Accepted in IEEE Canadian Conference on Electrical and Computer Engineering (CCECE 2019), 2019

Comparison of 3D 360-Degree Video Compression Performance Using Different Projections

Mohammadreza Jamali
Department of Software Engineering and IT
École de technologie supérieure
Montréal, Canada
Mohammadreza.Jamali.1@ens.etsmtl.ca

Firouzeh Golaghazadeh
Department of Software Engineering and IT
École de technologie supérieure
Montréal, Canada
Firouzeh.Golaghazadeh.1@ens.etsmtl.ca

Stéphane Coulombe
Department of Software Engineering and IT
École de technologie supérieure
Montréal, Canada
Stephane.Coulombe@etsmtl.ca

Ahmad Vakili
R&D Department
Summit Tech Multimedia
Montréal, Canada
Vakili@summit-tech.ca

Carlos Vazquez
Department of Software Engineering and IT
École de technologie supérieure
Montréal, Canada
Carlos.Vazquez@etsmtl.ca

***Abstract*—3D and multi-view 360-degree videos offer 3D immersive visual experience of the real world on head-mounted displays (HMDs). They are used in extended reality applications (including virtual reality, augmented reality and mixed reality) such as gaming, entertainment, education and medicine. However, these applications require a very high rate of compression and a great deal of computational resources. Compression efficiency and coding complexity are highly influenced by 360 projections used to map spherical frames to rectangular frames. To study the effect of this mapping process, in this paper, we evaluate the performance of various projections in 3D and multi-view 360-degree videos in terms of coding efficiency, quality and complexity. The evaluation is conducted for two coding scenarios: stereo coding scenario where the encoder considers the inter-view dependency for disparity estimation and simulcast coding scenario where two views are encoded separately. This performance evaluation approach provides valuable insights on using 360-degree projections in 3D and multi-view coding.**

***Index Terms*—360-degree video, projection, 3D video, multi-view video, performance evaluation**

I. Introduction

Omnidirectional (360-degree) video applications are diverse and increasingly popular. They range from immersive movies and live-streamed concerts to shopping and medicine. Moreover, 360-degree is the main video format used in virtual reality which provides numerous possibilities for users to utilize applications such as video games [1].

Processing a 360-degree video is a challenging issue since it is encoded as a spherical video while the user is looking at only one viewport and does not consider the rest of the video. In addition, current video coding standards consider the video as a two-dimensional rectangle while in 360-degree video, pixels are located on a sphere which creates new challenges. Thus, to encode a 360-degree video by current video coders, the spherical frames need to be mapped (projected) onto two-dimensional rectangular frames. As a result of this mapping process, the video content experiences a distortion, which makes projection selection a critical issue in 360-degree video. After projection, the frames become high-resolution frames which require high computational complexity for processing and huge data rates for transmission. These problems are aggravated in 3D and multi-view video.

In the literature, projection performance evaluation is conducted to demonstrate the impact of sphere-to-rectangle mapping on video distortions and discontinuities which lead to quality degradation. For example, [2] has evaluated the coding efficiency on 360-degree video and reported the results for three video codecs: AV1, HEVC reference software HM [3], and JVET JEM [4]. Furthermore, the impact of different projections on the coding efficiency of monoscopic video is studied in [5] by considering the quality of different viewports. Moreover, [6] provides a review of the latest advances in the area of omnidirectional video. This work studies the coding performance under different projection methods.

To extend previous studies, in this paper, we evaluate the performance of different projections in the context of multi-view 360-degree video. In doing so, a two-view setting is considered where the encoder chooses between simulcast coding and stereo coding. In simulcast coding, the dependency between two views is not used for disparity estimation and thus the views are encoded separately, while in stereo coding the views are jointly encoded. To carry out this performance evaluation, we develop software tools that allow us to evaluate the performance of various multi-view 360-degree components (e.g., projections and encoders) under various quality metrics. The developed software is an extension of 360Lib [7] from MPEG's joint video exploration team (JVET) and supports a wider range of encoders including multi-view and 3D encoders which are currently lacking in 360Lib.

The remainder of the paper is organized as follows. Section II provides a brief overview of the different projection types and quality metrics. Section III discusses our method-

This work was funded by Mitacs Canada, Natural Sciences and Engineering Research Council of Canada (NSERC) and Summit Tech Multimedia.

