POINT CLOUD COMPRESSION INCORPORATING REGION OF INTEREST CODING

Gustavo Sandri[1]*, Victor F. Figueiredo*[1]*, Philip A. Chou*[2]*, and Ricardo de Queiroz*[1]

[1]Universidade de Brasília, Brasília, Brazil; [2]Google, Seattle, WA USA

ABSTRACT

We introduce Region-of-Interest (ROI) coding for point cloud attributes, using an input-weighted distortion measure where the weights are determined by the ROI. In terms of coding, we use the Region Adaptive Hierarchical Transform (RAHT), which relies on a set of weights. We use a measure-theoretic interpretation of RAHT to determine that the weights of the transform should be set to the weights of the distortion measure. The ROI is chosen as the 3D region of the face, which is detected from a set of 2D projections using the well-known Viola-Jones algorithm. Experimental results show subjectively meaningful improvements (7-8 dB PSNR) in a face ROI with subjectively insignificant degradations (under 1 dB PSNR) in the non-ROI.

***Index Terms*—** Point cloud, region of interest, RAHT

1. INTRODUCTION

Point clouds, which represent the 3D world by sampling, have become increasingly important in recent years because of the proliferation of computational imaging aimed at 3D sensing.

Like raw images and video, point clouds and point cloud sequences contain large amounts of data. Therefore point cloud compression is required in any practical application. MPEG is currently standardizing a format for point cloud compression (PCC) to serve this purpose [1].

Like images and video, point clouds often have regions of interest (ROI) that have special semantic or perceptual significance or salience – for example faces – for which preservation of high fidelity during compression could be important. For images and video, ROI-driven compression, or *ROI coding*, is well studied. (See, e.g., [2].) However, for point clouds, there is little prior literature on ROI-coding. In this paper, we propose ROI coding for point clouds.

Point clouds consist of geometry and attributes. The geometry part of a point cloud is simply a list of 3D positions $\{\mathbf{x}_i\} = \{(x_i, y_i, z_i)\}$, $i = 1, \ldots, N$, where N is the number of points in the point cloud. The attribute part of the point cloud is a corresponding list of attributes $\{\mathbf{a}_i\} = \{(a_{i1}, \ldots, a_{iD})\}$, i=1, …, N, where D is the number of attributes per point. Commonly, the attributes include color components (Y_i, U_i, V_i), but may also include transparency, normals, motion vectors, and so forth. Once the geometry is given, the attributes may be thought of as a signal defined on a set of points.

Most point cloud codecs in the literature compress the geometry first and then compress the attributes given the geometry. Typical approaches to attribute coding include transform coding using the Graph Fourier Transform (GFT) [3, 4, 5, 6, 7, 8], the Gaussian Process Transform (GPT, which is the KLT of a Gaussian Process) [9, 10], and the Region Adaptive Hierarchical Transform (RAHT) [11, 12]. RAHT, unlike the GFT or GPT, does not require an eigen-decomposition, and has been one of the transforms initially adopted into MPEG PCC [1]. ROI coding for point clouds may be applicable to geometry, attributes, or both. For example, for geometry compression, the ROI may be used to adjust the geometric level of detail by adjusting an octree depth. However, in this paper we focus on ROI coding of attributes. For attribute compression, the ROI may be used for example to adjust the stepsizes of various transform coefficients, as is common in image and video ROI coding. However, we take a different approach.

Inspired by a recent measure-theoretic interpretation of RAHT [13], in which RAHT is shown to be a separable 3D wavelet transform that is orthonormal with respect to a uniform counting measure on the set of points, we achieve ROI coding by modifying the measure, and then using RAHT as usual. Modifying the measure is equivalent to modifying the weights in a weighted distortion measure. Hence one may consider our approach to ROI coding as modifying the distortion measure in accordance with the ROI, and then coding to minimize the modified distortion measure, formally known as an input-weighted distortion measure [14, 15, 16].

Our approach to ROI coding has the advantage that it is codec-independent. Instead of hacking each codec in a specific way to adjust its fidelity in the ROI, we advocate using the ROI to modify the distortion measure. The modified distortion measure is then available to any codec for its usual optimization, e.g., rate-distortion optimization. There is a simple mapping from the ROI to the distortion measure, which can be quantified (for example using perceptual experiments) independently of any particular codec. As our codec, we choose transform coding with RAHT because RAHT is automatically optimized for the distortion measure by virtue of its measure-theoretic interpretation.

