VIEWPOINT-AWARE BITRATE OPTIMIZATION FOR MULTI-ASSET 3D SCENES

Tomás M. Borges[*1,2] *Yago Sánchez*[2] *Cornelius Hellge*[2] *Ricardo L. de Queiroz*[3]

[1] Dept. of Electrical Engineering, UnB, Brasília, Brazil
[2] Fraunhofer HHI, Berlin, Germany
[3] Dept. of Computer Science, UnB, Brasília, Brazil

ABSTRACT

Efficient rate-distortion optimization (RDO) is a key challenge in immersive applications involving 3D scenes with heterogeneous assets, where uniform bitrate allocation fails to account for the viewpoint-dependent perceptual contribution of each asset to the rendered view. This paper proposes a low-complexity, viewpoint-aware coding framework that uses a Lagrangian RDO formulation based on asset importance, to guide bitrate allocation under dynamic viewing of such scenes. A simulation framework is also developed to assemble multi-asset 3D scenes and render 2D projections, to demonstrate the effectiveness of the proposed approach. Experiments with standard MPEG point cloud and dynamic mesh datasets, encoded using state-of-the-art MPEG codecs, show that the proposed framework enables the usage of different importance measures, yielding markedly different bitrate allocation behaviors, demonstrating the flexibility of the proposed approach. Results further show that rate-distortion performance is strongly dependent on scene layout and camera motion, further highlighting the need for viewpoint-aware RDO in immersive systems.

1. INTRODUCTION

The growing demand for immersive applications such as augmented reality (AR), virtual reality (VR), and free-viewpoint television (FTV) has led to significant advancements in the capture, transmission, and rendering of three-dimensional (3D) assets. These assets, often represented as 3D point clouds or 3D meshes, provide rich interactive environments that users can explore with six degrees of freedom (6DoF). However, the increasing complexity and size of 3D data, coupled with the need for real-time interaction, pose significant challenges for the efficient streaming and rendering of volumetric videos, particularly under bandwidth constraints. In scenarios such as 3D conferences, numerous assets must be seamlessly merged and rendered together in a single coherent 3D scene. These assets may be captured using different devices, from various locations, and often require integration into the same interactive environment in real-time. Therefore, it is essential that efficient compression and transmission methods are employed to facilitate the implementation of such applications.

User adaptation strategies involve taking into account what the user is actually seeing to adapt the delivery of content. A uniform treatment of data from different sources and at different positions, in the 3D scene, can lead to inefficient bandwidth use and to lower perceived quality, as more relevant parts of the scene, which are critical to quality of experience (QoE), are not prioritized. Additionally, at any given moment, large portions of volumetric data may be occluded or outside the user's field of view; exploiting viewpoint information therefore enables significant bitrate reduction without perceptible quality loss.

In this work, we propose a viewpoint-aware strategy for optimizing bitrate allocation across multiple 3D assets in the uplink of a simulated 3D meeting environment. This setup is implemented using a scene manager framework, illustrated in Fig. 1. Although our work does not model the downlink or explore deployment constraints, the scene manager could conceptually operate either remotely (e.g., in a split-rendering paradigm [1]) or locally alongside the viewer. Each 3D asset independently transmits its compressed bitstream (as a mesh or point cloud) to the scene manager, which then decompresses the data and assembles a coherent 3D scene. A 2D projection is rendered from this scene according to the viewpoint requested by the viewer. Finally, the rendered 2D frame is delivered to the user in an interactive streaming fashion. Given the complete scene configuration, we aim to answer the question of what bitrates should the scene manager negotiate with each asset in order to minimize the observed image distortion. Other aspects, such as transmission protocols, compression pipelines, or latency handling, fall outside the scope of this work.

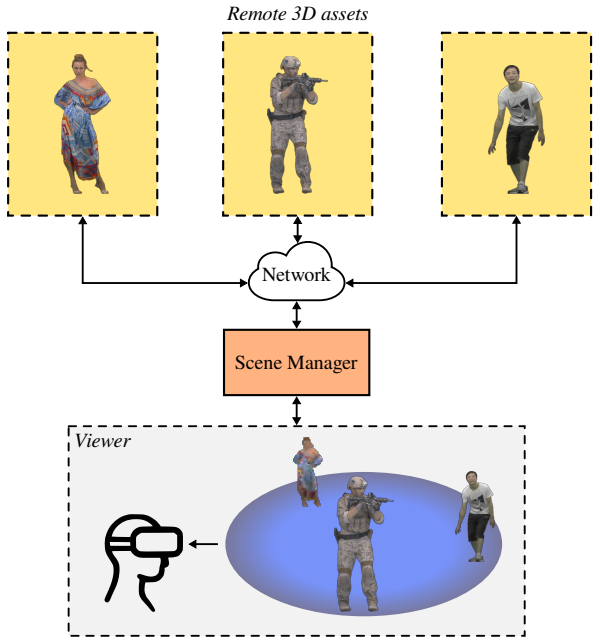


Fig. 1. Simplified framework for viewpoint-aware bitrate allocation between remote 3D assets and the scene manager.