Paper presented at 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), Edmonton, AB, Canada, 5-8 May 2019
https://doi.org/10.1109/CCECE.2019.8861868

ology for performance evaluation and section IV presents the experimental results. Finally, section V concludes the paper.

II. Projections and Quality Metrics

In this section, we explain the different projections which are evaluated in this study. Moreover, a summary of quality metrics which are specifically used for omnidirectional video is provided.

A. Projections

Modern video coding standards (such as high efficiency video coding (HEVC) [8], [9]) are designed for two-dimensional rectangular images. Therefore, a 360-degree video, which is displayed within a spherical domain, is projected onto a two-dimensional plane to be compatible with the currently available video coding standards. This can be performed by means of a projection format. To this end, many projection formats have been proposed and developed in the literature [10]. In the following, some of the projection formats included in the 360Lib package [7], which are used in this work, are discussed:

- Equirectangular projection (ERP) is a widespread and popular projection format used for omnidirectional video. Using ERP, a spherical image is mapped onto a rectangular image as it is done in the two-dimensional world map. ERP suffers from oversampling issues. The north and south poles of the sphere are severely stretched and oversampled compared to the center areas. The bit wasting caused by stretching at the poles can reduce the coding efficiency [11].
- In cube map projection (CMP), the sphere is assumed to be enclosed in a cube and each part of the spherical image is projected onto one of the six faces of the surrounding cube. Then, the six faces are unfolded to yield a two-dimensional representation of the sphere. There are two types of CMP, including non-compact and compact formats which differ based on the faces' positions. These two frame-packing formats are called non-compact CMP4x3 and compact CMP3x2, where the two numbers specify the number of columns and rows respectively. There are some extra gray areas in CMP4x3 (non-compact) format.
- In octahedron projection (OHP), the spherical image is projected onto eight triangular faces of an octahedron. Similar to CMP, there are two ways of compact and non-compact frame-packing [12]. Also, the discontinuities between neighboring faces can affect the coding efficiency.
- In segmented sphere projection (SSP), the sphere is divided into three tiles as north pole, south pole and equator. The north and south poles are projected onto two circles and the equator area is mapped onto a rectangle similar to the mapping in ERP. Compared to ERP, SSP has less extra pixels at the pole areas which can improve the coding efficiency, but its "Null" gray areas around the circles can have an opposite effect on coding efficiency.

B. Quality Metrics

Since traditional peak signal-to-noise ratio (PSNR) is not a good choice as an objective quality metric in the 360-degree video context [13], four new spherical quality metrics are implemented in 360Lib for 360-degree video quality evaluation: weighted spherical PSNR (WS-PSNR), spherical PSNR without interpolation (S-PSNR-NN), spherical PSNR with interpolation (S-PSNR-I) and craster parabolic projection PSNR (CPP-PSNR) which are explained below:

- In WS-PSNR, the distortion at each sample position is weighted by the area on the sphere covered by the given sample position. All samples on the two-dimensional projection plane are used in WS-PSNR calculation. The two inputs to the metric calculation must have the same resolution and projection format.
- In S-PSNR-NN, PSNR is calculated based on a set of points uniformly sampled on the sphere. To find the sample value at the corresponding position on the projection plane, nearest neighbor rounding is applied. The two inputs to the metric calculation may have different resolutions and/or projection formats.
- In S-PSNR-I, PSNR is calculated based on a set of points uniformly sampled on the sphere. To find the sample value at the corresponding position on the projection plane, bicubic interpolation is applied. The two inputs to the metric calculation may have different resolutions and/or projection formats.
- CPP-PSNR applies another projection format conversion to convert the two inputs into crasters parabolic projection (CPP). Then PSNR is calculated in the CPP domain. The two inputs to the metric calculation may have different resolutions and/or projection formats.

In this paper, we report the results based on all these quality metrics where values for three components (Y, U and V) are combined.

III. Evaluation Methodology

In this section, we discuss our methodology for performance evaluation based on 360Lib software package and MV-HEVC [14] as a multi-view encoding system. To this end, we use HTM [15] which is a 3D and multi-view video encoder proposed by Fraunhofer Heinrich Hertz Institute. The goal is to explore the compression-quality-complexity trade-off while different projections are applied to the video prior to encoding. For each projection, two scenarios of simulcast coding (coding a multi-view 360-degree video by two separate views) and stereo coding (coding the video as a stereo sequence to take advantage of inter-view dependency) are considered. The results of this evaluation are reported based on Bjøntegaard delta rate (BD-Rate) [16] as a measure of compression efficiency and encoding time as a measure of complexity.

