Intra-coding of 360-degree images on the sphere

Navid MAHMOUDIAN BIDGOLI, Thomas MAUGEY, Aline ROUMY
Inria, Univ Rennes, CNRS, IRISA, France

***Abstract*—Omni-directional images are characterized by their high resolution (usually 8K) and therefore require high compression efficiency. Existing methods project the spherical content onto one or multiple planes and process the mapped content with classical 2D video coding algorithms. However, this projection induces sub-optimality. Indeed, after projection, the statistical properties of the pixels are modified, the connectivity between neighboring pixels on the sphere might be lost, and finally, the sampling is not uniform. Therefore, we propose to process uniformly distributed pixels directly on the sphere to achieve high compression efficiency. In particular, a scanning order and a prediction scheme are proposed to exploit, directly on the sphere, the statistical dependencies between the pixels. A Graph Fourier Transform is also applied to exploit local dependencies while taking into account the 3D geometry. Experimental results demonstrate that the proposed method provides up to 5.6% bitrate reduction and on average around 2% bitrate reduction over state-of-the-art methods.**

***Index Terms*—omnidirectional image, compression, prediction, graph transform**

I. Introduction

Omnidirectional or 360° videos are visual contents defined on the 2D sphere. To compress these spherical contents, a classical approach consists in first projecting the content onto one or multiple 2D planes, and then sampling uniformly the projected content. This allows to benefit from the efficiency of conventional block-based 2D plane video coders. Examples of such projections are equirectangular [1], cube map [2], [3], rhombic dodecahedron [4], and dyadic [5]. Then, to achieve good compression performance, a 2D projection must satisfy the three following properties. First, *(i)* the projection should lead to the same distortion and the same image statistics as the ones induced by perspective projection, such that the projected images have the characteristics which the conventional video coders have been optimized for. Second, *(ii)* the sampling should be uniform on the sphere, to achieve fair rate-allocation. Finally, *(iii)* the connectivity between neighboring samples on the sphere should be preserved after projection, to better exploit spatial dependencies.

However, all the 2D projections listed above [1]–[5] fail to meet simultaneously these three requirements (see Table I for more details). For these reasons, there is the need to redesign conventional video coders to adapt to the characteristics of spherical contents.

One approach to design compression algorithms for spherical contents is to extend the 2D plane compression algorithm and take into account the 3D geometry. For instance, new motion compensation predictions are proposed in [6]–[8]. Another example is the transform proposed in [9], which is defined for equirectangular projected content. This graph transform compensates for the non-uniform sampling by weighting the graph with the geodesic distance between samples. Still, all these methods rely on a 2D projection. Therefore, these methods compensate for some of the drawbacks of the projection but do not handle all of them. In [10], by proposing a dictionary of atoms living on the 2D sphere, the geometry of the data is taken into account, but it focuses mostly on the transform coding of the images and no prediction is used to further remove the spatial correlation within the image.

In this paper, we propose to compress the spherical content and perform all processings directly on the sphere. The goal is to avoid projecting the data onto 2D planes. Requirements (ii) and (iii) are hence fulfilled, and the property (i) is no more required. Moreover, the focus of the paper is on intra coding, where images are coded independently of the others to allow resynchronization. Intra coding is a key element in

TABLE I
Characteristics of different 2D projections of 360° contents

projection / requirement	cube map [2], [3]	rhombic [4]	equirect. [1]	dyadic [5]
(i) same statistics as perspective projection	almost Yes	No*	No	No
(ii) uniform sampling on the sphere	No	almost Yes	No	approx. Yes
(iii) connectivity preservation	No	No	No	No

* Due to fitting rhombi to rectangles

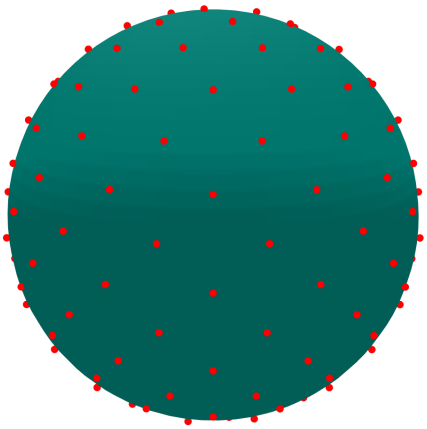


Fig. 1. Uniform sampling of the sphere.

This work was partially supported by the Cominlabs excellence laboratory with funding from the French National Research Agency (ANR-10-LABX-07-01) and by the Brittany Region (Grant No. ARED 9582 InterCOR).

video compression, as the bitrate devoted to intracoded frames represent a significant portion of the coded video bitrate. Therefore, reducing the compression rate of intra coded frames will have a significant impact on the overall video compression performance.