Our contributions include the following. We believe this is the first published work on ROI coding for point clouds.

We take a novel approach to ROI coding by modifying the distortion measure. We use a measure-theoretic interpretation of the RAHT, which is a transform used in the MPEG PCC, to perform encoding under the modified distortion measure. Finally, we show how to use existing 2D ROI detection to accomplish 3D ROI detection. Experimental results reveal subjectively meaningful improvements (7-8 dB PSNR) in a face ROI with subjectively insignificant degradations (under 1 dB PSNR) in the non-ROI, with no alteration in the encoding other than the weights.

2. ROI-WEIGHTED DISTORTION MEASURE AND MEASURE-THEORETIC RAHT

We consider a single scalar attribute, say Y_i, on points $\mathbf{x}_i$, $i = 1, \ldots, N$, of the point cloud. The *weighted squared error* between $Y = \{Y_i\}$ and its reproduction $\hat{Y} = \{\hat{Y}_i\}$ is defined

$$d(Y, \hat{Y}) = \sum_i w_i (Y_i - \hat{Y}_i)^2, \tag{1}$$

where w_i, $i = 1, \ldots, N$, are the *weights*. If the weight w_i reflects the semantic or perceptual importance of the point $\mathbf{x}_i$, then $d(Y, \hat{Y})$ may be called a *ROI-weighted* distortion measure. A codec that minimizes this distortion measure subject to a rate constraint will tend to reproduce Y_i as $\hat{Y}_i$ with squared error inversely proportional to w_i. For example, suppose $w_i = 16$ when $\mathbf{x}_i \in R$ and $w_i = 1$ otherwise, where R is a region of interest. Then the root mean squared (RMS) error in the ROI will be about $1/4$ the RMS error elsewhere. This is a natural way to specify the objective of ROI coding.

The weights in the weighted square error may be interpreted as a measure. A *measure* on a measurable space is a function μ that assigns a real number to each set such that the measure of the union of any sequence of disjoint subsets is the sum of measures of the subsets. Examples of measures are the Lebesgue measure on the real line, the counting measure on the integers, and any probability measure on a probability space. We focus on $\mathbb{R}^3$ as the measurable space, and define $\mu(S) = \sum_{i:\mathbf{x}_i \in S} w_i$ for any measurable set $S \subseteq \mathbb{R}^3$.

The definition of measure induces the definition of the integral, $\int f(\mathbf{x}) d\mu(\mathbf{x}) = \liminf_{\epsilon \to 0} \sum_n \mu(\{f(\mathbf{x}) \geq n\epsilon\}) = \sum_i w_i f_i$, where $f_i = f(\mathbf{x}_i)$. In turn, the definition of the integral induces the definition of the inner product, $\langle f, g \rangle = \int f(\mathbf{x}) g(\mathbf{x}) d\mu(\mathbf{x}) = \sum_i w_i f_i g_i$. In turn, the definition of the inner product induces the definitions of orthogonality, $f \perp g \Leftrightarrow \langle f, g \rangle = 0$, and norm, $||f|| = (\langle f, f \rangle)^{1/2}$. Altogether, these induce a Hilbert space. The weighted squared error (1) between Y and $\hat{Y}$ is precisely the squared norm $||f - \hat{f}||^2$ of this Hilbert space, where $f_i = Y_i$ and $\hat{f}_i = \hat{Y}_i$.

RAHT is region-adaptive to remain orthonormal regardless of the locations of the points. Recently RAHT has been shown to be interpretable as a separable piecewise constant spline wavelet that is orthonormal with respect to the inner product $\langle f, g \rangle$ defined by the weights w_i [13]. Thus if the weights are set to the weights in the ROI-weighted distortion measure, the transform will remain orthonormal, and moreover uniform scalar quantization of the transform coefficients with the quantization stepsize set to a constant will minimize the ROI-weighted distortion measure, at least at high rates.