*Work partially supported by CNPq under grant 88887.600000/2021-00.

Different techniques have been developed to account for the user's viewpoint to help the delivery of volumetric video [2–7]. A large body of work explores user-adaptive streaming, where delivery is adjusted according to the user's viewpoint [3–6, 8–14]. These approaches commonly rely on mechanisms such as tiling, level-of-detail (LoD) adaptation, distance- or visibility-based heuristics, or viewport-aware prioritization. Recent studies extend these ideas to point cloud and volumetric streaming using adaptive bitrate allocation, QoE-driven optimization, or tile-based rate control [3, 4, 6, 8, 14]. While effective, many existing approaches are system-specific, overly complex, or tightly coupled to particular representations, and fail to fully address the combined challenges of dynamic user behavior, occlusion, and the multitude of factors influencing QoE. In contrast, our work proposes a simple, general, and low-complexity viewpoint-aware bit-allocation framework applicable across heterogeneous 3D assets.

Several standards have been developed for the compression of volumetric content, including, Geometry-based PCC (G-PCC), for point cloud compression (PCC); and Video-based Dynamic Mesh Coding (V-DMC) for time-varying meshes [15–17]. G-PCC, uses the octree structure [18] to encode the geometry and the region adaptive hierarchical transform (RAHT) for the encoding of attributes [19]. In V-DMC, a video- and subdivision-based mesh coding solution [20], the input mesh is initially simplified and then subdivided, based on a variation of the subdivision wavelets [21].

2. RATE-DISTORTION MODELING AND OPTIMIZATION

2.1. Compression Control Strategy

In practice, controlling bitrate in 3D codecs (e.g., G-PCC and V-DMC) requires tuning multiple interdependent compression parameters, which significantly complicates rate-distortion optimization (RDO). In order to simplify this, we precomputed RD points using a range of predefined encoding configurations. The application therefore only needs to select the most suitable operating point from a discrete set of representations, rather than tuning low-level parameters. These representations mainly differ in geometry and texture quantization parameters (QPs), along with other codec-specific settings that affect rate and distortion.

Because 3D assets vary widely in size (number of points or faces), total bitrate is not directly comparable across assets. We therefore use normalized measures: bits per input voxel (bpiv) for point clouds and bits per input face (bpif) for meshes. Distortion is measured using projected mean squared error (PMSE) for point clouds and the image-based sampling metric (IBSM) for meshes [22].

Hence, we map all the K encoding parameters for an asset in a tensor, such that the m-th representation is achieved by having

$$\mathbf{Q}(m) = [\mathrm{QP}_{\mathrm{geo}}, \mathrm{QP}_{\mathrm{tex}}, par_1, par_2, ..., par_{K-2}]. \tag{1}$$

Then, the manager needs to choose a different RD point m from the set of available representations.

2.2. Distortion Modeling

For the targeted applications, 3D content is ultimately projected onto 2D images, as conventional displays remain the primary medium for end-user visualization. Traditional 3D metrics are therefore unsuitable, particularly when assets have different spatial resolutions, as point-wise distances are sensitive to sampling density.

We define a *scene* as a collection of 3D assets (or objects), placed in a common space where they can interact with each other. Considering a scene with N assets, each asset i is rendered using the m_i-th representation, which is one of the M different representations. Besides the representation choice, the individual distortion D of the i-th asset, as seen by the viewer, also depends on its position and the camera parameters used to capture the scene at a given viewpoint v,

$$D_i = D_i(m_i, \mathbf{X}_i(v), \mathbf{P}(v)), \tag{2}$$

where,

- $m_i \in \{1, 2, \ldots, M\}$ denotes the index of the representation selected for asset i, where M is the number of available representations;
- $\mathbf{X}_i(v)$ represents the 3D coordinates of the asset i in world space at viewpoint v;
- $\mathbf{P}(v)$ is the camera projection matrix, which includes both the intrinsic and the extrinsic parameters, and the camera translation vector.