A. Video Sequences

In this paper, we use 2K two-view 360-degree sequences which are all in raw YUV 4:2:0 format. They consist of two left and right views without any depth map. The sequences are provided by Summit Tech Multimedia [17] and Suometry [18] and are listed in Table I. They exhibit a wide range of characteristics and motions from a low-motion video in *Kitchen10* to a high-motion video in *Dance*. We believe these sequences let us provide a genuine comparison between different projections in the two cases of stereo and simulcast coding. These sequences (other than *Dance*) are available at *etsmtl.ca/vplab/360video*.

The original sequences are 4Kx2K videos in a top-bottom format. However, 3D and multi-view video encoders such as HTM, require two separate files, one per view, as inputs to the encoder. Thus, as a pre-processing step, two views are separated as two input files for left and right views. As a result, both views are 4Kx1K which are downsampled horizontally to have the standard 2:1 ratio and the size of 2Kx1K. It should be mentioned that in sequences *OutdoorA* and *OutdoorL*, 1/3 of the bottom of the frames is black due to capturing system configuration. For sequence *Kitchen10*, 1/3 of the top of the frames is black. This explains why these sequences have lower bit rates compared to the other two sequences as it will be reported in Tables II and IV.

TABLE I: 3D 360-degree test sequences (resolution and FPS are per view)

Sequence name	Motion	Resolution	FPS	Bit-depth
Aikidomirror1	Medium	1920x960	30	8
Dance	High	1920x960	30	8
OutdoorA	Medium	1920x960	30	8
Kitchen10	Low	1920x960	30	8
OutdoorL	Medium	1920x960	30	8

B. Evaluation Framework

To conduct the experiments, we use a framework recommended by JVET [19] which is shown in Fig. 1. In total, there are four steps of projection conversion, encoding, decoding and projection reverse conversion. The input to the projection conversion step is a sequence in ERP format which is converted to another projection prior to encoding. Moreover, during the first step of conversion, sequences go through a downsampling which keeps 75% of the original video pixels for encoding [19]. In the last step of reverse conversion, an upsampling is carried out to bring back the resolution to the original size. The output of the system is a sequence in the original ERP format.

While this framework has been used for one-view video, we extend it to two-view stereo and two-view simulcast coding. Both right and left views (which are in ERP format) are converted to a new projection and provided to the HTM encoder as the input. After decoding, the reverse conversion is applied to both views and as the output, we have two reconstructed views in ERP format.

C. Encoder Settings

The encoder is configured in low-delay P configuration which means there is no B frame and the GOP size is set to 1. As a result, each frame uses only one previous frame to do the motion estimation. In the stereo case and for the second view, each frame uses one previous frame from the current view and one frame from the neighboring view to perform the prediction. It should be noted that the distance between two intra frames (IDR frames) is set to 32 in all experiments.

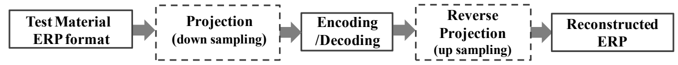

Fig. 1: 360-degree video evaluation framework recommended by JVET.

TABLE II: Bitrate and quality comparison of different projections in stereo coding (results are averaged over four QPs: 22, 27, 32 and 37)