To be specific, let $\mathbb{R}^3$ be partitioned uniformly into cubes of size $2^{-m} \times 2^{-m} \times 2^{-m}$, half-cubes of size $2^{-m} \times 2^{-m} \times 2^{-(m+1)}$, and quarter-cubes of size $2^{-m} \times 2^{-(m+1)} \times 2^{-(m+1)}$, and let $\mathcal{F}_{3m}$, $\mathcal{F}_{3m+1}$, and $\mathcal{F}_{3m+2}$ be the spaces of all functions $f_\ell : \mathbb{R}^3 \to \mathbb{R}$ that are piecewise constant on these blocks, for $\ell = 3m$, $3m+1$, and $3m+2$, respectively. The nested sequence of function spaces $\mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \cdots \subseteq \mathcal{F}_\ell \subseteq \mathcal{F}_{\ell+1} \subseteq \cdots$ approximates ever more finely (with respect to the norm, i.e., the weighted squared error) the space of piecewise continuous functions.

Now let $B_{\ell,n}$ denote a block at level ℓ indexed by n, let $1_{B_{\ell,n}}(\mathbf{x})$ be its indicator function, and let $w_{\ell,n} = \mu(B_{\ell,n})$ be its measure. Then $\mathcal{F}_\ell$ is spanned by the basis functions

$$\phi_{\ell,n}(\mathbf{x}) = w_{\ell,n}^{-1/2} 1_{B_{\ell,n}(\mathbf{x})}, \tag{2}$$

which are orthogonal to each other and are normalized with respect to the inner product and norm induced by the weighted measure. Similarly, let $B_{\ell+1,n_0}$ and $B_{\ell+1,n_1}$ denote the sub-blocks of $B_{\ell,n}$, and let $\mathcal{G}_\ell$ be the orthogonal complement of $\mathcal{F}_\ell$ in $\mathcal{F}_{\ell+1}$. Then $\mathcal{G}_\ell$ is spanned by the basis functions

$$\psi_{\ell,n}(\mathbf{x}) = \frac{-w_{\ell+1,n_0}^{-1} 1_{B_{\ell+1,n_0}}(\mathbf{x}) + w_{\ell+1,n_1}^{-1} 1_{B_{\ell+1,n_1}}(\mathbf{x})}{(w_{\ell+1,n_0}^{-1} + w_{\ell+1,n_1}^{-1})^{-1/2}} \tag{3}$$

which are orthogonal to each other and to the functions (2), and are normalized, as can be verified by the diligent reader. Thus any function $f_{\ell+1} \in \mathcal{F}_{\ell+1}$ can be written as

$$f_{\ell+1}(\mathbf{x}) = \sum_n F_{\ell,n} \phi_{\ell,n}(\mathbf{x}) + \sum_n G_{\ell,n} \psi_{\ell,n}(\mathbf{x}), \tag{4}$$

where the $F_{\ell,n} = \langle f_{\ell+1}, \phi_{\ell,n} \rangle$ are known as low-pass coefficients and the $G_{\ell,n} = \langle f_{\ell+1}, \psi_{\ell,n} \rangle$ are known as high-pass coefficients. After some algebraic manipulation, (2) and (3) can be expressed recursively as the "two-scale equations"

$$\phi_{\ell,n}(\mathbf{x}) = a\phi_{\ell+1,n_0} + b\phi_{\ell+1,n_1} \tag{5}$$

$$\psi_{\ell,n}(\mathbf{x}) = -b\phi_{\ell+1,n_0} + a\phi_{\ell+1,n_1}, \tag{6}$$

where $a = \frac{\sqrt{w_{\ell+1,n_0}}}{\sqrt{w_{\ell,n}}}$ and $b = \frac{\sqrt{w_{\ell+1,n_1}}}{\sqrt{w_{\ell,n}}}$. Substituting these into the definitions of $F_{\ell,n}$ and $G_{\ell,n}$, we obtain

$$\begin{bmatrix} F_{\ell,n} \\ G_{\ell,n} \end{bmatrix} = \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \begin{bmatrix} F_{\ell+1,n_0} \\ F_{\ell+1,n_1} \end{bmatrix}, \tag{7}$$

which is a Givens rotation whose angle of rotation depends on the relative weights of the sub-blocks.

