SRBF-Gaussian: Streaming-Optimized 3D Gaussian Splatting

Dayou Zhang[1,2] Graduate Student Member, IEEE, Zhicheng Liang[1,2] Graduate Student Member, IEEE,
Zijian Cao[1,2], Dan Wang[4] Senior Member, IEEE, Fangxin Wang[2,1,3*] Member, IEEE

[1]Shenzhen Future Network of Intelligence Institute
[2]School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen
[3]The Guangdong Provincial Key Laboratory of Future Networks of Intelligence
[4]Department of Computing, The Hong Kong Polytechnic University

Abstract

3D Gaussian Splatting (3DGS) has emerged as a groundbreaking 3D scene representation technique, offering unprecedented visual quality and rendering efficiency. However, the substantial data volume of 3DGS scenes poses significant challenges for streaming applications. Existing research on 3DGS has primarily focused on compression and rendering efficiency, neglecting the specific requirements of streaming transmission. Moreover, the Spherical Harmonics color representation in 3DGS complicates viewport-based transmission partitioning. Achieving hierarchical Gaussian streaming without noticeable quality degradation also remains a significant challenge.

To address these challenges, we propose SRBF-Gaussian, a new paradigm that revolutionizes the traditional 3DGS format. Our approach introduces viewport-dependent color encoding based on Spherical Radial Basis Functions (SRBFs) and HSL color space, enabling selective transmission of viewport-relevant color data. This reduces data transmission while maintaining visual quality. We implement adaptive Gaussian pruning and transmission, optimized for current viewports and network conditions. Additionally, we develop coherent multi-level Gaussian representations for smooth transitions between quality levels. Our system incorporates user-behavior-aware streaming strategies to anticipate and pre-fetch relevant data. In cloud VR scenarios, our approach demonstrates substantial improvements, achieving a 5.63% - 14.17% increase in PSNR, a 7.61% - 59.16% reduction in latency, and a 10.45% - 30.12% improvement in overall Quality of Experience (QoE).

Index Terms: 3D Gaussian Splatting, 3D Scene Transmission, Quality of Experience, Bitrate Adaptation.

1 Introduction

The rapid advancement of 3D reconstruction and rendering techniques has ushered in a new era of immersive virtual experiences. Among these techniques, 3D Gaussian Splatting (3DGS) [27] has emerged as a groundbreaking approach, offering unprecedented visual quality and rendering efficiency for complex 3D scenes. 3DGS represents 3D scenes as a collection of 3D Gaussians, each with its own position, scale, rotation, and appearance attributes, enabling high-fidelity reconstruction and efficient rendering of complex environments. **However, the substantial data volume of 3DGS scenes poses significant challenges for streaming applications.** A typical courtyard-sized scene in 3DGS format can demand 700MB to 1.5GB of data, far exceeding the capacity of current network infrastructures for real-time transmission, especially in the context of immersive, interactive experiences.

Table 1: Supported Features of Previous 3DGS-based Research.

	Viewport-based Gaussian Pruning	Viewport-based Attribute Pruning	Differential Transmission
CVPR24 Compress-3D [45]	×	×	×
ECCV24 SOGG [41]	×	×	×
CVPR24 Compact-3D [33]	✓	×	×
CVPR24 Scaffold-GS [37]	✓	×	×
Ours	✓	✓	✓

Existing research on 3DGS has primarily focused on compression techniques and rendering efficiency, neglecting the specific requirements of streaming media transmission. As illustrated in Tab.1, recent approaches [45, 41, 33, 37] do not adequately address key streaming challenges. These methods lack support for viewport-dependent attribute pruning (selective transmission of view-relevant color attributes) and Gaussian pruning (transmitting only view-relevant Gaussian ellipsoids), as well as differential transmission that adapts content quality based on network bandwidth conditions - all crucial features for adaptive streaming. In dynamic network scenarios, current methods require full scene retransmission to improve quality when bandwidth increases, rather than supporting incremental updates, which significantly impacts both bandwidth efficiency and user experience.