Video sequence	Projection	Bitrate (kbps)	PSNR (dB)	S-PSNR-NN (dB)	WS-PSNR (dB)	S-PSNR-I (dB)	CPP-PSNR (dB)
Aikidomirror1	ERP	1623	39.7	39.5	39.5	39.6	39.6
	CMP3x2	1767	36.1	39.3	39.3	39.4	39.4
	CMP4x3	1785	36.1	39.3	39.3	39.4	39.4
	SSP	1858	36.8	39.6	39.6	39.7	39.7
	OHP	1912	37.0	39.4	39.4	39.5	39.6
Dance	ERP	3449	37.3	36.5	36.5	36.8	36.8
	CMP3x2	3909	37.0	36.7	36.7	37.0	37.0
	CMP4x3	3903	37.0	36.7	36.7	37.0	37.0
	SSP	4103	37.5	37.1	37.1	37.4	37.4
	OHP	4219	37.4	36.9	36.9	37.2	37.2
OutdoorA	ERP	1086	40.2	39.9	39.9	40.0	40.0
	CMP3x2	1052	38.7	39.9	39.9	40.0	40.0
	CMP4x3	1070	38.7	39.9	39.8	40.0	40.0
	SSP	1160	39.3	40.3	40.2	40.4	40.4
	OHP	1433	39.6	40.2	40.2	40.4	40.4
Kitchen10	ERP	337	43.6	42.8	42.8	42.9	42.9
	CMP3x2	379	43.1	43.0	43.0	43.1	43.1
	CMP4x3	392	43.1	43.0	43.0	43.1	43.1
	SSP	427	43.7	43.3	43.3	43.4	43.4
	OHP	564	43.7	43.1	43.2	43.3	43.3
OutdoorL	ERP	656	42.7	41.3	41.3	41.6	41.6
	CMP3x2	743	42.4	41.7	41.6	41.9	41.9
	CMP4x3	759	42.4	41.7	41.6	41.9	41.9
	SSP	802	43.1	42.1	42.1	42.4	42.3
	OHP	1044	43.1	42.0	42.0	42.3	42.2
Average	ERP	**1430**	**40.7**	**40.0**	**40.0**	**40.2**	**40.2**
	CMP3x2	**1570**	**39.5**	**40.1**	**40.1**	**40.3**	**40.3**
	CMP4x3	**1582**	**39.5**	**40.1**	**40.1**	**40.3**	**40.3**
	SSP	**1670**	**40.1**	**40.5**	**40.5**	**40.7**	**40.6**
	OHP	**1834**	**40.2**	**40.3**	**40.3**	**40.5**	**40.5**

IV. Experimental Results

The experiments are carried out for two scenarios of stereo coding and simulcast coding of multi-view 360-degree video. Different projections are considered in the encoding step of both scenarios. In the stereo case, right and left views are encoded using the inter-view dependency and in the simulcast coding, right and left views are encoded separately. It should be mentioned that for both scenarios, we use MV-HEVC test model HTM-16.2 which can be configured to avoid inter-view prediction in simulcast coding. The experiments are run by a

PC equipped with an Intel® Core™ i7-4790 CPU @ 3.60 GHz and 32 GB of RAM. The configuration is multi-view (without depth map) with two views. We report the results, using 100 frames of two-view 360-degree sequences which are listed in Table I. The results are based on frames per hour (FPH) and BD-Rate where multi-view stereo with ERP format is used as the anchor. FPH is calculated as follows:

$$\mathrm{FPH} = \frac{3600 \times 100}{\mathit{Total\ time}}. \tag{1}$$

The results are averaged over four quantization parameters (QPs): 22, 27, 32 and 37. They, therefore, cover low and high bitrate scenarios. The time is reported for four steps of projection conversion, encoding, decoding and projection reverse conversion to demonstrate which part is affected the most by using different projections. The final FPH is measured using the total time, measured in seconds, of four steps. To calculate BD-Rate, we apply Bjøntegaard algorithm by using the following combined PSNR [20] for each quality metric:

$$\mathrm{PSNR} = \frac{6 \cdot \mathrm{PSNR_Y} + \mathrm{PSNR_U} + \mathrm{PSNR_V}}{8}. \tag{2}$$

This assures us that we measure the impact of the proposed algorithms on the total quality.

Table II presents the results for different projections based on PSNR, S-PSNR-NN, WS-PSNR, S-PSNR-I and CPP-PSNR along with the averages over all sequences. The results in this table are reported for the stereo scenario. Considering the average section in this table, we observe that ERP provides the lowest bitrate and the best quality when we consider PSNR as the quality metric. For other objective quality metrics (which are spherical quality metrics and more appropriate for 360-degree video) and to see which projection is superior (i.e., provides the best trade-off between quality and bitrate), we need to provide BD-Rate as shown in Fig. 2. Moreover, Table III shows time consumption at each step for different projections in the stereo case. Based on this table, we observe that ERP and CMP3x2 are the fastest projections considering the total time and they provide the lowest time consumption at all steps. It should be noted that time results reported in this table and in other similar tables are measured on a specific machine using non-optimized codes (360Lib and HTM). They are reported here for the sake of comparison and without any liability. The comparison should be considered vertically between different projections and not between conversion, encoding, decoding and reverse conversion for a given projection. For instance, we see that CMP4x3 and OHP require significantly more time to encode and decode compared to other projections. It should also be noted that the reverse projection is performed on the whole frame while in some applications, only the region of interest might need to be rendered.