RAHT applies (7) recursively to expand $f_L \in \mathcal{F}_L$ as

$$f_L(\mathbf{x}) = \sum_n F_{0,n} \phi_{0,n}(\mathbf{x}) + \sum_{\ell=0}^{L-1} \sum_n G_{\ell,n} \psi_{\ell,n}(\mathbf{x}), \tag{8}$$

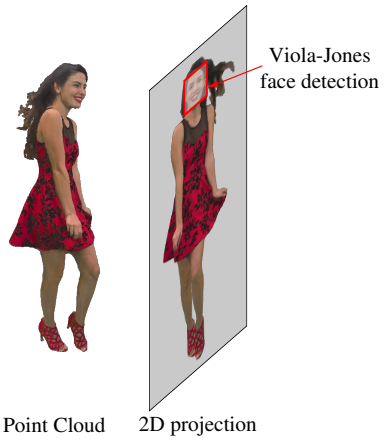


Fig. 1. Face detection in point clouds using projections and the Viola-Jones algorithm.

where L is chosen large enough so that each cube $B_{L,n}$ contains at most a single point, say $\mathbf{x}_i$ with value $f_i = f(\mathbf{x}_i)$. The number of coefficients is N, i.e., RAHT is critically sampled. (For details, see [13].) Note that $\phi_{L,n}(\mathbf{x}) = w_i^{-1/2} 1_{B_{L,n}}(\mathbf{x})$, and therefore $F_{L,n} = \langle f, \phi_{L,n} \rangle = w_i^{1/2} f_i$. This generalizes RAHT in [11], for which $w_i = 1$ for all points $i = 1, \ldots, N$.

The RAHT coefficients are uniformly scalar quantized with stepsizes $\Delta(F_{0,n})$ and $\Delta(G_{\ell,n})$, $\ell = 0, \ldots, L-1$, and are entropy coded. Because Givens rotations are orthonormal, energy is preserved. Thus the squared quantization error is

$$\sum_n (F_{0,n} - \hat{F}_{0,n})^2 + \sum_{\ell=0}^{L-1} \sum_n (G_{\ell,n} - \hat{G}_{\ell,n})^2 = \sum_{i=1}^{N} w_i (f_i - \hat{f}_i)^2, \tag{9}$$

which is the same as the ROI-weighted distortion measure (1) when $f_i = Y_i$. Since a constant stepsize $\Delta = Q_{step}$ minimizes the squared quantization error subject to an entropy constraint, at least at high rates [14], setting the stepsizes of the RAHT coefficients to a constant also minimizes the ROI-weighted distortion measure desired for ROI coding.

In summary, with RAHT, at the encoder, voxels in ROI should have initial weights set to $w_i = w$ and initial attributes scaled by $\sqrt{w}$. The decoder should scale back the attributes.

3. REGION-OF-INTEREST DETERMINATION AND SIGNALING

In our work, the ROI is chosen to be the subject's face in the point cloud. As our brain is more sensitive to artifacts introduced in the face on reconstructed images, we believe that prioritizing the subject's face quality during compression will lead to a better subjective quality.

The face as the ROI is identified as illustrated in Fig. 1. The point cloud is rotated to a given viewing angle defined by a pair of azimuth and elevation angles, and then projected to a 2D image. Using the Viola-Jones algorithm [17], the face is detected and the corresponding voxels are marked as *face*. The process is repeated for different viewing angles. We chose to vary the azimuth starting at $0°$ up to $250°$ in steps of $10°$, and vary the elevation from $-70°$ up to $90°$ in steps of $10°$. Those voxels marked as *face* in at least 20% of the viewing angles are marked as being in the ROI.

(a) $width = 1$ (b) $width = 2$ (c) $width = 4$ (d) $width = 8$

Fig. 2. Holes in the ROI

This process generates some holes in the ROI as some voxels in the face are occluded depending on the viewing angle. This is most visible on the cheeks, since most of the projections where Viola-Jones is able to detect the face are frontal or semi-frontal projections. Fig. 2(a) shows an example of these holes. To overcome this problem we expand the ROI to its neighboring voxels. The point cloud is regularly divided in cubes of fixed width, referred as blocks. If at least one voxel inside each cube is marked as ROI, all the other voxels inside the same cube are also marked as ROI. In Fig. 2, we show the result of expanding the ROI using cubes with different widths. Larger widths results in fewer holes and we decided to use cubes with width equal to 8 in the rest of this work. This ROI expansion method was chosen because of its simplicity, since it can be easily implemented using the Morton code associated to each voxel.

