Exploring ERP Distortions to Reduce the Encoding Time of 360 Videos

Iago Storch*, Luciano Agostini, Bruno Zatt, Daniel Palomino
Programa de Pós-Graduação em Computação, Universidade Federal de Pelotas (UFPel), Pelotas - Brazil
{icstorch, agostini, zatt, dpalomino}@inf.ufpel.edu.br

***Abstract*—The encoding of 360 videos presents many challenges. These videos require a planar projection stage before encoding, and there are no studies on how projection interferes with the encoder decisions. Also, due to the increased resolution, encoding 360 videos is computationally costlier than conventional videos. In this work, we perform an extensive evaluation of how the texture distortions caused by the projection interfere with the encoder's behavior and exploit these interferences to propose multiple fast algorithms to accelerate the encoding. Experimental results show that the proposed algorithms reduce encoding time significantly with minor coding efficiency penalties, while the evaluations expose novel insights into the encoder behavior.**

I. Introduction

Digital videos are ubiquitous in modern society. A study conducted by CISCO points that by the year 2022, 82% of all internet traffic will be composed of digital videos [2]. In addition to that, 360 videos (also called spherical videos) are getting popular as well. A 360 video represents all points of view in a scene from a specific point in space such that the user can freely look around during playback. The immersion provided by 360 videos is being explored in many applications, including entertainment, education/training, and real estate business [3] [4] [5]. In this context, an efficient encoding of 360 videos is crucial to improve such applications.

Video coding is a computing-intensive task – all multimedia devices come with a dedicated chipset to perform it. Spherical videos push the challenge of video coding even further: since 360 videos present an entire 360º×180º field of view [6], they require an ultra-high resolution so the viewports maintain a reasonable resolution. In this context, it is crucial to have low-complexity solutions to perform the encoding of 360 videos.

There is no video coding standard to perform a spherical-based encoding. Instead, 360 videos are projected into a rectangular surface and encoded by conventional video encoders. There are several projection formats for 360 videos, such as the equirectangular projection (ERP), cubemap, and octahedral [7]. However, ERP is the most commonly used, and it is the one adopted in this work. When using the ERP projection, each latitude line from the sphere (i.e., each circle parallel to the equator line) is mapped into a row of the rectangle. However, since the latitude lines closer to the north/south poles have a radius smaller than the equator line, they must be horizontally stretched to fill the rectangular frame width. This is similar to the plate carrée projection used in geographical maps, where the textures around the equator are represented faithfully and textures closer to the poles are horizontally stretched. Notwithstanding, ERP projection does not stretch the textures vertically. Considering that conventional video coders are not designed to deal with these distortions, they may present a specific behavior when encoding such videos.

In this scenario, the objectives of this dissertation are obtaining a deep understanding of how the ERP distortions influence the encoder decisions and exploring it to reduce the encoding time of 360 videos. This work takes the High Efficiency Video Coding (HEVC) standard [8] as a case study since it presents the best coding efficiency available in consumer market devices. More specifically, the main contributions of this work can be summarized as follows:

- An evaluation of the correlation between ERP distortion intensity and multiple encoding decisions, namely: SKIP mode selection, motion vector length, fractional motion estimation redundancy, and block size selection;
- Four algorithms for accelerating different stages of the encoder based on texture distortion, summarized as: an early decision by SKIP mode, limitation of motion vector length, adjustment of fractional motion estimation precision, and discard of unlikely block sizes;
- Two top-level mechanisms to employ multiple algorithms simultaneously [1].

II. Overview on Video Coding and Related Works

In modern video coding standards, most of the encoding time lies in the interframes and intraframe prediction [9]. Also, modern standards allow recursive partitioning: the video is divided into a grid of regular blocks, and each block is divided into sub-blocks recursively. In HEVC, blocks in the regular grid are Coding Tree Units (CTUs) and have dimensions of 64×64 samples. Each CTU is recursively divided into Coding Units (CUs) using a quaternary tree, and the CUs have dimensions from 8×8 up to 64×64. In this approach, the prediction is performed for all intermediary blocks, and the same region (i.e., one CTU) is encoded multiple times with different block size combinations. Each combination of block size and prediction mode achieves a quality and a bitrate. The tradeoff between these two metrics (referred to as *coding efficiency*) is evaluated to decide which combination to use.

*This is a summary of a M.Sc. dissertation developed in an one-year period (Feb. 2019 – Feb. 2020) [1].

[1] All code generated in this dissertation was tracked in a public repository, available at: https://github.com/iagostorch/HM-16.16_360_InterOpt

The **intraframe prediction** is responsible for exploring the spatial redundancies inside the frame by representing a block based on spatially neighboring blocks or on a simplified version of the block itself. The HEVC standard provides 35 intraframe prediction modes: 33 angular modes used to predict strongly oriented textures and 2 special modes used to predict smooth or irregular textures. To reduce the computational burden of encoding 360 videos, works [10] and [11] propose accelerating the intraframe prediction of 360 videos. The authors of [10] propose a scheme to discard some unlikely block sizes and prediction modes based on a greedy-search approach, whereas in [11], the authors evaluate the vertical and horizontal complexity of samples in a region to bypass the evaluation of some prediction modes and block sizes.