Moreover, applying traditional streaming techniques [51, 60, 25, 63, 24, 34, 59] to 3DGS presents unique challenges due to its inherent structure. The use of Spherical Harmonics (SH) for color representation, while effective for comprehensive scene visualization, is inefficient for viewport-dependent streaming contexts [14]. This makes it difficult to implement partial streaming based on the viewer's perspective. Additionally, attempts at incremental streaming of multi-level Gaussians often result in noticeable visual quality degradation, manifesting as object loss rather than resolution reduction [49]. These mismatches between 3DGS and conventional streaming methods call for a tailored 3DGS streaming approach with guaranteed visual quality and accurate network condition adaptation.

To address these challenges, we introduce *SRBF-Gaussian*, a novel streaming-optimized framework built on the foundations of 3D Gaussian Splatting. Our work tackles two critical issues in immersive content streaming: (1) We propose a viewport-dependent color representation scheme that replaces traditional Spherical Harmonics with a hybrid model utilizing the HSL color space [48] and Spherical Radial Basis Functions (SRBF) [22]. This innovation enables efficient partial streaming of color information based on the user's viewport. We extract base colors using HSL and employ an SRBF-based neural network to approximate viewport-dependent lighting efficiently, significantly reducing bandwidth requirements while maintaining visual fidelity. (2) Moreover, we present a multi-

- * *corresponding author.*

bitrate streaming strategy that enables the reuse of low-bitrate scene representations in high-bitrate scenarios. Our approach involves training with a subset of Gaussian ellipsoids fixed across different quality levels, allowing for incremental quality improvements. This method facilitates smooth transitions between quality levels and efficient adaptation to varying bandwidth conditions, enhancing overall streaming performance and visual consistency. Our contributions include:

- We conduct a comprehensive analysis of the inefficiencies inherent in using Spherical Harmonics for color representation in streaming 3DGS scenarios, providing insights into the limitations of current approaches. (§ 3)
- We develop a novel color extraction and representation method using the HSL color space, coupled with an SRBF-based approximation, enabling efficient, viewport-dependent color expression for streaming applications.(§ 5)
- We design and implement a multi-bitrate streaming framework that allows for seamless quality transitions and efficient scene reuse across different bandwidth scenarios, addressing key challenges in adaptive 3D content delivery. (§ 6)
- We perform an extensive evaluation of our proposed framework, demonstrating significant improvements in streaming efficiency, visual quality, and adaptation to varying network conditions compared to state-of-the-art methods.(§ 7)

In conclusion, our research addresses a significant gap in the application of 3DGS for cloud-based streaming services. *SRBF-Gaussian* have substantially enhanced the efficiency of streaming 3DGS, thereby facilitating more accessible and immersive virtual experiences across a broader spectrum of network conditions. This advancement may contribute to various domains, including virtual reality applications, remote collaboration platforms, and interactive entertainment systems.

2 Related Work

2.1 3D Gaussian Splatting

3D Gaussian Splatting (3DGS), an emerging technique [27], integrates the strengths of both implicit [40] and explicit radiance fields [56]. It employs 3D Gaussians as a flexible and efficient representation to accurately depict the scene. As shown in Fig. 1, 3DGS excels in high-quality rendering and real-time performance, particularly for complex scenes and high-resolution outputs.

At the core of 3DGS is the composition of numerous Gaussian ellipsoids, each possessing unique attributes that contribute to the overall scene representation. These ellipsoids serve as the foundational elements, characterized by position, rotation, scale, opacity, and color. Position determines the splat's location in 3D space, while rotation and scale define its orientation and size. Opacity controls transparency, facilitating the blending of overlapping elements, while color determines visual appearance. Together, these attributes enable 3DGS to intricately capture the complexity and fidelity of real-world scenes, making it an influential tool for rendering interactive visuals of exceptional quality.

2.2 3D Scene Streaming

In the past two decades, significant progress has been made in the development of immersive telepresence systems. However, several challenges still need to be addressed. These challenges primarily involve efficient streaming and management of reconstructed models, as well as the use of high-quality visualization through AR and VR equipment. Earlier approaches were limited by the available hardware [26, 30]. Thankfully, recent advancements in streaming and visualization technology have resulted in impressive immersive AR/VR-based live telepresence experiences.

