by the proposed mode-adaptive subsampling. In this context, the *L_Area* parameter is not discussed because it does not affect the total number of operations.

In [10], an HEVC intra prediction module targeting UHD 8K resolution at 120 frames per second (fps) is presented. This architecture processes all 35 prediction modes supported by the standard in parallel computing the SAD value of all 35 candidates also in parallel using trees of adders called SAD Trees. Each of the 35 SAD Trees is capable of comparing 256 samples (16×16 block or smaller) in a single cycle, whereas 32×32 blocks need to be divided into four smaller 16×16 blocks. The target throughput is achieved at 529 MHz.

Table III shows the number of operators needed by a SAD Tree in three different configurations. It also shows the area and power results of these SAD Trees when synthesized to the TSMC 40nm standard cells technology with the same target frequency used in [10]. It can be observed that the proposed algorithm reduces the number of operations by 50% and 75%, depending on the subsampling, and also reduces the depth of the tree, which leads to a shorter critical path and less bits used for the result. Both area and power can be reduced by up to 71% when using the more aggressive subsampling parameter.

TABLE IV
SYNTHESIS RESULTS FOR DIFFERENT SSE TREE CONFIGURATIONS

	Full [14]	H_Area=50%	H_Area=25%
Required Operators	64 8-bit subs 64 9-bit mults 32 17-bit adds 16 18-bit adds 8 19-bit adds 4 20-bit adds 2 21-bit adds 1 22-bit add	32 8-bit subs 32 9-bit mults 16 17-bit adds 8 18-bit adds 4 19-bit adds 2 20-bit adds 1 21-bit add	16 8-bit subs 16 9-bit mults 8 17-bit adds 4 18-bit adds 2 19-bit adds 1 20-bit add
Area (KGates)	33.0	16.7	8.1
Power (mW)	54.7	28.4	14.1

SSE Tree (TSMC 40nm, 1296 MHz)

In [14], an AV1 directional intra prediction module targeting UHD 4K at 60 fps is presented. The authors describe the architecture as being able to process 56 prediction modes in parallel, but a distortion metric design is not presented in this work. This design is not capable of processing entire blocks in a single cycle and, instead, it processes one row of up to 64 samples per cycle. This way, this design requires a total of 56 SSE Trees, and each must be able to compute the difference between 64 original and 64 predicted samples. The target throughput is achieved at 1,296 MHz.

Table IV shows the number of operators needed by an SSE Tree to be used in the design presented in [14], together with its area and power results. When compared to the less demanding SAD metric, the SSE Tree requires an extra level of multipliers, which in turn affects the size of the subsequent adders and the resulting critical path. This makes the proposed subsampling method even more important in an AV1 hardware encoder, and the obtained results show an area and power reduction of up to 75.5% and 74.2%, respectively.

V. CONCLUSION

This work presented a mode-adaptive subsampling capable of reducing significantly the number of SAD/SSE operations during intra prediction. The algorithm generates the subsampling masks offline to be used without any overhead added during encoding time. The subsampling masks can be generated and employed in any video encoder following any format. When applied to AV1, the benefits were observed on software and hardware solutions.

On software, depending on the selected parameters, the algorithm can reduce the encoding time by 15.36% while increasing BD-BR by 0.6%, or even reduce encoding time by 7.97% while decreasing BD-BR by −0.02%. On hardware, a reduction of around 75% can be achieved for both area and power. The conducted experiments also showed that the performance of the mode-adaptive subsampling is even greater for higher resolution videos, confirming that the algorithm is suitable for modern UHD content.

ACKNOWLEDGEMENTS

This study was financed in part by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* – Brazil (CAPES) – Finance Code 001. It was also supported by the CNPq and FAPERGS agencies.

This work was partially supported by *Fundação para a Ciência e a Tecnologia* (FCT) under projects UIDB/50021/2020 and PTDC/EEI-HAC/30485/2017.

REFERENCES

[1] Cisco Systems, “Global 2022 Forecast Highlights”, 2018. Accessed on: Oct. 2021. [Online]. Available: https://www.cisco.com/c/dam/m/en_us/solutions/service-provider/vni-forecast-highlights/pdf/Global_2022_Forecast_Highlights.pdf

[2] P. Rivaz, J. Haughton, "AV1 Bitstream & Decoding Process Specification," Alliance for Open Media, 2019. [Online] https://aomedia.org/av1/specification/

[3] J. Han et al., "A Technical Overview of AV1," in Proceedings of the IEEE, vol. 109, no. 9, pp. 1435-1462, Sept. 2021, doi: 10.1109/JPROC.2021.3058584.

[4] 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.

[5] Information technology: Coded representation of immersive media -- Part 3: Versatile video coding. ISO/IEC DIS 23090-3. 2020.

[6] D. Mukherjee *et al*., "A Technical Overview of VP9—The Latest Open-Source Video Codec," in SMPTE Motion Imaging Journal, vol. 124, no. 1, pp. 44-54, Jan. 2015.

[7] Information technology: High efficiency coding and media delivery in heterogeneous environments – Part 2: High efficiency video coding. ISO/IEC 23008-2. 2013.

[8] Information technology: Coding of audio-visual objects – Part 10: Advanced Video Coding. ISO/IEC 14496-10. 2003.

[9] Information technology: Generic coding of moving pictures and associated audio information – Part 2: Video. ISO/IEC 13818-2. 1996.

[10] M. Correa, B. Zatt, M. Porto and L. Agostini, "High-throughput HEVC intrapicture prediction hardware design targeting UHD 8K videos," 2017 IEEE International Symposium on Circuits and Systems (ISCAS), 2017, doi: 10.1109/ISCAS.2017.8050702.

[11] Alliance for Open Media. [Online] https://aomedia.googlesource.com/aom/+/refs/tags/v2.0.0

[12] M. Karczewicz, Y. Ye, “Common test conditions and evaluation procedures for enhanced compression tool testing,” Joint Video Experts Team (JVET), document JVET-W2017-v1, Jul. 2021. [Online] http://jvet-experts.org

[13] S. Parker *et al*., "On transform coding tools under development for VP10," Proc. SPIE 9971, Applications of Digital Image Processing XXXIX, 997119, 2016, doi: 10.1117/12.2239105.

[14] M. Correa, L. Neto, D. Palomino, G. Correa and L. Agostini, "ASIC Solution for the Directional Intra Prediction of the AV1 Encoder Targeting UHD 4K Videos," 2020 IEEE International Symposium on Circuits and Systems (ISCAS), 2020, doi: 10.1109/ISCAS45731.2020.9180526.