We want to model the total distortion $\mathcal{D}$ from a viewpoint v, considering a set of chosen compressed versions

$$\mathbb{M} = \{m_1, m_2, \ldots, m_N\} \tag{3}$$

for each of the N assets in the scene,

$$\mathcal{D}(\mathbb{M}, v) = \sum_{i=1}^{N} D(m_i, v). \tag{4}$$

We assert that (2) can be simplified by considering two main components: *compression distortion*, which solely depends on the chosen compressed representation m_i; and *view-dependent distortion*, which depends on the camera configuration and viewpoint v, shaping how the chosen representation is perceived when rendered. In particular, since we are combining multiple assets in a single scene and analyzing their effect on the projected 2D image, MSE offers a convenient additive measure that supports straightforward comparison across projections. To make this metric independent of the camera, we take orthographic projections from different views from each asset's representation, then compute the MSE between the original and compressed projections and average the error to obtain the PMSE. For point clouds, PMSE is computed using projections from the six faces of a surrounding cube, while for meshes we use the IBSM metric [22], which similarly relies on projected MSE but samples more viewpoints using a Fibonacci sphere. IBSM is adopted here for convenience, as it is readily available in the V-DMC software. We argue that we can use projected metrics to measure the compression distortion effects from the 3D codecs in (2). We refer to this measured distortion as $\tilde{D}(m_i)$.

In order to take into account the view-dependent distortion effects of $\mathbf{X}_i(v)$ and $\mathbf{P}(v)$ in (2), we introduce an importance measure $\alpha_i(v)$ to ponder $\tilde{D}(m_i)$ according to the specific position and camera configuration of each asset in the scene. By combining these two factors, we can have a proxy for the total distortion (4) as:

$$\mathcal{D}(\mathbb{M}, v) \propto \tilde{\mathcal{D}}(\mathbb{M}, v) = \sum_{i=1}^{N} \alpha_i(v) \tilde{D}(m_i). \tag{5}$$

2.3. Importance Measure

Deciding what is important in a scene is highly subjective and depends on the application. For example, importance might be more

strongly weighted towards a person speaking in a video conference, an object that is close to the camera, or areas that occupy a significant portion of the viewer's field of view. The distortion model presented in this work is agnostic to how this importance measure is calculated, as long as $0 \leq \alpha_i(v) \leq 1$ and $\sum_i \alpha_i(v) = 1$, for each v.

Previous works [6, 8, 23] have considered factors such as the Euclidean distance of an asset from the camera and its projected area as relevant indicators for bitrate allocation. We also advocate to use an importance measure based on the occupied area (in pixels) of each asset in the projection of the viewpoint; we believe that this is a good generalization of the importance problem for most applications. We reason that this measure takes into account the camera parameters, the distance between assets and the camera, occlusions, and voxel distinguishability[1] as voxels get smaller (more than one voxel will be projected to a single pixel, making their area smaller). To allow fairness in the measure for assets with very distinct sizes (e.g., an adult and a child), we normalize the area occupied by each asset by the length of the diagonal of its bounding box, such that

$$\alpha_i(v) = \frac{\frac{\texttt{pixel\_count}_i(v)}{\texttt{diagonal}_i}}{\sum_{i=1}^{N}\left(\frac{\texttt{pixel\_count}_i(v)}{\texttt{diagonal}_i}\right)}. \tag{6}$$

2.4. Bit-Allocation Optimization

We want to select the set $\mathbb{M}$ of optimal representations for the N assets which minimize the scene's total distortion at each viewpoint v, as in (5), subject to

$$\mathcal{R}(\mathbb{M}) = \sum_{i=1}^{N} R(m_i) \leq \mathcal{R}_b, \tag{7}$$

where $\mathcal{R}_b$ is a given budget bitrate. Since the overall bitrate and distortion are additive across assets, and the RD points are usually bound by a well-behaved curve, we tackle the above problem by defining a Lagrangian cost function,

$$\begin{aligned}\mathcal{J}(\mathbb{M}, v) &= \mathcal{R}(\mathbb{M}) + \lambda\tilde{\mathcal{D}}(\mathbb{M}, v) \\ &= \sum_{i=1}^{N} R(m_i) + \lambda\alpha_i(v)\tilde{D}(m_i),\end{aligned} \tag{8}$$