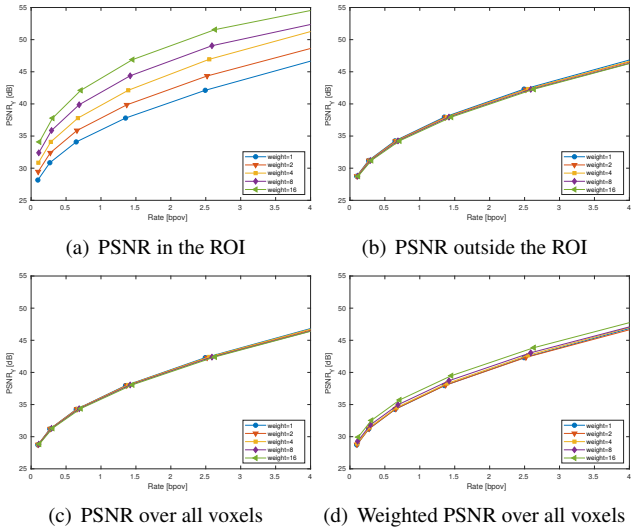
The ROI location needs to be conveyed to the decoder. Let there be M occupied blocks, so that we need to encode a binary vector $b = [b_0, b_1, \ldots, b_{M-1}]$ indicating whether each block belongs or not to the ROI. If we sort the blocks along their Morton codes, to preserve neighborhoods, the b_i bits can be used to to generate a differential vector $\overline{b} = [\overline{b_0}, \overline{b_1}, \ldots, \overline{b_{M-1}}]$ where

$$\overline{b_i} = \begin{cases} b_0 & i = 0 \\ 1 & b_{i-1} \neq b_i, i > 0 \\ 0 & b_{i-1} = b_i, i > 0 \end{cases}. \tag{10}$$

Vector $\overline{b}$ has long sequences of zeros. It is encoded with an algorithm based on the run-length Golomb-Rice coder, with the exception that only the run-lengths are encoded with Golomb-Rice. Other binary coders may be used as well.

4. EXPERIMENTAL RESULTS

To test our proposed encoder we used 6 point clouds: Boxer, Longdress, Loot, Redanblack, Soldier and Thaidancer, all voxelized with depth 10 (i.e. $1024 \times 1024 \times 1024$ voxels),



(a) PSNR in the ROI (b) PSNR outside the ROI

(c) PSNR over all voxels (d) Weighted PSNR over all voxels

Fig. 3. Average rate-distortion curves.

yielding 849452, 857966, 805285, 757691, 1089091 and 689953 occupied voxels, respectively [18, 19].

Fig. 3 shows the average rate-distortion curves computed for the 6 tested point clouds. Bits for the side information are included. Weights for ROI voxels were set to $w = 1, 2, 4, 8, 16$, while weights outside the ROI have weight 1. In Fig. 3(a) the PSNR is computed only for voxels in the ROI. When $w = 1$ there is no difference in weight for voxels inside and outside the ROI and, in this particular case, there is no need to encode vector $\bar{b}$. As w increases, the transform favors voxels in the ROI and we see an increase in PSNR for any given rate, because larger weights for voxels in the ROI result in better reconstruction quality. The effect is the opposite for voxels outside the ROI (Fig. 3(b)), since there is a transfer of bits from the rest to the ROI, for the same bit-rate. The drop in overall PSNR is negligible (Fig. 3(c)), while the weighted PSNR (i.e. $10 \log 255^2 / WMSE$, where $WMSE$ is the weighted mean squared error), slightly improves with $w > 1$ (Fig. 3(d)).

In Fig. 4, the point cloud Thaidancer was encoded with different ROI weights. Subjectively, Fig. 4(b) seems to have a better quality since our brain is more sensitive to artifacts in the face than in rest of the scene. Fig. 5 shows a close up of the face for the reconstructed point clouds shown in Fig. 4.

5. CONCLUSION

We introduced ROI coding for point clouds, taking a novel approach to ROI coding by modifying the distortion measure to a weighted distortion measure. The weights of the weighted distortion measure are reflected in the measure under the RAHT transform. To detect the 3D ROI, we combined 2D ROI from the well-known Viola-Jones algorithm. Experimental results reveal subjectively meaningful improvements (7-8 dB PSNR) in the ROI with subjectively insignificant degradations (under 1 dB PSNR) outside the ROI, with no change in encoder or decoder complexity.

Future work includes optimizing the ROI weights using perceptual studies, extending the approach to multi-level weights (e.g., from saliency maps), optimizing the side information, and applying ROI coding to point cloud geometry.