Live 3D streaming has been successfully achieved in small-scale scenarios, such as teleconferencing [8, 12] and collaborative environments [38, 65]. These scenarios often involve expensive multi-camera static and pre-calibrated systems [17, 11, 32]. Additionally, for larger scenarios beyond a few square meters, live telepresence has been accomplished using low-cost and lightweight incremental scene capture with a moving depth camera [42, 54]. This approach allows remote users to immerse themselves in a live-captured environment, regardless of the specific sensor configurations. Bandwidth requirements, which were previously impractical, have been overcome by recent methods that enable group-scale sharing of telepresence experiences in live-captured environments while handling network interruptions [53, 54]. However, the practical sharing of live-captured 3D experiences in large-scale environments [55, 58], particularly for multiple users with low-cost setups, remains an ongoing challenge.

2.3 Color Space and Intrinsic Image Decomposition

The representation and manipulation of color have been fundamental to computer graphics and image processing. Over the years, various color spaces have been developed to address different needs and applications. The CIERGB color space [57], introduced in 1931, was one of the first standardized color spaces. However, its limitations in perceptual uniformity led to the development of more advanced color spaces. The YUV color space [47], widely used in video encoding, separates luminance (Y) from chrominance (UV) components, allowing for efficient compression. The LAB color space [23], designed to be perceptually uniform, has found extensive use in color management systems. Meanwhile, the HSV (Hue, Saturation, Value) and HSL (Hue, Saturation, Lightness) color spaces [50] offer intuitive color manipulation by separating color information from intensity.

While color space transformations can help separate different visual components, the complete separation of illumination and material properties presents an even more challenging problem known as intrinsic image decomposition [39]. This fundamental computer vision challenge aims to separate an observed image into its reflectance and shading components [5]. Land and McCann's Retinex theory [31] provided an early computational framework, while Horn [20] advanced this by introducing priors about edge characteristics. Recent works have further developed these concepts, with Grosse et al. [18] introducing sophisticated decomposition techniques, and Bell et al. [6] extending these approaches to complex real-world scenes. The field has seen significant progress through both traditional approaches using carefully designed priors [1, 2, 3] and modern deep learning methods [43, 66, 67].

In our work, we take a fundamentally different approach as the 3DGS scenes we deal with have fixed illumination and reflectance conditions. This means we do not need to employ complex intrinsic image decomposition techniques to separate and recalculate shading and reflectance components. Instead, our goal is simply to utilize a color space that better isolates illumination-induced color variations from inherent object colors. By fixing the base color components and only transmitting the necessary components for viewport-dependent variations, we can significantly reduce the transmission data volume while maintaining visual quality.

3 Motivation: Rethinking 3D Gaussian Splatting for Streaming Applications

The rapid advancement of 3D Gaussian Splatting (3DGS) has revolutionized neural rendering, offering unprecedented levels of visual fidelity and rendering efficiency. However, as we push the boundaries of immersive experiences and real-time rendering, particularly in the context of VR cloud applications and cloud-based 3D scenes, we encounter fundamental challenges that necessitate a critical reevaluation of the current 3DGS paradigm. This section explores

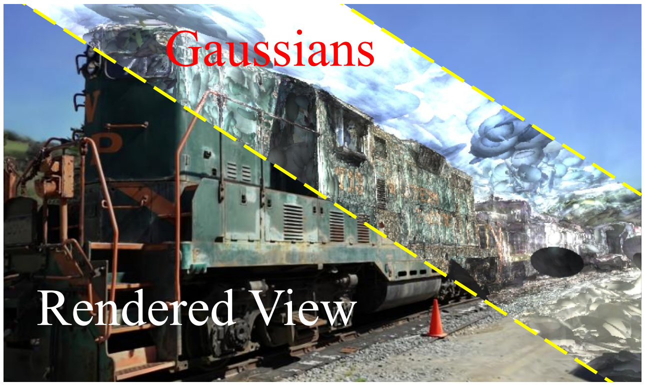

Figure 1: 3DGS reconstructed scene.

two pivotal questions in adapting 3DGS for streaming scenarios:

- Can the current 3DGS representation efficiently utilize bandwidth in cloud VR and similar streaming scenarios?
- Does the existing 3DGS framework effectively handle quality transitions under fluctuating network conditions in streaming environments?