The **interframes prediction** is responsible for exploring the temporal redundancies in the video, and it is achieved by representing one block based on blocks from previous frames. In HEVC, the interframes prediction of a block is performed mainly by integer motion estimation (IME), fractional motion estimation (FME), and SKIP mode. The IME employs a block-matching algorithm to find the most similar block inside a search range in a previous frame, and the displacement between the blocks is named a motion vector (MV). Then, FME uses interpolation to generate blocks in fractional-pixel positions around the best match from IME and selects the best one. In HEVC, FME generates blocks in positions displaced by ¼ and ½ pixels around the block. Finally, SKIP mode inherits the MV of a neighboring block and discards the prediction error – therefore, saving considerable bitrate.

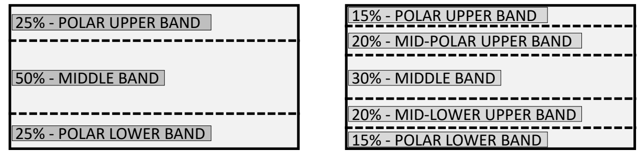
The works [12] and [13] present techniques to accelerate the interframes prediction. The technique proposed in [12] consists of limiting the minimum width of blocks and reducing the precision of FME throughout the frame to account for the distortion caused by ERP projection. In [13], the authors propose a KNN classifier to determine when a block should be partitioned into smaller blocks or not, considering the distortion and the number of bits required to encode the block.

III. Evaluation of ERP 360 Video Coding

Different stages of the encoder are evaluated aiming to find the correlation between encoder decisions and texture properties, and to support the development of fast algorithms.

To perform such evaluations, a set of conventional and 360 videos are encoded, and relevant intermediate information is traced. The encodings are performed according to the respective Common Test Conditions (CTCs) for conventional [14] and 360 [15] videos, using the recommended HEVC Test Model 16.16 (HM) [16] encoder along with 360Lib 5.0 [7].

The evaluation stage consisted of encoding the first 20 frames of four 360 videos (*AerialCity*, *Broadway*, *PoleVault*, and *SkateboardInLot* [15]) and four conventional videos (*NebutaFestival*, *PeopleOnStreet*, *SteamLocomotiveTrain*, and *Traffic* [14]). These videos are selected based on their texture and motion complexity to obtain a representative evaluation set. Some evaluations divide the frame *into three or five bands* to model the encoder's behavior in regions of distinct distortion, with positions and frame areas presented in Fig. 1.



(a) Three bands (a) Five bands

Fig. 1. Division of a frame into bands.

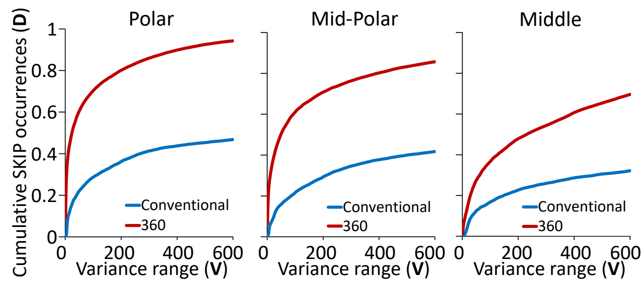


Fig. 2. Cumulative distribution of SKIP per block variance, CUs 32x32.

A. Evaluation of SKIP mode in ERP 360 video coding

Since the ERP projection makes the textures smoother in the top/bottom regions of the frame and the stretching intensity is reduced towards the center, some regions of the frame may be more susceptible to be encoded with SKIP mode, and the block homogeneity can interfere on SKIP selections. This is investigated by an evaluation crossing the SKIP occurrences with block homogeneity in different frame regions.

This evaluation uses a division into five bands to analyze regions of distinct projection distortion, and the homogeneity of a block is measured as the variance of its samples [17]. This set of variances is used to compute the cumulative distribution of SKIP mode occurrences in different scenarios. A scenario represents a combination of band position, CU size, video resolution, and QP value (a parameter from the encoder that controls compression rate). This distribution for conventional and 360 videos, considering CUs 32×32 and averaging all QP values [14] [15] and videos is presented in Fig. 2. The plots from Fig. 2 can be read as: each cumulative distribution $\boldsymbol{D}$ is associated to a variance $\boldsymbol{V}$, therefore, $\boldsymbol{D}$*%* of all SKIPs occur in blocks with variance in the range $\boldsymbol{[0,V)}$.