In multi-view simulcast coding, the encoder does not consider the inter-view dependency and encodes the two views separately. Similar to the stereo case, we consider different video sequences and different projections to obtain the results based on bitrate, quality and FPH. Table IV shows the results for each sequence along with the average over all sequences. Considering the average section in this table, we observe that ERP provides the lowest bitrate and the best quality when we consider PSNR as the quality metric. These results are similar to the results we observed for the stereo case. For other objective quality metrics (spherical quality metrics) and to see which projection is superior (i.e, provides the best trade-off between quality and bitrate), we need to provide BD-Rate as demonstrated in Fig. 2. In addition, Table V shows the time consumption at each step for different projections in the simulcast case. Based on this table and similar to the stereo case, we can conclude that ERP and CMP3x2 are the fastest projections considering the total time. Table VI presents the FPH versus BD-Rate for all sequences. The averages in this table show the overall trade-off between quality and complexity for different projections in the two scenarios of stereo and simulcast coding.

To visually show the results, Fig. 2 depicts the FPH versus BD-Rate averaged over all sequences and QPs. This figure together with Table VI indicate that in simulcast coding, the

TABLE III: Time consumption comparison of different projections in stereo coding, numbers are in seconds (results are averaged over four QPs: 22, 27, 32 and 37)

Video sequence	**Projection**	**Conversion time**	**Encoding time**	**Decoding time**	**Reverse conversion time**	**Total time**
Aikidomirror1	ERP	31	1545	7	337	1919
	CMP3x2	34	1556	7	341	1939
	CMP4x3	39	2706	12	339	3096
	SSP	37	1648	7	350	2042
	OHP	50	2738	12	387	3188
Dance	ERP	27	1664	7	317	2014
	CMP3x2	31	1754	7	326	2119
	CMP4x3	38	2972	22	343	3375
	SSP	35	1858	8	332	2234
	OHP	46	2951	13	368	3378
OutdoorA	ERP	29	1311	6	326	1672
	CMP3x2	33	1301	6	336	1675
	CMP4x3	35	2472	11	336	2853
	SSP	35	1387	6	334	1762
	OHP	48	2506	11	375	2939
Kitchen10	ERP	30	1247	5	330	1612
	CMP3x2	32	1240	5	334	1612
	CMP4x3	34	2371	10	329	2744
	SSP	35	1338	6	340	1719
	OHP	46	2384	11	366	2806
OutdoorL	ERP	28	1248	5	327	1608
	CMP3x2	33	1281	6	340	1660
	CMP4x3	36	2400	11	336	2782
	SSP	39	1384	6	350	1780
	OHP	47	2459	11	375	2892
Average	ERP	**29**	**1403**	**6**	**327**	**1765**
	CMP3x2	**33**	**1426**	**6**	**335**	**1801**
	CMP4x3	**36**	**2584**	**13**	**337**	**2970**
	SSP	**36**	**1523**	**7**	**341**	**1908**
	OHP	**47**	**2608**	**12**	**374**	**3041**

TABLE IV: Bitrate and quality comparison of different projections in simulcast coding (results are averaged over four QPs: 22, 27, 32 and 37)