The proposed approach rely on a quasi-uniform sampling of the sphere (as illustrated in Fig. 1) to get a pixelization of the spherical content, called HEALPix [11]. Then, a complete image coder is built. This coder follows the key steps of 2D plane image compression but are defined on the sphere. More precisely, from the obtained discrete representation of the spherical content, a partitioning of the content into spherical blocks (S-blocks) is performed. Both pixelization and S-block partitioning are presented in Section II. Then, a scanning order is proposed on the sphere. This order induces the causal information that can be used to process the current S-block. Then, prediction on a S-block level is proposed in order to exploit redundancies between S-blocks. Scanning order and prediction are presented in Section III. Finally, the output of the prediction is further processed to remove within the S-block redundancies. To do so, a graph transform is used (see Section III), similar to the transform used in [12].

II. Sphere pixelization and S-block partitioning

Uniform sampling of the sphere is obtained with the Hierarchical Equal Area isoLatitude Pixelization (HEALPix) scheme [11], a popular sampling scheme used in cosmology and astrophysics. The process starts with a tessellation of the sphere into 12 equal area regions, see Fig. 2(a). Each region is a quadrilateral with curvilinear non-geodesic boundaries. The centers of the 12 regions are located on only three isolatitude circles. Then, to increase the resolution, each region is further divided into 4 equal area regions, see Fig. 2(b). Finally, the partitioning is repeated to reach the desired resolution, see Fig. 2(c). Once the desired resolution has been achieved, the pixelization is obtained by assigning one pixel to the center of each region.

The first and principal property of the HEALPix pixelization is to yield uniform sampling of the sphere. Second, the isolatitude property allows fast and efficient processing of huge size data. These properties were the original requirements that motivated the construction of HEALPix and are also important in the context of image compression. Moreover, HEALPix offers other properties, which are of great interest in our context. First, pixels can be gathered into blocks of pixels, where the number of pixels in each side is a power of two. The blocks are called spherical blocks (S-blocks) as they differ by construction from planar 2D blocks. Second, the isolatitude property allows to define S-blocks which are spread on isolatitude circles i.e. with horizontal neighbors. This is a figure of merit, as the correlation between neighboring horizontal S-blocks is greater than between vertical ones.

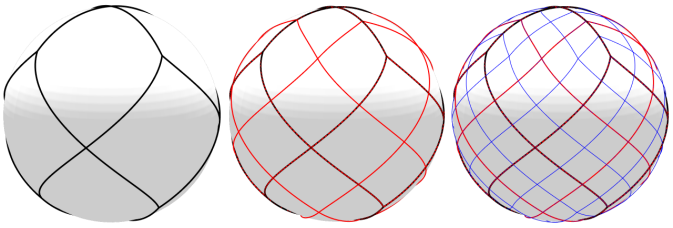


Fig. 2. HEALPix hierarchical pixelization of the sphere. (a) Base-resolution decomposition into 12 pixels. (b) Second decomposition into 48 pixels. (c) Third decomposition into 192 pixels.

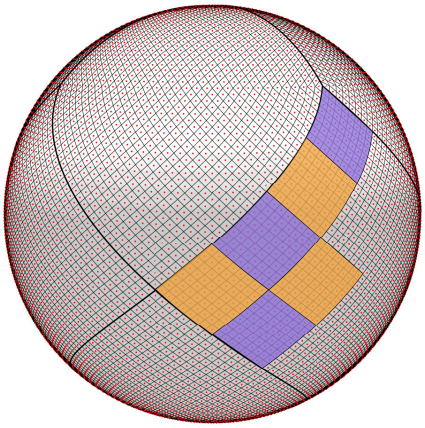


Fig. 3. Partitioning of the sphere into S-block of 8x8 pixels. An example is shown with a resolution of 12288 pixels. Pixel centers are depicted by red dots, and base-resolution tessellation is shown with black boundaries. An 8x8 pixel S-block (depicted in yellow or violet) corresponds to the third decomposition of the base-resolution tessellation.