The results from Fig. 2 show that the distribution for conventional videos is flatter than for 360 videos for the same variance range, which means that a smaller share of SKIPs is selected for this variance range. When probing the polar bands of 360 videos at a variance 200, for instance, 80% of all SKIPs are selected in blocks with a variance in the range [0,200), whereas for conventional videos, only 36% of all SKIPs are selected in blocks with variance in the same range. A similar conclusion is drawn for the other bands. Finally, Fig. 2 shows that as we move from the polar to the middle band, the share of SKIPs in the same variance range is reduced significantly.

TABLE I
HORIZONTAL COMPONENT OF MVS IN RASTER SEARCH

	Upper		Middle		Lower	
	μ	σ	μ	σ	μ	σ
Conventional	**16.0**	24.4	**15.0**	24.0	**16.8**	26.4
360	**9.8**	15.2	**15.1**	19.8	**12.5**	21.3

B. Evaluation of TZS algorithm in ERP 360 video coding

The HM employs a fast IME algorithm named Test Zone Search (TZS) [16], and the Raster Search stage – a subsampled full search – is the most time-consuming stage of it [18]. Since ERP projection stretches the textures closer to the polar regions, these regions have more homogeneous textures that may influence on the Raster Search selections.

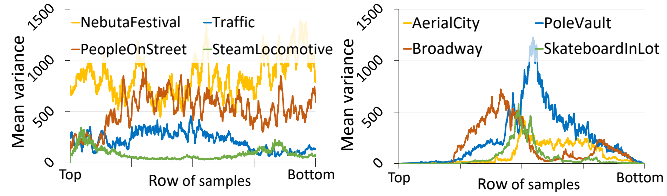
To assess the impact of ERP projection in Raster Search, an evaluation employing three bands is conducted to measure the horizontal component of MVs provided by Raster Search in different regions. This evaluation considers the absolute value of the MVs, and the results are presented in Table I, where μ and σ stand for the mean and standard deviation, respectively.

The results in Table I show that for conventional videos, the mean of horizontal component is very similar in all bands: 16.0, 15.0, and 16.8 for upper, middle, and lower bands, respectively. For 360 videos, however, the middle band presents a larger mean value when compared to the polar bands. Also, while the middle band of 360 videos presents an average value similar to conventional videos, the average in the polar bands is smaller than conventional videos. These results show that since the stretching makes the same texture spamming several blocks horizontally in polar bands, the encoder tends to select shorter MVs (pointing to closer blocks) in these regions since it represents a better bitrate-distortion tradeoff.

C. Evaluation of FME in ERP 360 video coding

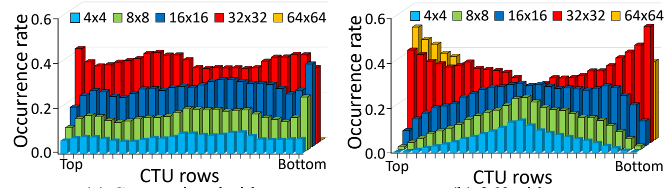
The FME employs an interpolation filter to generate intermediate samples between original samples. However, since the ERP projection stretches the textures horizontally similarly to horizontal FME, horizontal FME may be redundant in some regions of 360 videos. To assess this, the homogeneity of horizontally-neighboring samples in the frame is evaluated considering that if neighboring samples are similar, FME generates redundant fractional samples. This evaluation computes the variance of samples in a sliding 16×1 window (16 sequential samples in the same row), moving in steps of 4 samples. The variance of all windows in the same row is then averaged to represent the homogeneity of the row.

The results of this evaluation for conventional and 360 videos are presented in Fig. 3. This evaluation shows that in conventional videos, the mean row variance is distributed in an up-and-down disorderly fashion or in a homogeneous manner – i.e., the samples in any region of the video can present low or high similarity. For 360 videos, however, it is clear that the mean row variance is close to zero in the distorted polar regions, and it increases towards the middle of the frame – that is, the samples in polar regions of 360 videos are usually



(a) Conventional videos (b) 360 videos

Fig. 3. Variance of samples in each frame row.



(a) Conventional videos (b) 360 videos

Fig. 4. Distribution of block sizes.

very similar, and the variability increases towards the center of the frame. Also, the mean variance around the middle of 360 videos is similar to that of conventional videos.

D. Evaluation of intra block sizes in ERP 360 video coding

In HEVC, the intraframe prediction is performed in blocks called Prediction Units (PUs), which are sub-blocks of CUs and can have square dimensions from 4×4 up to 64×64. Since ERP projection creates homogeneous textures in some regions that may affect block sizes selection, the distribution of block size selections in the frame is computed. In this evaluation, the PU sizes selected for each row of the regular grid (i.e., for each row of CTUs) are traced and grouped to represent the distribution of PU sizes for each row of the frame. Each PU is assigned a weight proportional to its dimensions to assess the share of frame area encoded with each PU size.