Video sequence	**Projection**	**Bitrate (kbps)**	**PSNR (dB)**	**S-PSNR-NN (dB)**	**WS-PSNR (dB)**	**S-PSNR-I (dB)**	**CPP-PSNR (dB)**
Aikidomirror1	ERP	1761	39.9	39.7	39.7	39.8	39.9
	CMP3x2	1902	36.2	39.5	39.5	39.6	39.7
	CMP4x3	1920	36.2	39.5	39.5	39.6	39.6
	SSP	2013	36.9	39.8	39.8	40.0	40.0
	OHP	2133	37.2	39.6	39.6	39.8	39.8
Dance	ERP	4592	37.5	36.7	36.7	37.0	37.0
	CMP3x2	5093	37.2	36.9	36.9	37.1	37.1
	CMP4x3	5079	37.2	36.9	36.9	37.2	37.2
	SSP	5333	37.7	37.3	37.3	37.6	37.6
	OHP	5601	37.5	37.1	37.1	37.3	37.4
OutdoorA	ERP	1224	40.5	40.2	40.1	40.3	40.3
	CMP3x2	1188	38.9	40.1	40.1	40.3	40.3
	CMP4x3	1210	38.9	40.1	40.1	40.3	40.3
	SSP	1326	39.5	40.5	40.5	40.7	40.7
	OHP	1681	39.8	40.5	40.5	40.7	40.6
Kitchen10	ERP	413	44	43.1	43.1	43.3	43.3
	CMP3x2	457	43.4	43.3	43.4	43.5	43.5
	CMP4x3	475	43.4	43.3	43.4	43.5	43.5
	SSP	533	44.0	43.6	43.6	43.7	43.8
	OHP	751	44.0	43.5	43.5	43.6	43.6
OutdoorL	ERP	811	43.1	41.8	41.8	42	42
	CMP3x2	905	42.8	42.1	42.1	42.4	42.3
	CMP4x3	924	42.9	42.1	42.1	42.4	42.3
	SSP	991	43.6	42.6	42.6	42.9	42.8
	OHP	1309	43.5	42.5	42.5	42.8	42.7
Average	ERP	**1760**	**41**	**40.3**	**40.3**	**40.5**	**40.5**
	CMP3x2	**1909**	**39.7**	**40.4**	**40.4**	**40.6**	**40.6**
	CMP4x3	**1922**	**39.7**	**40.4**	**40.4**	**40.6**	**40.6**
	SSP	**2039**	**40.3**	**40.8**	**40.8**	**41.0**	**41.0**
	OHP	**2295**	**40.4**	**40.6**	**40.6**	**40.8**	**40.8**

TABLE V: Time consumption comparison of different projections in simulcast coding, numbers are in seconds (results are averaged over four QPs: 22, 27, 32 and 37)

Video sequence	**Projection**	**Conversion time**	**Encoding time**	**Decoding time**	**Reverse conversion time**	**Total time**
Aikidomirror1	ERP	28	1259	6	331	1625
	CMP3x2	31	1252	7	339	1628
	CMP4x3	37	2280	12	343	2672
	SSP	36	1348	7	345	1736
	OHP	46	2242	12	370	2670
Dance	ERP	27	1534	8	314	1882
	CMP3x2	31	1628	9	330	1997
	CMP4x3	34	2668	14	333	3049
	SSP	36	1729	12	335	2112
	OHP	47	2640	14	361	3062
OutdoorA	ERP	28	1097	6	327	1458
	CMP3x2	32	1090	6	335	1464
	CMP4x3	34	2071	11	331	2447
	SSP	35	1178	7	339	1559
	OHP	46	2126	11	367	2550
Kitchen10	ERP	27	1018	5	320	1371
	CMP3x2	33	1050	6	339	1427
	CMP4x3	36	2049	11	340	2435
	SSP	36	1129	6	342	1514
	OHP	48	2044	11	373	2475
OutdoorL	ERP	29	1073	6	332	1440
	CMP3x2	34	1114	6	352	1506
	CMP4x3	36	2049	11	343	2440
	SSP	36	1164	6	347	1553
	OHP	48	2102	11	376	2538
Average	ERP	**28**	**1196**	**6**	**325**	**1555**
	CMP3x2	**32**	**1227**	**7**	**339**	**1604**
	CMP4x3	**35**	**2223**	**12**	**338**	**2609**
	SSP	**36**	**1310**	**8**	**342**	**1695**
	OHP	**47**	**2231**	**12**	**369**	**2659**

total process is faster due to the fact that inter-view prediction is avoided. In addition, BD-Rate is higher because the motion estimation is restricted to the same view and cannot take advantage of the correlation existing between two views. We can observe that stereo coding has 15% to 28% BD-Rate improvement over its simulcast counterpart which shows that stereo coding improves coding efficiency even for 360-degree video. Among different projections, ERP_Stereo and SSP_Stereo provide the lowest BD-Rate (lowest bitrate at the same quality) and ERP_Simulcast and CMP3x2_Simulcast are the fastest projections compared to others.