III. Proposed coder

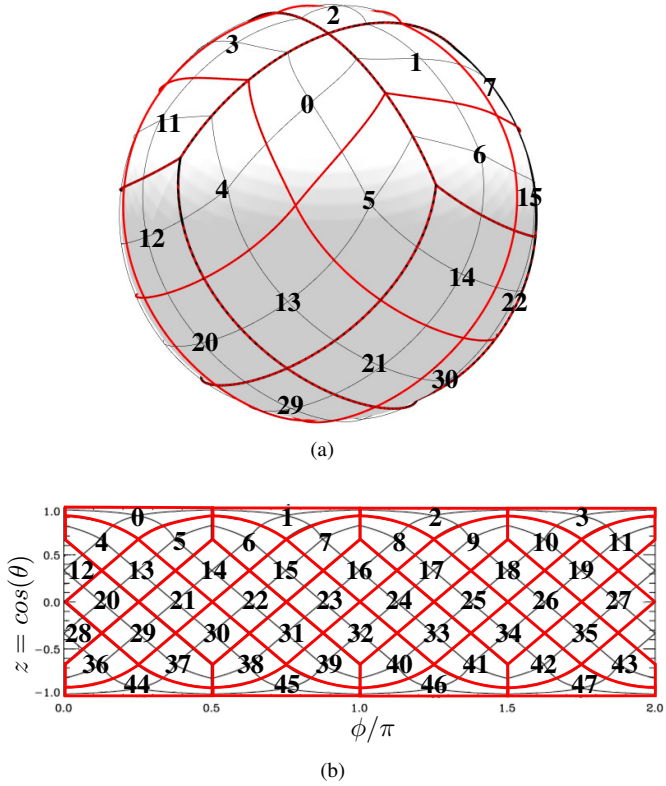
Prediction and transform are two key steps in image compression. Their goals are to exploit the short and long range spatial redundancies of an image. More precisely, the prediction exploit the redundancies between S-blocks whereas the transform rather exploits local redundancies i.e., within a S-block. In this section, we present a novel image coder, where these two types of redundancy are exploited directly on the sphere, and not in a projected domain as classically done.

A. Intra-prediction and scanning order

From the partitioning of the sphere into S-blocks presented in Section II, one can define a block based prediction algorithm to remove redundancies between adjacent S-blocks. The principle is to predict the current S-block from neighboring S-blocks that have already been encoded/decoded and are therefore available at the decoder. This requires to define a scanning order of the S-blocks.

One property of omnidirectional images is that the correlation between S-blocks with same latitude is higher than between S-blocks with the same longitude. Therefore, we propose a horizontal scanning order, where the S-blocks are processed ring by ring. Moreover, at the beginning of the compression, less neighboring S-blocks are available. So it is important to start the compression with smooth S-blocks

for which directional prediction is not needed. This motivates to start the scanning at the north pole since, in 360°-degree content, smooth scenes such as sky or ceiling are usually depicted at the north pole. In a nutshell, the coder starts scanning at the north pole, then it moves down from the north pole to the south pole along each iso-latitude ring (Fig. 4). It is worth noting that the proposed scanning order is fixed and therefore does not require any signalization.



(a)

(b)

Fig. 4. Proposed scanning order for the case when S-blocks are 2x2 and sphere is pixelized into 192 pixels. S-blocks are processed ring by ring starting from the north pole. Red lines are the border of the S-blocks and black thin lines are the border of the pixels. (a) S-blocks on the sphere. (b) Cylindrical projection of S-blocks and pixels.

The prediction consists in predicting the current S-block from neighboring S-blocks. The neighboring S-blocks are the closest (to achieve reliable prediction) and already decoded (to avoid error propagation) S-blocks. Based on our proposed scanning order, the neighboring S-blocks are the ones placed north-east, north, and north-west of the current S-block. For instance, the S-block with index 13 is predicted from the S-blocks with index 4, 0 and 5, whereas the S-block with index 5 is predicted from the only already decoded neighboring S-block 0 (see Fig. 4a and Fig. 4b). Note that since the neighboring S-blocks are defined directly on the sphere, the connectivity between S-blocks is preserved. This is different from the compression methods based on a 2D projection followed by the classical 2D plane intra prediction algorithm [13].

B. Residual coding with graph transform

The residual signal is generated by subtraction of the prediction signal from the current S-block. To further decorrelate the residual signal, a transform is applied to the residual signal. When the signal is defined on a 2D-regular grid, the 2D-Discrete Cosine Transform (DCT) [13] is very efficient to exploit local dependencies. In our context, the signal is defined on the sphere. Therefore, we propose to apply a transform that takes the sphere geometry into account. Indeed, a Graph Fourier Transform (GFT) [14] is defined based on the geodesic distance between pixel samples.