(a) $weight = 1$, $Q_{step} = 128$, $file\ size = 11225\ bytes$ (b) $weight = 16$, $Q_{step} = 152$, $file\ size = 11162\ bytes$

Fig. 4. Point cloud Thaidancer ($N_{vox} = 689953$) coded with different weights for voxels in the ROI. The Q_{step} was adjusted to result in similar file sizes.

(a) $weight = 1$ (b) $weight = 16$

Fig. 5. Close up in the face of the reconstructed point clouds shown in Fig. 4

6. REFERENCES

[1] S. Schwarz, M. Preda, V. Baroncini, M. Budagavi, P. Cesar, P. A. Chou, R. A. Cohen, M. Krivokuća, S. Lasserre, Z. Li, J. Llach, K. Mammou, R. Mekuria, O. Nakagami, E. Siahaan, A. Tabatabai, A. Tourapis, and V. Zakharchenko, "Emerging MPEG standards for point cloud compression," *IEEE J. Emerging Topics in Circuits and Systems*, accepted for publication.

[2] H. Hadizadeh and I. V. Bajić, "Saliency-aware video compression," *IEEE Trans. Image Process.*, vol. 23, Jan. 2014.

[3] C. Zhang, D. Florêncio, and C. Loop, "Point cloud attribute compression with graph transform," in *2014 IEEE Int'l Conf. Image Processing (ICIP)*, Oct 2014.

[4] D. Thanou, P. A. Chou, and P. Frossard, "Graph-based motion estimation and compensation for dynamic 3d point cloud compression," in *IEEE Int'l Conf. Image Processing (ICIP)*, Sept 2015.

[5] ——, "Graph-based compression of dynamic 3d point cloud sequences," *IEEE Trans. Image Processing*, vol. 25, no. 4, April 2016.

[6] E. Pavez and P. A. Chou, "Dynamic polygon cloud compression," in *IEEE Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP)*, March 2017.

[7] E. Pavez, P. A. Chou, R. L. de Queiroz, and A. Ortega, "Dynamic polygon cloud compression," *CoRR*, vol. abs/1610.00402, 2016. [Online]. Available: http://arxiv.org/abs/1610.00402

[8] R. A. Cohen, D. Tian, and A. Vetro, "Attribute compression for sparse point clouds using graph transforms," in *IEEE Int'l Conf. Image Processing (ICIP)*, Sept 2016.

[9] P. A. Chou and R. L. de Queiroz, "Gaussian process transforms," in *IEEE Int'l Conf. Image Processing (ICIP)*, Sept 2016.

[10] R. L. de Queiroz and P. A. Chou, "Transform coding for point clouds using a Gaussian process model," *IEEE Trans. Image Processing*, vol. 26, no. 8, Aug. 2017.

[11] ——, "Compression of 3D point clouds using a region-adaptive hierarchical transform," *IEEE Trans. Image Process.*, vol. 25, no. 8, Aug. 2016.

[12] G. Sandri, R. L. de Queiroz, and P. A. Chou, "Comments on "Compression of 3D Point Clouds Using a Region-Adaptive Hierarchical Transform"," *ArXiv e-prints*, May 2018. [Online]. Available: https://arxiv.org/abs/1805.09146v1

[13] M. Krivokuća, P. A. Chou, and M. Koroteev, "A volumetric approach to point cloud compression," *ArXiv e-prints*, Sep. 2018. [Online]. Available: https://arxiv.org/abs/1810.00484

[14] R. M. Gray and D. Neuhoff, "Quantization," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, Oct. 1998.

[15] T. Linder and R. Zamir, "High-resolution source coding for non-difference distortion measures: The rate-distortion function," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, Mar. 1999.

[16] J. Li, N. Chaddha, and R. M. Gray, "Asymptotic performance of vector quantizers with a perceptual distortion measure," *IEEE Trans. Inf. Theory*, vol. 45, no. 4, May 1999.

[17] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, Dec 2001, pp. I–I.

[18] E. d'Eon, B. Harrison, T. Myers, and P. A. Chou, "8i voxelized full bodies — a voxelized point cloud dataset," ISO/IEC JTC1/SC29/WG1 & WG11 JPEG & MPEG, input documents M74006 & m40059, Jan. 2017.

[19] M. Krivokuća, P. A. Chou, and P. Savill, "8i voxelized surface light field (8iVSLF) dataset," ISO/IEC JTC1/SC29/WG11 MPEG, input document m42914, Jul. 2018.