The results for this evaluation on conventional and 360 videos are presented in Fig. 4. This distribution shows that in conventional videos, each PU size has a similar occurrence rate irrespective of position, and different PU sizes have different occurrences. For ERP 360 videos, on the contrary, the selection of PU sizes is directly related to the block position. In the polar regions, most of the blocks are encoded using large PU sizes while small PU sizes are rarely selected. In contrast, in CTU rows closer to the middle of the frame the occurrences of smaller PUs are increased at the expense of larger PUs. In the center of the frame, the PU size selection in conventional and 360 videos is very similar.

IV. PROPOSED FAST DECISION ALGORITHMS

Based on the previous analyses, a set of fast algorithms are proposed to reduce the encoding time of 360 videos, namely: *Early SKIP*, *Reduced SR*, *Reduced FME*, and *Reduced Intra Sizes*. The three first algorithms are applied in interframes prediction, and they relate as presented in Fig. 5. Reduced Intra Sizes is applied in intraframe prediction, but since it does not interact with other algorithms, its pipeline is not presented.

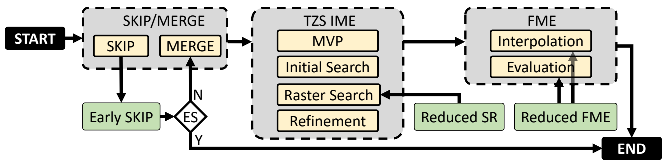


Fig. 5. Overview of the interframes prediction with ptoposed algorithms.

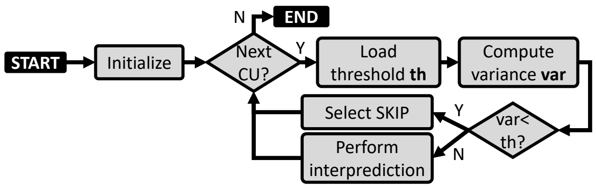


Fig. 6. Early SKIP algorithm.

The first step in interframes prediction is evaluating SKIP mode for the current CU. If *Early SKIP* performs an early decision (*'ES'* conditional), the remaining interframes prediction stages (MERGE, IME, and FME) are bypassed. Otherwise, *Reduced SR* scales down the search range of IME according to distortion intensity, and *Reduced FME* adjusts the precision of FME adaptively to projection distortion.

A. Early SKIP algorithm

Early SKIP algorithm accelerates the interframes prediction by adaptively performing an early selection of SKIP mode and bypassing the remaining interframes prediction stages. It explores the observation that the stretched regions of 360 videos are likely to be encoded with SKIP mode and that block homogeneity is a suitable metric for predicting its usage. To decide between an early decision or not, the algorithm evaluates the block homogeneity, dimensions, and vertical position in the frame, in addition to the QP value and video resolution. The cumulative distribution of SKIP occurrences per block variance (as in Fig. 2) considering these parameters is used to model the likelihood of selecting SKIP mode.

The Early SKIP algorithm is presented in Fig. 6. Since the video resolution, QP, and a parameter ***D*** ($D \in [0, 1]$) controlling the intensity of complexity reduction will remain fixed during the encoding, the *'Initialize'* block performs a pre-load of the thresholds corresponding to these parameters to make them easily accessible in the future.

The *'Next CU?'* conditional represents the main encoder loop: it processes one CU at a time until all CUs have been encoded (branching to 'END'). Then, *'Load threshold'* loads the threshold for current CU into variable *'th'* based on current block size and band position. In sequence, the variance of current CU is computed and stored in *'var'*. If the current CU presents a variance smaller than the threshold (conditional *'var* $<$ *th?'*), then an early decision is performed, and remaining interframes prediction stages are bypassed; else, the entire interframes prediction is performed for the current CU. Each iteration of the *'Next CU?'* loop will provide a CU of different size and position, causing *'Load threshold'* to update the current threshold 'th' to account for the encoding scenario. The variance thresholds are extracted from the cumulative distributions of SKIP mode per block variance according to several parameters (such as Fig. 2).

These distributions are generated considering all combinations of resolution, block size, band position, and QP. Then, the distributions are probed in a set of cumulative values (***D***) to generate $\{D, \textit{variance}\}$ pairs representing the likelihood of selecting SKIP mode in different encoding scenarios (i.e., a combination of resolution, block size, band position, and QP). Although *Early SKIP* can use any values of D (which employ different thresholds), a regression was performed to evaluate different combinations, and it concluded that using $D_{POLAR} = 0.35$ and $D_{MID-POL} = 0.15$ leads to the best tradeoff between time saving and coding efficiency. Note that *Early SKIP* is more aggressive in more distorted bands and it does not interfere with the encoding of the middle band.