where, λ is a Lagrange multiplier that balances the trade-off between rate and distortion, in our search for the minimum cost [24]. Moreover,

$$\begin{aligned}&\min_{\mathbb{M}}\left[\mathcal{R}(\mathbb{M}) + \lambda\tilde{\mathcal{D}}(\mathbb{M}, v)\right] = \\ &\min_{\{m_1,\ldots,m_N\}}\left[\sum_{i=1}^{N} R(m_i) + \lambda\alpha_i(v)\tilde{D}(m_i)\right] = \\ &\sum_{i=1}^{N}\min_{\{m_1,\ldots,m_N\}}\left[R(m_i) + \lambda\alpha_i(v)\tilde{D}(m_i)\right].\end{aligned} \tag{9}$$

Hence, we reduce the problem to minimize the Lagrangian function for each individual asset. By making $\lambda_i(v) = \lambda\alpha_i(v)$ as the individual Lagrangian multiplier for asset i at viewpoint v, its optimal representation $m_i^*|_{\lambda_i(v)}$ can be found by solving

$$\begin{aligned}m_i^*|_{\lambda_i(v)} &= \arg\min_{m_i}[J(m_i)] \\ &= \arg\min_{m_i}[R(m_i) + \lambda_i(v)\tilde{D}(m_i)].\end{aligned} \tag{10}$$

[1] Although display resolution, pixel density and the viewing distance also influence voxel distinguishability, we simplify the discussion by assuming constant viewing conditions and displays characteristics.

Note that, since $0 \leq \alpha_i(v) \leq 1$, we have that $0 \leq \lambda_i(v) \leq \lambda$.

Solving (10) becomes straightforward once the values of $R(m_i)$ and $\tilde{D}(m_i)$ are precomputed (or possibly estimated). For each asset and for each viewpoint, we calculate $\alpha_i(v)$, and then select the representation that has the lowest associated cost.

Note, also, that the case of uniform allocation implies

$$\alpha_i(v) = \frac{1}{N},\ \forall\, i, v, \tag{11}$$

i.e., uniform $\lambda_i(v)$, as per design.

3. EXPERIMENTS

3.1. Setup

Fig. 2 illustrates the details of the scene manager envisioned for the experiments. For each frame, the manager decodes each incoming 3D asset and uses the provided metadata to assemble a scene. Based on the user-specified viewpoint, it renders a 2D projection of this scene and calculates the importance value $\alpha_i(v)$ for each asset for the current viewpoint.

Given a Lagrange multiplier λ, the system then solves the optimization problem from Sec.2.4 to determine the optimal representation (i.e., optimal QPs) for each asset. These encoding settings are sent back to the remote sources and the compressed assets are, then, transmitted to the scene manager, accordingly. Here, for simplicity, we considered a clairvoyant prediction of the user's viewpoint, such that every frame is optimized for the *current* position, without delay.

The manager then renders a 2D frame and sends it to the user. If the user requests a new viewpoint, the updated parameters are fed back into the system, triggering a new round of importance evaluation, QP selection, and rendering. This setup enables the evaluation of different importance measures under dynamically changing viewpoints.

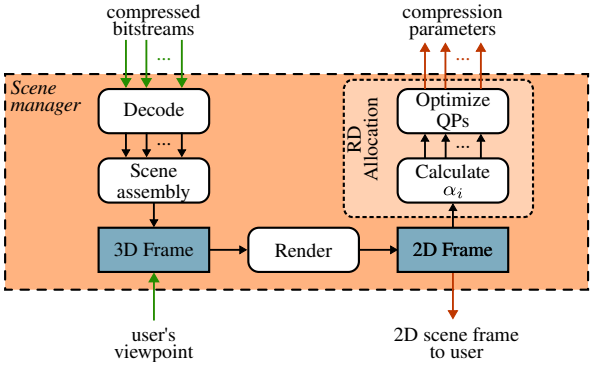


Fig. 2. Framework of the scene manager used for the simulations.