Our investigation reveals that while 3DGS offers impressive capabilities, its current form is not optimized for these streaming scenarios. However, our measurements indicate significant untapped potential for improvement.

3.1 Huge Potential for Gaussian and Attribute Pruning

Our analysis of the Tank&Temple dataset [29] reveals a substantial opportunity for optimization in 3DGS streaming:

Necessity for Field of View (FoV) Based Gaussian Pruning: The three-dimensional structure of scenes creates complex occlusion patterns between different objects and even between various surfaces of the same object. These occlusion relationships directly impact the visibility of Gaussians in the spatial representation. For instance, in Fig. 1, the locomotive not only occludes its own rear surface but also obscures the ground and mountains behind it. From this particular viewpoint, only 33.58% of the Gaussians contribute to the final rendering. More broadly, our analysis of the Tank&Temple dataset reveals that, on average, a mere 20.21% of Gaussians are visible from any given training viewport. Consequently, there exists substantial potential for bandwidth reduction through view-dependent Gaussian pruning.

The Paradox of Spherical Harmonics and Viewport-Dependent Streaming: The current use of Spherical Harmonics (SH) for color representation in 3DGS, as shown in Fig. 2, while mathematically elegant, leads to significant inefficiencies in streaming contexts. SH requires the transmission of 48 parameters to represent the full spherical color distribution for each Gaussian, regardless of the actual visible portion.

To quantify this phenomenon, we conducted a statistical analysis of Gaussian rendering angles using spherical surface area measurements. Our methodology involved approximating each Gaussian's sphere of influence using an icosahedron (a regular polyhedron with 20 faces). We further refined this approximation by subdividing each triangular face of the icosahedron into four smaller triangles, resulting in a more granular representation of 80 triangular regions. We considering a triangular region fully visible if it was observed from any part of the triangular. To ensure comprehensive coverage, we rendered the scene from a diverse array of angles, accumulating visibility data across multiple viewpoints.

The resultant Cumulative Distribution Function (CDF), as illustrated in Fig. 3, reveals that 39.4% of Gaussians are visible from less than 41.25% of all angles, and 80.4% of Gaussians are visible from less than 60% of all angles. This finding underscores a significant potential for view-dependent attribute pruning, as it demonstrates that a substantial portion of Gaussians have limited visibility across different viewing angles.

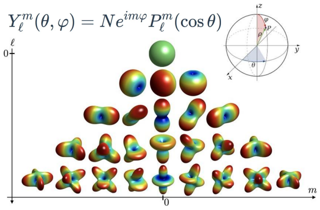
$Y_\ell^m(\theta,\varphi) = Ne^{im\varphi}P_\ell^m(\cos\theta)$
ℓ
0
m
0
z
y
x
φ
ρ
P
θ

Figure 2: Visualization of spherical harmonics basis functions.

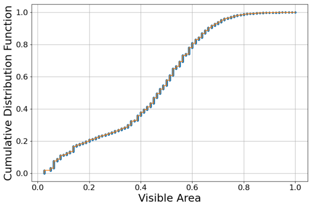

Figure 3: Cumulative Distribution Function of the visible portion of Gaussians in 3DGS.

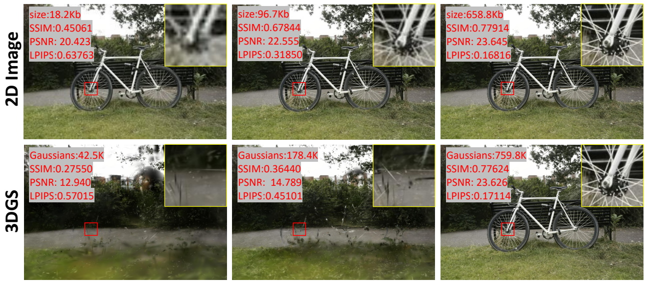

Figure 4: Comparison between traditional image downsampling and LOD-based gaussians downsampling in 3DGS.