B. Reduced SR Algorithm

Reduced SR reduces the search range width of Raster Search adaptively to the intensity of distortion. The algorithm is based on the observation that the horizontal component of motion vectors tends to be smaller in distorted regions of ERP videos.

In this work, *Reduced SR* is implemented using an approach in five bands, as presented in Fig. 1 (b). If the block is in the middle band, the search range is not modified. If the block is in the severely distorted polar bands, the search range width is reduced significantly. Finally, blocks in the mid-polar bands have their search range reduced moderately. A scaling factor $S \in [0, 1]$ determines the reduction in the default search range in each band, and a regression was performed to determine the scaling factor that results in the best performance tradeoff. After testing different combinations of scaling factors, it was concluded that using $S_{POLAR} = 0.7$ and $S_{MID-POL} = 0.3$ (i.e., reduce the search range by 70% and 30%) leads to the best tradeoff between time saving and coding efficiency. The search range in the middle band is not modified.

C. Reduced FME Algorithm

Reduced FME reduces the encoding time by adjusting the FME precision and skipping the interpolation and evaluation of horizontally-fractional samples adaptively to the intensity of projection distortion. Since ERP projection does not cause vertical stretching, only horizontal precision is adjusted. *Reduced FME* employs a division into five bands to model the variation in distortion intensity, and it assumes that the greater the distortion, the more redundant is the result of FME.

In this context, FME is modified to operate in three modes: *complete*, *moderate*, and *simplified*. *Complete* mode uses maximum horizontal precision (both ½ and ¼), and it is employed in the middle band. *Moderate* mode is used in the moderately-distorted mid-polar bands and it employs only ½ precision horizontally. Finally, *simplified* mode skips horizontal FME and it is used in the highly-distorted polar bands.

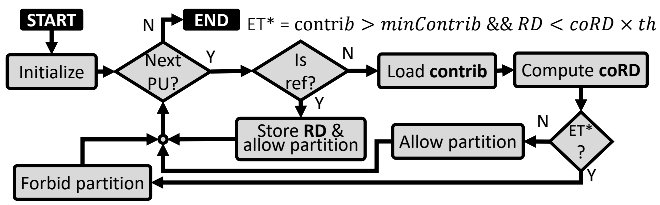


Fig. 7. Reduced Intra Sizes algorithm.

D. Reduced Intra Sizes algorithm

The *Reduced Intra Sizes* algorithm is intended to skip evaluating unlikely PU sizes during intraframe prediction based on the intensity of projection distortion. It also explores the rd-cost (which measures how fit the current block size and encoding mode are for a given region) of the blocks to assist in the decisions and make the algorithm more adaptive.

Since temporal neighboring frames are very similar, co-located regions of successive frames tend to be encoded with similar encoding modes (i.e., block sizes and prediction modes) and present similar best rd-costs. In this context, *Reduced Intra Sizes* compares the rd-cost of co-located regions and assumes that the block sizes follow a distribution such as Fig. 4 to accelerate the encoding. Since *Reduced Intra Sizes* explores the decisions of previous frames, it is disabled every once in a ***refreshRate*** to encode a frame without interferences (a reference frame) to reduce error propagation.

The *Reduced Intra Sizes* algorithm is presented in Fig. 7. In the initialization stage, the algorithm fetches the distribution of PU sizes (such as Fig. 4) for the current resolution, and the *'Next PU?'* conditional represents the main encoder loop iterating over all PUs in intraframe prediction. Once the current PU is predicted, if it is in a reference frame (*'is ref?'*), the rd-cost of the current block (*RD*) is stored, and the block is further partitioned; otherwise, further partitioning may be forbid. First, the *'contribution'* of the current block in the current row is fetched from memory (the height of one bar from Fig. 4), and the rd-cost of the co-located region (*'coRD'*) is computed. Finally, if the PU size presents a contribution larger than ***minContrib*** and its rd-cost is smaller than the co-located rd-cost assuming a tolerance ***th*** (*'ET?'* conditional), further partitioning is forbid; otherwise, it is allowed.

minContrib, ***th***, and ***refreshRate*** are parameters from the algorithm. Parameter ***minContrib*** controls the threshold between considering that a given block size occurs "frequent" or "not frequent". ***th*** controls the margin for a rd-cost to be considered similar to its co-located rd-cost. ***refreshRate*** controls how often a frame will be encoded without interference to refresh the reference rd-costs. Finally, a study was conducted to identify the best operating point, and it concluded that using $minContrib = 0.35$, $th = 1.01$, and $refreshRate = 4$ leads to the best tradeoff between time saving and coding efficiency.