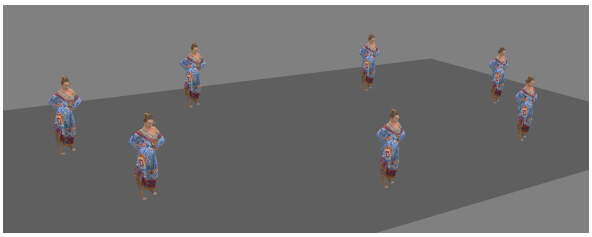
Each asset includes metadata to ensure correct placement in the scene, such as geometry resolution (bit depth), frame-to-world scale, bounding box dimensions, and indicators for vertical and forward axes. The geometry resolution and frame-to-world scale enable the coherent merging of assets, providing visual consistency and realism to the scene. Assuming a prior negotiation to determine broadcast resolution, we normalized assets to a common scale and resolution, opting for the lowest resolution after scaling to reduce complexity. The bounding box dimensions are used in positioning and calculating centroids for distance measures, while the vertical and forward axes ensure proper orientation of the assets.

For the scene assembly, the dimensions of the virtual stage must also be defined, as well as the type of 3D assets (meshes or point clouds). After assembling each 3D frame, user viewpoints are used to calculate the importance measures of each asset. Point cloud and meshes tested were human figures from MPEG's G-PCC and V-DMC respective Common Test Conditions (CTC) [25, 26].

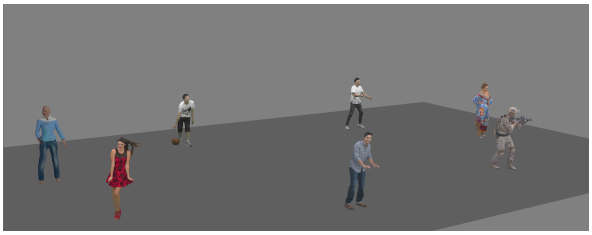
3.2. Tests

Experiments were conducted using a simulated scene manager to evaluate viewpoint-aware bit allocation across multiple 3D assets in three scenarios of increasing complexity (Fig. 3): (1) single static asset repeated at different positions (point clouds), (2) multiple static assets (point clouds), and (3) multiple dynamic assets (meshes).

For Tests 1 and 2, assets were compressed using G-PCC v23.0, with 16 geometry and 16 texture QPs, yielding 256 representations per asset. To ensure consistent rendering across assets, sparse point clouds resulting from geometry quantization were densified using nearest-neighbor interpolation (NNI) [27]. For Test 3, dynamic meshes were encoded with V-DMC v9.0 using 12 representations per asset to reflect a more realistic number of operating points.[2]



Test 1



Test 2

Test 3

Fig. 3. Visualizations of the three tests scenarios.

A camera path specifying the user's viewpoint over time was defined for each test, and six target bitrates were evaluated. For Tests 1 and 2, a shared 100-frame pan-zoom trajectory was used. For Test 3, a longer 300-frame path was defined to better emulate realistic user navigation, combining scene exploration with focused inspection of a single asset. The selected paths are arbitrary but composed of multiple segments with varying visibility and occlusion conditions, enabling segment-wise analysis of viewpoint/path effects. Occlusions and complete removals of assets from the field of view are expected to have the strongest impact on viewpoint-dependent RDO, as rate allocated to such assets does not translate into quality improvement.

In each scenario, seven rate-allocation strategies were evaluated:

i) **Equal Rate**: same bitrate for all assets (no RDO).
ii) **Proportional Rate**: bitrate proportional to each asset's visible area (no RDO).
iii) **Uniform**: equal importance for all assets with RDO.
iv) **Visibility-Aware Uniform**: same as (iii), but only assets visible in the current view are considered.
v) **Distance**: importance proportional to inverse distance.
vi) **Visibility-Aware Distance**: same as (v), but ignoring assets outside the view frustum.
vii) **Area**: importance proportional to normalized projected area, as in (6).

These strategies allow us to isolate the impact of optimization itself, visibility awareness, and the choice of importance model.

To evaluate rate-allocation performance, Bjøntegaard-Delta (BD) metrics [28] were computed for both rate and PSNR, using the uniform allocation strategy as reference. While MSE is employed during optimization due to its suitability for aggregating per-asset distortions, PSNR is reported for method comparison, as it remains a standard and widely adopted metric.

Fig. 4 reports per-frame results for Test 1. Dotted lines mark frames where the number of visible assets changes, delineating different camera path segments with distinct visibility and occlusion conditions. As expected, these transitions dominate viewpoint-dependent RDO, as bitrate assigned to non-visible assets does not contribute to perceived quality.