V. Conclusion

The coding of 360-degree video, more specifically 3D and multi-view 360-degree video, is quite challenging. Such applications require high resolution frames with high quality outputs demanding significant resources to perform projection and encoding-decoding, especially imperative for real-time applications. To address these requirements, in this study, we conducted a performance evaluation to show the trade-off between different spherical projections used in the encoding-decoding process in two cases of stereo and simulcast coding. In multi-view 360-degree video, stereo coding requires lower bitrate compared to simulcast coding because it uses one of the views for disparity estimation; however, it is slower due to the extra time spent for inter-view prediction. In

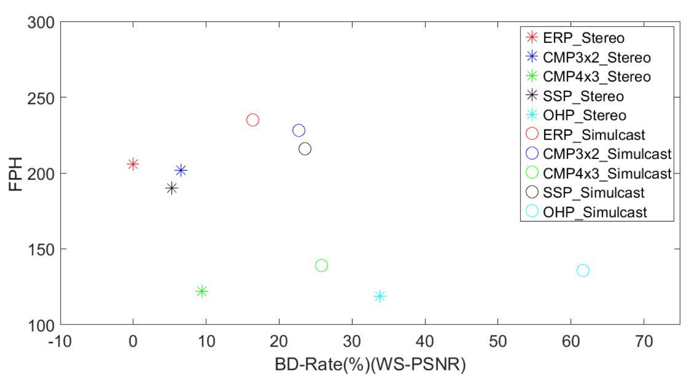

Fig. 2: Frames per hour (FPH) versus BD-Rate, averages over all sequences (ERP_Stereo is considered as anchor for BD-Rate and FPH is averaged over four QPs: 22, 27, 32 and 37).

TABLE VI: BD-Rate (WS-PSNR) versus frames per hour (FPH) of different projections (ERP_Stereo is considered as anchor for BD-Rate and FPH is averaged over four QPs: 22, 27, 32 and 37)

Video sequence	Projection	BD-Rate (%)	FPH	Projection	BD-Rate (%)	FPH
Aikidomirror1	ERP_Stereo	0	188	ERP_Simulcast	3.7	222
	CMP3x2_Stereo	17.6	186	CMP3x2_Simulcast	21.2	221
	CMP4x3_Stereo	19.5	116	CMP4x3_Simulcast	23.3	135
	SSP_Stereo	12.6	176	SSP_Simulcast	16.5	207
	OHP_Stereo	25.6	113	OHP_Simulcast	34.2	135
Dance	ERP_Stereo	0	179	ERP_Simulcast	37.8	191
	CMP3x2_Stereo	7.6	170	CMP3x2_Simulcast	46.3	180
	CMP4x3_Stereo	7.3	107	CMP4x3_Simulcast	45.6	118
	SSP_Stereo	0.7	161	SSP_Simulcast	37.3	170
	OHP_Stereo	11.8	107	OHP_Simulcast	54.7	118
OutdoorA	ERP_Stereo	0	215	ERP_Simulcast	8.2	247
	CMP3x2_Stereo	-2.3	215	CMP3x2_Simulcast	5.8	246
	CMP4x3_Stereo	0.3	126	CMP4x3_Simulcast	8.7	147
	SSP_Stereo	-3.2	204	SSP_Simulcast	6.8	231
	OHP_Stereo	25.6	122	OHP_Simulcast	41.7	141
Kitchen10	ERP_Stereo	0	223	ERP_Simulcast	14.6	263
	CMP3x2_Stereo	4.3	223	CMP3x2_Simulcast	17.9	252
	CMP4x3_Stereo	11.0	131	CMP4x3_Simulcast	25.4	148
	SSP_Stereo	12.4	209	SSP_Simulcast	33.0	238
	OHP_Stereo	63.7	128	OHP_Simulcast	108.5	145
OutdoorL	ERP_Stereo	0	224	ERP_Simulcast	17.7	250
	CMP3x2_Stereo	5.4	217	CMP3x2_Simulcast	22.3	239
	CMP4x3_Stereo	8.7	129	CMP4x3_Simulcast	26.0	148
	SSP_Stereo	4.1	202	SSP_Simulcast	23.7	232
	OHP_Stereo	42.1	124	OHP_Simulcast	69.2	142
Average	ERP_Stereo	**0**	**206**	ERP_Simulcast	**16.4**	**235**
	CMP3x2_Stereo	**6.5**	**202**	CMP3x2_Simulcast	**22.7**	**228**
	CMP4x3_Stereo	**9.4**	**122**	CMP4x3_Simulcast	**25.8**	**139**
	SSP_Stereo	**5.3**	**190**	SSP_Simulcast	**23.5**	**216**
	OHP_Stereo	**33.8**	**119**	OHP_Simulcast	**61.7**	**136**