V. Experimental Results and Comparison

The proposed algorithms are implemented over HM-16.16 [16] besides 360Lib-5.0 [7], following the CTCs360 [15]. The performance is assessed according to complexity reduction and coding efficiency. The complexity reduction is measured as the amount of processing time saved using the proposed algorithms when compared to the original HM. The coding efficiency is assessed using the Bjøntegaard Delta-Rate (BD-BR) metric [19], as recommended by the CTCs360 [15], and it represents the bitrate increase for a solution to achieve the same visual quality as a reference. The algorithms for interframes prediction are tested using the *RandomAccess* configuration, and the algorithm for intraframe prediction uses the *AllIntra* configuration from HM. The results for all algorithms are presented in Table II, where columns *'TS'* represent the time saving, and *'BD-BR'* represents the coding efficiency.

A. Results of interframes prediction

The results for interframes prediction show that the *Early SKIP* algorithm is able to reduce encoding time by 13.59%, on average, whereas the average coding efficiency penalty is 0.217% BD-BR. For the *Reduced SR*, the average time saving is 2.31%, whereas the average impact on coding efficiency is 0.001% BD-BR. In *Reduced FME*, the average time saving is 10.20%, with an average BD-BR of 0.421%.

When the three algorithms are enabled simultaneously in the *'InterTech'* column, the average time saving is 22.84%, with an average coding efficiency penalty of 0.625% BD-BR. These results show that the proposed algorithms for interframes prediction can accelerate the encoder significantly with minor harm to coding efficiency.

The results for two related works that accelerate the interframes prediction are also presented in Table II. The work of Liu [13] reduces encoding time by 32%, on average, whereas the average coding efficiency penalty is 1.3% BD-BR. When compared to *InterTech*, Liu [13] achieves a greater time reduction, however, the coding efficiency penalty is twice that of *InterTech*. The TS/BDBR ratio of [13] is 24.6, whereas the TS/BDBR ratio of *InterTech* is 35.0 – that is, the proposed scheme presents a better tradeoff. When the results of Ray [12] are evaluated, it is visible that Ray reduces encoding time by 15%, on average, with a coding efficiency improvement of -0.2% BD-BR. Although [12] can reduce the encoding time and improve the coding efficiency, the algorithms proposed in [12] are evaluated in the Joint Exploration Model [20]: a highly exploratory encoder built on top of HEVC to evaluate new encoding tools for future video coding standards. Considering this, the results of [12] and *InterTech* are not directly comparable. Also, it is likely that the results of [12] would not be replicable in a standardized video coding framework.

B. Results of intraframe prediction

Table II also presents the results for *Reduced Intra Sizes* and *IntraTech*. The *IntraTech* algorithm is composed of *Reduced Intra Sizes* and *Reduced Intra Modes* – the latter being a technique designed prior to the dissertation and published at [21]. The *Reduced Intra Modes* algorithm employs a division into five bands, and it skips the evaluation of some prediction modes based on an analysis of the prediction modes

TABLE II
RESULTS OF PROPOSED ALGORITHMS AND COMPARISON TO RELATED WORKS

	Interframes Prediction								**Intraframe Prediction**			
	Early SKIP		**Reduced SR**		**Reduced FME**		**InterTech**		**Reduced Intra Sizes**		**IntraTech**	
Video	TS [%]	BD-BR [%]	TS [%]	BD-BR [%]	TS [%]	BD-BR [%]	TS [%]	BD-BR [%]	TS [%]	BD-BR [%]	TS [%]	BD-BR [%]
AerialCity	9.99	0.163	0.59	-0.001	12.61	0.654	20.81	0.789	12.16%	0.369	24.95	1.328
PoleVault	10.09	0.025	0.84	0.002	11.14	0.286	19.71	0.309	11.43%	0.211	24.69	0.750
Balboa	21.44	0.305	3.02	0.034	9.57	0.524	28.39	0.847	9.75%	0.322	21.81	1.279
BranCastle	2.67	0.343	5.64	0.026	6.79	0.613	14.54	0.893	9.72%	0.164	25.18	0.718
Broadway	20.61	0.656	3.24	0.038	8.42	0.995	27.63	1.753	9.40%	0.546	21.66	1.612
Landing	12.49	0.525	4.68	-0.034	7.73	0.574	22.01	1.128	10.64%	0.434	22.66	1.991
ChairliftRide	5.43	0.087	3.78	-0.024	10.15	0.931	17.82	1.085	11.31%	0.681	24.06	1.450
GasLamp	19.61	0.002	0.28	0.001	12.29	0.043	26.78	0.045	10.18%	0.653	22.46	1.780
Harbor	21.16	0.056	0.17	0.035	12.01	0.099	28.57	0.142	9.70%	0.474	21.16	1.470
KiteFlite	7.58	-0.007	0.7	-0.016	11.23	0.274	17.96	0.301	10.72%	0.482	24.06	1.070
SkateboardInLot	13.47	0.45	4.72	-0.041	8.04	0.05	23.06	0.509	10.12%	0.037	21.26	0.581
Trolley	18.58	-0.003	0.09	-0.002	12.41	0.014	26.78	0.022	9.72%	0.083	23.20	0.532
Average	**13.59**	**0.217**	**2.31**	**0.001**	**10.20**	**0.421**	**22.84**	**0.652**	**10.41**	**0.370**	**23.09**	**1.213**
Ray [12] **							**15.00**	**-0.200**			-	-
Liu [13]							**32.00**	**1.300**			-	-
Wang [10]							-	-			**24.50**	**0.240**
Zhang [11]							-	-			**53.00**	**1.300**

** This work uses the experimental encoder *Joint Exploration Model* (JEM) [20] built on top of HEVC

distribution in different frame regions (similar to Fig. 4) and the redundancy in intermediate encoder decisions [21].