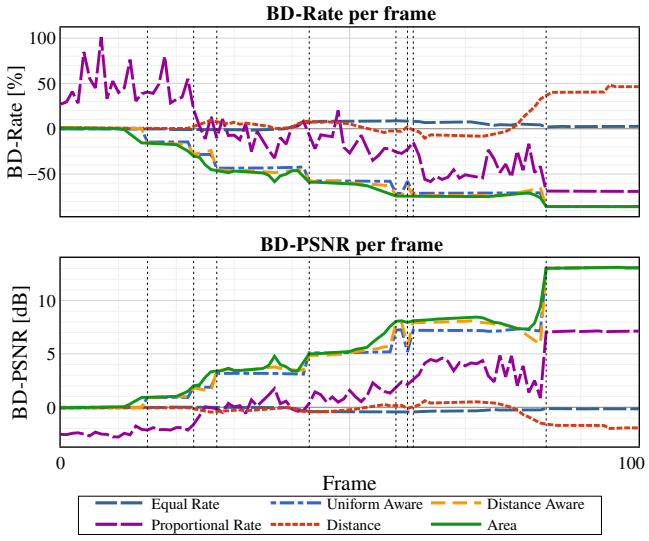


Fig. 4. BD-rate per frame Test 1, using the performance of the uniform importance strategy as reference. Dotted lines mark the frames at which assets enter or exit the camera's view frustum.

Rate-only strategies perform poorly, as allocating bitrate without considering distortion can select suboptimal representations. Incorporating visibility substantially improves performance, as seen for the visibility-aware uniform and distance-based methods. Distance

[2]Configuration files, camera paths, and results are available at https://datacloud.hhi.fraunhofer.de/s/ry6ZtStJ5Kao2Di.

alone, however, is an unreliable importance proxy, since proximity does not always correlate with projected size or visibility. The area-based strategy consistently achieves the best performance by jointly accounting for occlusion and projected size, which are key determinants of viewpoint-dependent distortion.

Although the chosen camera path is arbitrary, it can be interpreted as a combination of simpler path segments, each representing different typical viewpoint configurations. As a result, it serves to illustrate a range of user's viewpoints within a single experiment. For this reason, the average results over all frames, as illustrated in Fig. 5, should be interpreted with caution, as they are sensitive to the relative duration of individual path segments. For instance, if all assets were visible throughout all frames, the difference between the methods would have been significantly smaller. This suggests that no single method is universally optimal in all scenarios. Still, the proposed framework allows for performance assessment under different conditions and can adapt to varying importance methods as needed.

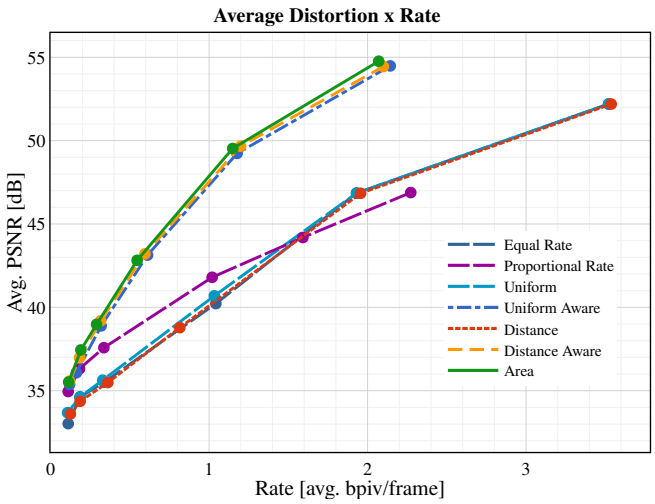


Fig. 5. Average Distortion vs Rate for Test 1.

The results in Fig. 6 for Test 2 show a behavior similar to that observed in Test 1, with one notable exception as the "Equal Rate" approach performed significantly worse in this case. This decline in performance is due to the differences in compression distortion curves across assets. Allocating the same bitrate to all assets, without accounting for distortion, leads to severe quality degradation.

Results from Test 3, illustrated in Fig. 7, follow the same trend as the previous ones, with "Equal Rate" and "Distance" strategies performing poorly. The "Distance" strategy, however, showed an improvement and outperformed the anchor. This is mainly due to the camera path, which kept most of the assets visible for a larger number of frames. This fact also explains the reduced gains seen in the "Area" method, when comparing to the previous tests.

4. CONCLUSION

This paper presented a viewpoint-aware framework for optimizing bit allocation across multiple 3D assets in immersive scenes. By distinguishing between compression-dependent and viewpoint-dependent distortions, we introduced an importance-driven strategy that enables more efficient bitrate allocation than uniform or heuristic approaches.