More precisely, for each S-block we construct a weighted undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, where the set of the nodes in the graph $\mathcal{V}$ represents the set of pixels, $\mathcal{E}$ is the set of edges which represents the connectivity between pixels, and $\mathbf{W}$ is the weighted adjacency matrix. We use 8-connected neighbors (except for the boundary pixels of the S-block which have less neighbors) to define $\mathcal{E}$. We use the weighted adjacency matrix $\mathbf{W}$ suggested in [12]

$$W_{ij} = \begin{cases} exp\Big(-\frac{d(x_i,x_j)^2}{\rho^2}\Big) & \text{if pixels i and j are neighbors} \\ 0 & \text{otherwise,} \end{cases}$$

where $d(x_i, x_j)$ represents the geodesic distance between pixels i and j, and

$$\rho = \frac{1}{|\mathcal{E}|} \sum_{(v_i,v_j)\in\mathcal{E}} d(x_i, x_j)$$

is the average geodesic distance over all connected pixels in the S-block.

Following [14], the combinatorial graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W}$ can be used to define a Fourier basis on the graph. The degree matrix $\mathbf{D}$ is a diagonal matrix where $D_{ii} = \sum_j W_{ij}$. By construction, $\mathbf{L}$ is symmetric positive semi-definite and has a complete set of orthonormal eigenvectors $\mathbf{U} = [\mathbf{u}_1, ..., \mathbf{u}_N]$ with non-negative eigenvalues $\lambda_1, ..., \lambda_N$ where N is the number of pixels in the S-block. The graph Fourier basis is defined as the Laplacian eigenvectors. The GFT of a signal $\mathbf{s} \in \mathbb{R}^N$ is its projection on the eigenvectors given by $\mathbf{U}$

$$\hat{\mathbf{s}} = \mathbf{U}^T\mathbf{s}.$$

The coefficients are then quantized and compressed using an arithmetic coder.

Finally, to avoid error propagation, a DPCM like encoder scheme [15] is performed as in all classical compression schemes [13]. More precisely, encoded S-blocks are decoded and used as a reference to predict next S-blocks.

IV. Experimental results

In this section, we compare the proposed spherical coder with several baselines. In the following, the rate of each coding scheme corresponds to the amount of bits that are necessary to describe the compressed image. Computing the quality loss due to 360° image compression is less straightforward since the input images differ between the different coding schemes. We adopt the strategy proposed in [5]. Since 360° images are

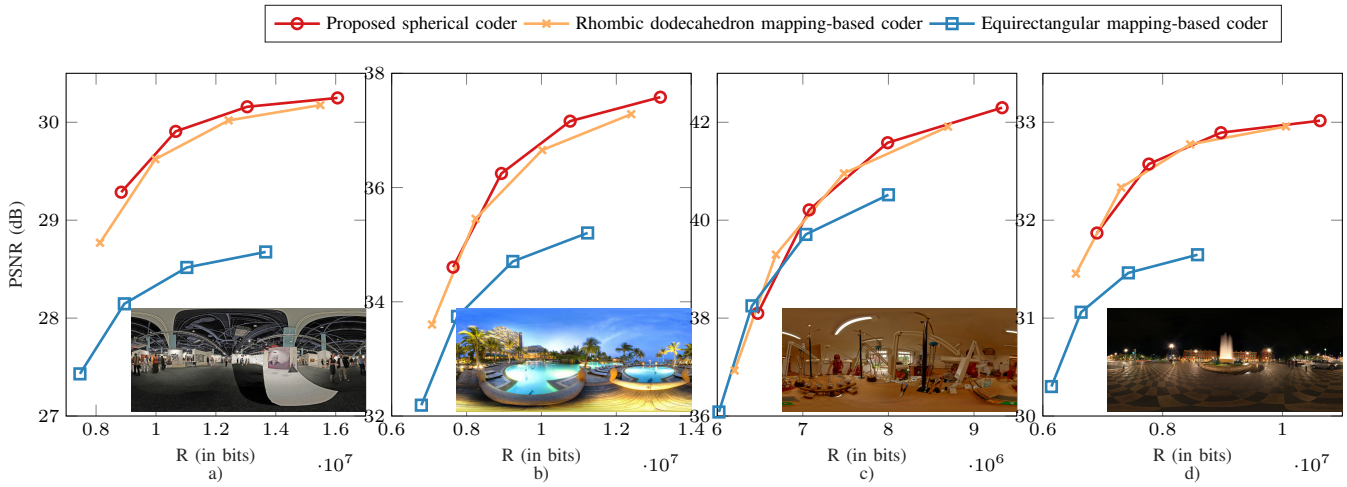


Fig. 5. RD curves of 4 images for different coding methods. (a) Image Exhibition. (b) Image Pool. (c) Image workshop. (d) Image Plaza.

aimed to be partly visualized by users, the compression loss is estimated on the viewport images. For that purpose, we take four high resolution panoramic images from [16] which are in equirectangular format, from which we generate several ground-truth viewports. Then, to perform a fair comparison of the compression performances of the schemes, we follow the recommendations of [5]. In particular, we down-sample the input equirectangular to a lower resolution such that each method has the same number of input pixels to process. Hence no method is favored over another. Then the same viewports are generated from the compressed images and the PSNR is calculated on each of them. The final given PSNR value corresponds to the PSNR averaged over all viewports.