The results for *Reduced Intra Sizes* show that it reduces the encoding time by 10.41%, on average, with an average coding efficiency penalty of 0.37% BD-BR. Also, in [21] it is discussed that *Reduced Intra Modes* reduces encoding time by 15.77%, on average, with an average coding efficiency of 0.64% BD-BR. Finally, when both algorithms are enabled together (II), the average results show that it reduces encoding time by 23.09% with a coding efficiency loss of 1.213% BD-BR. The results show that *Reduced Intra Sizes* achieves significant results and that it can be combined with other algorithms to improve performance.

The results of two related works for intraframe prediction are presented in Table II. When their results are evaluated, Wang [10] reduces the encoding time by 24.5%, on average, with an average coding efficiency of 0.24% BD-BR, whereas Zhang [11] reduced encoding time by 53%, on average, with an average coding efficiency of 1.3% BD-BR. When compared to *IntraTech*, [10] achieves a similar time saving with smaller coding efficiency loss, whereas [11] achieves a greater time reduction with a slightly higher coding efficiency penalty. Although these works outperform *IntraTech*, they are mainly based on generic techniques that are applied in the same manner to all regions of the frame irrespective of projection distortion. In this context, [10] [11] are likely to obtain similar results if applied to conventional videos, whereas *IntraTech* is designed considering the projection distortion and works adaptively throughout the frame. Finally, since *IntraTech* and [10] [11] exploit different properties of the videos, it is possible to combine them and improve the overall performance.

VI. CONCLUSION

This work presented a set of evaluations that model the behavior of a video encoder when encoding ERP 360 videos and a set of algorithms to reduce encoding time. The evaluations showed that the projection distortion interferes in the selection of SKIP mode, on the magnitude of motion vectors, on the redundancy of FME, and on the selection of block sizes. Based on the evaluations, a set of algorithms were designed to accelerate the encoding of ERP 360 videos adaptively to projection distortion. *Early SKIP* performs an early decision by SKIP mode, *Reduced SR* reduces the search range of motion estimation, *Reduced FME* adjusts the precision of FME, and *Reduced Intra Sizes* avoids the evaluation of some blocks. Experimental evaluations showed that these algorithms present results competitive with related works, and they can be combined with other algorithms to improve the performance.

A. Dissertation outcome and publications

The first assessment on the encoding of 360 videos conducted before the dissertation received an honor mention at the Microelectronics Students Forum [22][2]. Then, in the one-year period in which this dissertation was developed a series of papers were produced. The evaluation of block sizes was published in the Journal of Integrated Circuits and Systems (Qualis B1) [23]. The evaluations of SKIP selection and *Early SKIP* algorithm were published at EUSIPCO (Qualis A3) [24]. A prior work was published at ICASSP (Qualis A1) [21] and incorporated into *IntraTech*. Finally, the *InterTech* scheme was accepted for publication on the IEEE Transactions on Circuits and Systems for Video Technology (Qualis A1) and will be available on the next issue [25]. The author also advised an undergraduate student in the development of a method for discarding block sizes when encoding ERP 360 videos, and such work was published at VCIP (Qualis B1) [26].

ACKNOWLEDGMENT

The authors thank FAPERGS, CNPq and CAPES (Finance Code 001) Brazilian research support agencies that financed this investigation.

[2] Available at https://ccs2.ufpel.edu.br/wp/2018/09/06/computacao-e-premiada-no-chip-in-the-pampa-2018/

REFERENCES

[1] I. C. Storch, "Exploração das distorções da projeção erp para redução de complexidade da codificação de vídeos omnidirecionais," Master's thesis, Universidade Federal de Pelotas, 2020. [Online]. Available: http://guaiaca.ufpel.edu.br:8080/handle/prefix/6086

[2] V. Cisco, "Cisco visual networking index: Forecast and methodology, 2017–2022," *CISCO White paper*, 2019.

[3] B. Gleb, "Five innovative ways you can use virtual reality in the real estate business," https://rubygarage.org/blog/virtual-reality-in-real-estate, 2020, [Online; accessed 9-July-2021].