addition, we observed that ERP_Stereo and SSP_Stereo let the encoder achieve highest compression while ERP_Simulcast and CMP3x2_Simulcast provide the fastest overall process of projection, encoding, decoding and reverse projection.

ACKNOWLEDGEMENT

The authors would like to thank Summit Tech Multimedia and Suometry for providing the video sequences.

REFERENCES

[1] L. Argyriou, D. Economou, V. Bouki, and I. Doumanis, "Engaging immersive video consumers: Challenges regarding 360-degree gamified video applications," in *15th International Conference on Ubiquitous Computing and Communications*, pp. 145–152, Dec 2016.

[2] P. Topiwala, W. Dai, M. Krishnan, A. Abbas, S. Doshi, and D. Newman, "Performance comparison of AV1, HEVC, and JVET video codecs on 360 (spherical) video," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, 2017.

[3] K. Il-Koo, "High efficiency video coding (HEVC) test model 15 (HM15) encoder description," *Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, April 2014.

[4] JEM-5.0.1, https://jvet.hhi.fraunhofer.de/svn/svn_HMJEMSoftware/tags/HM-16.6-JEM-5.0.1/, [Accessed: 1-Feb-2019].

[5] M. Yu, H. Lakshman, and B. Girod, "A framework to evaluate omnidirectional video coding schemes," in *2015 IEEE International Symposium on Mixed and Augmented Reality*, pp. 31–36, Sep. 2015.

[6] Z. Chen, Y. Li, and Y. Zhang, "Recent advances in omnidirectional video coding for virtual reality: Projection and evaluation," *Signal Processing*, vol. 146, pp. 66 – 78, 2018.

[7] 360Lib-2.1, https://jvet.hhi.fraunhofer.de/svn/svn_360Lib/tags/360Lib-2.1/, [Accessed: 1-Feb-2019].

[8] ISO/IEC JTC 1/SC 29/WG 11, "High Efficiency Video Coding," 2013.

[9] G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, pp. 1649–1668, Dec 2012.

[10] A. Abbas and D. Newman, "A novel projection for omni-directional video," in *Applications of Digital Image Processing XL*, vol. 10396, p. 103960V, International Society for Optics and Photonics, 2017.

[11] C. Zhang, Y. Lu, J. Li, and Z. Wen, "AHG8: Segmented sphere projection for 360-degree video," in *Joint Video Exploration Team of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JVET-E0025*, 2017.

[12] H. Lin, C. Li, J. Lin, S. Chang, and C. Ju, "AHG8: An efficient compact layout for octahedron format," *Joint Video Exploration Team of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JVET-D0142*, 2016.

[13] F. Lopes, J. Ascenso, A. Rodrigues, and M. P. Queluz, "Subjective and objective quality assessment of omnidirectional video," *Proc. SPIE*, vol. 10752, pp. 10752 – 10752 – 17, 2018.

[14] G. Tech, Y. Chen, K. Müller, J. Ohm, A. Vetro, and Y. Wang, "Overview of the multiview and 3D extensions of high efficiency video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, pp. 35–49, Jan 2016.

[15] HTM-16.2, https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/, [Accessed: 1-Feb-2019].

[16] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," *VCEG of ITU-T, VCEG-M33*, April 2001.

[17] Summit Tech Multimedia, https://www.summit-tech.ca/, [Accessed: 1-Feb-2019].

[18] Suometry, https://www.suometry.com/, [Accessed: 1-Feb-2019].

[19] J. Boyce, E. Alshina, A. Abbas, and Y. Ye, "JVET common test conditions and evaluation procedures for 360 video," *Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-E1030*, 2017.

[20] V. Sze, M. Budagavi, and G. J. Sullivan, "High efficiency video coding (HEVC)," *Integrated Circuit and Systems, Algorithms and Architectures, Springer*, pp. 1–375, 2014.