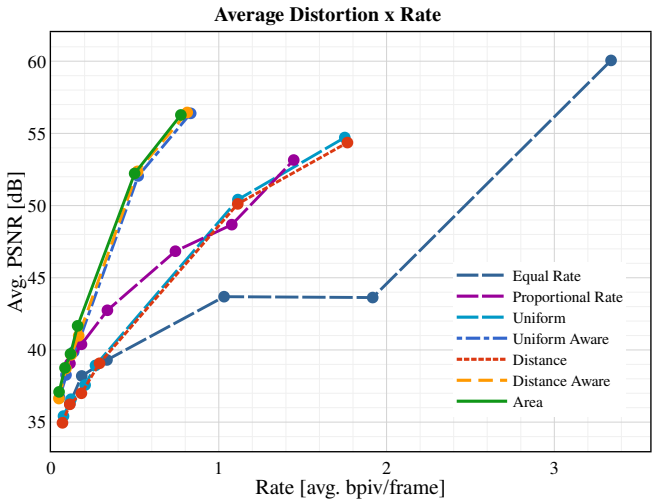


Fig. 6. Average Distortion vs Rate for Test 2.

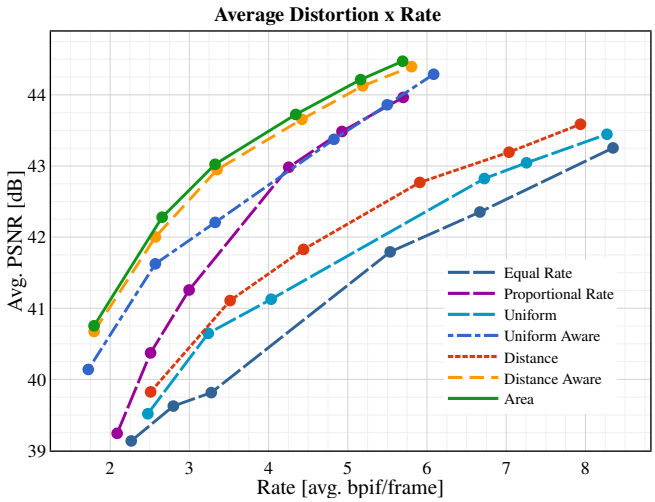


Fig. 7. Average Distortion vs Rate for Test 3.

Experiments show that different importance measures lead to substantially different allocation behaviors, with the area-based strategy achieving large BD-rate gains over uniform allocation on frame-averaged results. We also observed that performance is strongly context-dependent, varying with camera motion, scene layout, and the number of visible assets. Overall, the results demonstrate that incorporating perceptual relevance and scene awareness is essential for effective bitrate management in multi-asset volumetric systems.

5. REFERENCES

[1] J. Son, Y. Sanchez, C. Hellge, and T. Schierl, "Split Rendering with L4S Over 5G for Latency Critical Interactive XR Applications," *IEEE Commun. Mag.*, vol. 62, no. 8, pp. 46–52, Aug. 2024.

[2] I. Viola and P. Cesar, *Volumetric video streaming*, pp. 425–443, Elsevier, 2023.

[3] S. Petrangeli, G. Simon, H. Wang, and V. Swaminathan, "Dynamic adaptive streaming for augmented reality applications," in *Proc. IEEE Int. Symp. Multimedia*, Dec. 2019, pp. 56–567.

[4] J. Park, P. A. Chou, and J.-N. Hwang, "Rate-utility optimized streaming of volumetric media for augmented reality," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 9, pp. 149–162, 2019.

[5] S. Subramanyam, I. Viola, A. Hanjalic, and P. Cesar, "User centered adaptive streaming of dynamic point clouds with low complexity tiling," in *Proc. ACM Int. Conf. Multimedia*, Oct. 2020, pp. 3669–3677.

[6] J. v. d. Hooft, T. Wauters, F. De Turck, C. Timmerer, and H. Hellwagner, "Towards 6DoF HTTP adaptive streaming through point cloud compression," in *Proc. ACM Int. Conf. Multimedia*, Oct. 2019, pp. 2405–2413.

[7] Z. Liu, Q. Li, X. Chen, C. Wu, S. Ishihara, J. Li, and Y. Ji, "Point cloud video streaming: Challenges and solutions," *IEEE Netw.*, vol. 35, no. 5, pp. 202–209, Sept. 2021.