[4] K. DeTrempe, "Hospital-wide access to virtual reality alleviates pain and anxiety for pediatric patients," https://www.stanfordchildrens.org/en/about/news/releases/2017/virtual-reality-alleviates-pain-anxiety, 2017, [Online; accessed 9-July-2021].

[5] A. Wexelblat, *Virtual reality: applications and explorations*. Academic Press, 2014.

[6] A. Torii, A. Imiya, and N. Ohnishi, "Two-and three-view geometry for spherical cameras," in *Proceedings of the sixth workshop on omnidirectional vision, camera networks and non-classical cameras*. Citeseer, 2005, pp. 81–88.

[7] Y. Ye, E. Alshima, and J. Boyce, "Jvet-e1003: Algorithm descriptions of projection format conversion and video quality metrics in 360lib," *Joint Video Exploration Team (JVET) of ITU-T SG*, vol. 16, 2017.

[8] JCT-VC, "Recommendation itu-t h. 265," *International Standard ISO/IEC*, pp. 23 008–2, 2013.

[9] J. Vanne, M. Viitanen, T. D. Hamalainen, and A. Hallapuro, "Comparative rate-distortion-complexity analysis of hevc and avc video codecs," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1885–1898, 2012.

[10] Y. Wang, Y. Li, D. Yang, and Z. Chen, "A fast intra prediction algorithm for 360-degree equirectangular panoramic video," in *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE, 2017, pp. 1–4.

[11] M. Zhang, X. Dong, Z. Liu, F. Mao, and W. Yue, "Fast intra algorithm based on texture characteristics for 360 videos," *EURASIP Journal on Image and Video Processing*, vol. 2019, no. 1, pp. 1–16, 2019.

[12] B. Ray, J. Jung, and M.-C. Larabi, "A low-complexity video encoder for equirectangular projected 360 video content," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1723–1727.

[13] Z. Liu, P. Song, and M. Zhang, "A cu split early termination algorithm based knn for 360-degree video," in *2019 Data Compression Conference (DCC)*. IEEE, 2019, pp. 594–594.

[14] F. Bossen *et al.*, "Common test conditions and software reference configurations," *JCTVC-L1100*, vol. 12, no. 7, 2013.

[15] J. Boyce, E. Alshina, A. Abbas, and Y. Ye, "Jvet-h1030: Jvet common test conditions and evaluation procedures for 360○ video," *Joint Video Explor. Team, Macau, China, Rep. JVET-H1030*, 2017.

[16] JCT-VC. (2017, Aug.) Hevc test model version 16.16. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.16/

[17] B. Zatt, M. Shafique, S. Bampi, and J. Henkel, "An adaptive early skip mode decision scheme for multiview video coding," in *28th Picture Coding Symposium*. IEEE, 2010, pp. 42–45.

[18] P. Gonçalves, G. Correa, M. Porto, B. Zatt, and L. Agostini, "Multiple early-termination scheme for tz search algorithm based on data mining and decision trees," in *2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2017, pp. 1–6.

[19] G. Bjontegaard, "Calculation of average psnr differences between rd-curves," *VCEG-M33*, 2001.

[20] J. Chen, E. Alshina, G. Sullivan, and J. R. Ohm, "Jvet-c1001: Algorithm description of joint exploration test model 3," *Joint Video Exploration Team (JVET) of ITU-T SG*, 2016.

[21] I. Storch, B. Zatt, L. Agostini, G. Correa, L. A. da Silva Cruz, and D. Palomino, "Spatially adaptive intra mode pre-selection for erp 360 video coding," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 2178–2182.

[22] I. Storch, D. Palomino, L. Agostini, B. Zatt, and L. Cruz, "An assessment of hevc intra-frame prediction over 360-degrees videos," in *18th Microelectronics Students Forum*. SBMicro, 2018, pp. 1–4.

[23] I. Storch, L. A. da Silva Cruz, L. Agostini, B. Zatt, and D. Palomino, "The impacts of equirectangular 360-degrees videos in the intra-frame prediction of hevc," *Journal of Integrated Circuits and Systems*, vol. 14, no. 1, pp. 1–10, 2019.

[24] I. Storch, G. Correa, B. Zatt, L. Agostini, and D. Palomino, "Esa360-early skip mode decision algorithm for fast erp 360 video coding," in *2020 28th European Signal Processing Conference (EUSIPCO)*. IEEE, 2021, pp. 535–539.

[25] I. Storch, L. Agostini, B. Zatt, S. Bampi, and D. Palomino, "Fastinter360: A fast inter mode decision for hevc 360 video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 8, p. to appear, 2021.

[26] B. Beling, I. Storch, L. Agostini, B. Zatt, S. Bampi, and D. Palomino, "Erp-based ctu splitting early termination for intra prediction of 360 videos," in *2020 IEEE International Conference on Visual Communications and Image Processing (VCIP)*. IEEE, 2020, pp. 359–362.