We compare our proposed spherical coder with two mapping-based baselines: the equirectangular and the rhombic dodecahedron [4]. The number of pixels for the three methods have been set to be almost identical (equal to 3145728 pixels). For the equirectangular-based coder we keep the aspect ratio of the down-sampled equirectangular image to be equal to the ground truth. The rate-distortion plots are displayed in Fig. 5. We can see that our spherical coder clearly outperforms the equirectangular-mapping based coder. More interestingly, the proposed spherical coder also outperforms Rhombic-dodecahedron mapping-based coder. In particular, Table II shows that the proposed method allows to reduce the rate by 2% on average, and up to 5.6%, with respect to the method based on a 2D mapping of the spherical content onto the faces of the rhombic dodecahedron [4]. Whereas both approaches rely on the same HEALPIX pixelization technique, we can conclude that processing the data directly on the sphere is more efficient than mapping and perform 2D processing. This validates the intuitions brought by our spherical coder.

TABLE II
BD-RATE GAIN OF THE PROPOSED METHOD WITH RESPECT TO THE [4].

	Exhibition	Pool	Workshop	Plaza	**Average**
BD-rate gain	**-5.63** %	**-2.89** %	0.20 %	**-0.12** %	**-2.11** %

V. Conclusion

In this paper, we have proposed a new coder for 360° images which performs the compression directly on the sphere. Based on a uniform sampling of the sphere, our coder extends the regular tools of image compression to the spherical domain. Experiments demonstrate the benefits of such an approach, in particular with respect to conventional non-uniform sampling methods, and also based on a uniform sampling method which is mapped to several 2D image planes for compression.

References

[1] J. Snyder, *Flattening the Earth: Two Thousand Years of Map Projections*. University of Chicago Press, 1993.

[2] K.-T. Ng, S.-C. Chan, and H.-Y. Shum, "Data compression and transmission aspects of panoramic videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 1, pp. 82–95, Jan 2005.

[3] E. Kuzyakov and D. Pio, "Next-generation video encoding techniques for 360 video and vr," 2016. [Online]. Available: https://code.fb.com/virtual-reality/next-generation-video-encoding-techniques-for-360-video-and-vr/

[4] C. Fu, L. Wan, T. Wong, and C. Leung, "The rhombic dodecahedron map: An efficient scheme for encoding panoramic video," *IEEE Transactions on Multimedia*, vol. 11, no. 4, pp. 634–644, June 2009.

[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*, Sep. 2015, pp. 31–36.

[6] F. De Simone, P. Frossard, N. Birkbeck, and B. Adsumilli, "Deformable block-based motion estimation in omnidirectional image sequences," in *2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*, Oct 2017, pp. 1–6.

[7] B. Vishwanath, T. Nanjundaswamy, and K. Rose, "Rotational motion model for temporal prediction in 360 video coding," in *2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*, Oct 2017, pp. 1–6.

[8] ——, "Motion compensated prediction for translational camera motion in spherical video coding," in *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)*, Aug 2018, pp. 1–4.

[9] M. Rizkallah, F. De Simone, T. Maugey, C. Guillemot, and P. Frossard, "Rate distortion optimized graph partitioning for omnidirectional image coding," in *2018 26th European Signal Processing Conference (EUSIPCO)*, Sep. 2018, pp. 897–901.

[10] I. Tosic and P. Frossard, "Low bit-rate compression of omnidirectional images," in *2009 Picture Coding Symposium*, May 2009, pp. 1–4.

[11] K. M. Górski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann, "HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere," *The Astrophysical Journal*, vol. 622, no. 2, pp. 759–771, apr 2005.

[12] N. Perraudin, M. Defferrard, T. Kacprzak, and R. Sgier, "Deepsphere: Efficient spherical convolutional neural network with healpix sampling for cosmological applications," *Astronomy and Computing*, vol. 27, pp. 130 – 146, 2019.

[13] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.

[14] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, May 2013.

[15] K. Sayood, *Introduction to Data Compression, Third Edition (Morgan Kaufmann Series in Multimedia Information and Systems)*, 5th ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2018.

[16] J. Xiao, K. A. Ehinger, A. Oliva, and A. Torralba, "Recognizing scene viewpoint using panoramic place representation," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 2695–2702.