[8] M. Hosseini and C. Timmerer, "Dynamic adaptive point cloud streaming," in *Proc. ACM Packet Video Workshop*, June 2018.

[9] G. Cernigliaro, M. Martos, M. Montagud, A. Ansari, and S. Fernandez, "PC-MCU: point cloud multipoint control unit for multi-user holoconferencing systems," in *Proc. ACM NOSSDAV*, June 2020.

[10] J. Jansen, S. Subramanyam, R. Bouqueau, G. Cernigliaro, M. M. Cabré, F. Pérez, and P. Cesar, "A pipeline for multiparty volumetric video conferencing: transmission of point clouds over low latency DASH," in *Proc. ACM Multimedia Syst. Conf.*, May 2020.

[11] Z. Liu, J. Li, X. Chen, C. Wu, S. Ishihara, Y. Ji, and J. Li, "Fuzzy logic-based adaptive point cloud video streaming," *IEEE Open J. Comput. Soc.*, vol. 1, pp. 121–130, 2020.

[12] Y. Alkhalili, T. Gruczyk, T. Meuser, A. F. Anta, A. Khalil, and A. Mauthe, "Content-aware adaptive point cloud delivery," in *IEEE Int. Conf. Multimedia Big Data*, Dec. 2022, pp. 13–20.

[13] M. Rudolph and A. Rizk, "View-adaptive streaming of point cloud scenes through combined decomposition and video-based coding," in *Proc. ACM Int. Workshop Adv. Point Cloud Compression, Processing and Analysis*, Oct. 2022, pp. 41–49.

[14] L. Wang, C. Li, W. Dai, S. Li, J. Zou, and H. Xiong, "QoE-driven adaptive streaming for point clouds," *IEEE Trans. Multimedia*, vol. 25, pp. 2543–2558, 2023.

[15] 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. M. Tourapis, and V. Zakharchenko, "Emerging MPEG standards for point cloud compression," *IEEE J. Emerg. Sel. Top. Circuits Syst.*, vol. 9, no. 1, pp. 133–148, 2019.

[16] D. Graziosi, O. Nakagami, S. Kuma, A. Zaghetto, T. Suzuki, and A. Tabatabai, "An overview of ongoing point cloud compression standardization activities: video-based (V-PCC) and geometry-based (G-PCC)," *APSIPA Trans. Signal and Inf. Process.*, vol. 9, 2020.

[17] W. Zou, S. Zhang, and M. Yang, F.and Preda, "Standardization Status of MPEG Video-based Dynamic Mesh Coding (V-DMC)," in *Proc. IEEE ICASSP*, Apr. 2025, pp. 1–5.

[18] D. Meagher, "Geometric modeling using octree encoding," *Comput. Vision. Graph.*, vol. 19, no. 2, pp. 129–147, Jun 1982.

[19] R. L. de Queiroz and P. A. Chou, "Compression of 3D point clouds using a region-adaptive hierarchical transform," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3947–3956, aug 2016.

[20] K. Mammou, J. Kim, A. M. Tourapis, D. Podborski, and D. Flynn, "Video and subdivision based mesh coding," in *Proc. IEEE EUVIP*, Sept. 2022.

[21] M. Lounsbery, T. D. DeRose, and J. Warren, "Multiresolution analysis for surfaces of arbitrary topological type," *ACM Trans. Graph.*, vol. 16, no. 1, pp. 34–73, Jan. 1997.

[22] J.-E. Marvie, Y. Nehmé, D. Graziosi, and G. Lavoué, "Crafting the MPEG metrics for objective and perceptual quality assessment of volumetric videos," *Springer Qual. User Exp.*, vol. 8, no. 1, June 2023.

[23] J. Park, P. A. Chou, and J.-N. Hwang, "Volumetric media streaming for augmented reality," in *IEEE Glob. Commun. Conf.*, Dec. 2018, pp. 1–6.

[24] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, 1998.

[25] MPEG 3DGH, "Common test conditions for G-PCC," output doc w24178/N944, ISO/IEC JTC 1/SC 29/WG 7, Sapporo, JP, July 2024.

[26] MPEG 3DGH, "Common test conditions for V-DMC," output doc w24200/N964, ISO/IEC JTC 1/SC 29/WG 7, Sapporo, JP, July 2024.

[27] T. M. Borges, D. C. Garcia, and R. L. de Queiroz, "Fractional super-resolution of voxelized point clouds," *IEEE Trans. Image Process.*, vol. 31, pp. 1380–1390, 2022.

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