This mismatch between SH's unified representation and the localized nature of user perception in interactive environments leads to unnecessary data transmission. In cloud-based streaming scenarios, where bandwidth optimization is crucial, the transmission of complete SH representations for each Gaussian becomes not just inefficient but potentially detrimental to the user experience, particularly in scenarios requiring low-latency interactions.

3.2 The Dilemma of Hierarchical Streaming in 3DGS

Traditional approaches to streaming optimization, such as downsampling [16, 21, 46] in point cloud streaming or scalabel encoding [44, 7] in video streaming, face unique challenges when applied to 3DGS. We illustrate this point with the following experiments.

Quality Degradation in Tiered Gaussian Transmission: We replicated and analyzed the LOD-based hierarchical Gaussian classification methods presented in Octree-GS [49], conducting extensive measurements on the results. Our findings reveal a non-linear relationship between Gaussian density and perceived visual quality, as shown in Fig. 4. Unlike traditional image downsampling, where quality degradation often occurs gradually, the reduction in Gaussian density can lead to abrupt and severe losses in both visual coherence and semantic integrity of the scene. Specifically, when the number of Gaussians is reduced to 23% of the original, critical scene elements such as the bicycle in the yard become virtually indiscernible, with the PSNR [19] dropping to 14.78. In contrast, a 2D image compressed to 14.7% of its original size maintains clear object representation and a PSNR of 22.55. This stark difference underscores the challenges in downsampling 3DGS representations compared to traditional 2D image compression techniques.

Incompatibility in Multi-Resolution Gaussian Scenes: Our investigations further reveal a critical limitation in combining 3DGS representations of varying resolutions or quality levels. As illustrated in Fig. 5, the naive combination of sparse and dense Gaussian

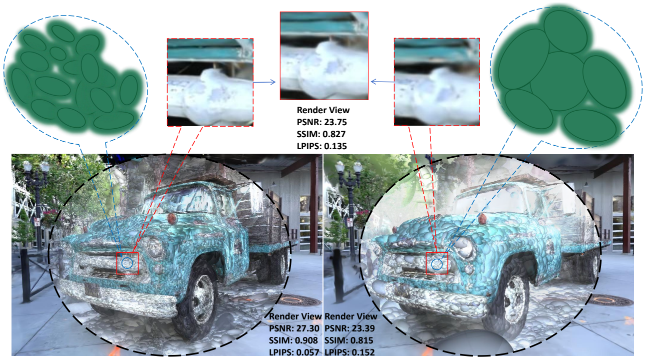

Figure 5: Illustration of the challenges in merging Gaussians of different densities, demonstrating that lower-quality Gaussians are not simply subsets of higher-quality representations.

distributions often results in visual artifacts and quality degradation rather than enhancement.

This incompatibility stems from the complex spatial relationships between Gaussians of different densities. Gaussians at identical spatial coordinates in scenes of different densities frequently overlap and contradict each other. There is no one-to-one correspondence between Gaussians in low and high-density representations, making it challenging to seamlessly integrate or transition between different quality levels.

3.3 Towards a New Paradigm in 3DGS Streaming

Our analysis reveals that while 3DGS offers impressive rendering capabilities, its current form is not optimized for streaming applications, particularly in cloud-based VR and interactive 3D environments. The challenges identified the urgently need of a new paradigm of 3DGS that is inherently designed for streaming and interactive applications.

This paradigm shift encompasses several key aspects: (1) Viewport-dependent color encoding that moves beyond SH to more flexible, directionally-aware color representations; (2) Adaptive Gaussian pruning and transmission methods to dynamically determine and prioritize perceptually significant Gaussians; (3) Coherent multi-resolution representations that maintain visual and semantic coherence across different levels of detail; and (4) User-behavior-aware streaming strategies that incorporate prediction and scene understanding to anticipate and pre-fetch relevant Gaussian data.

By addressing these challenges, we aim to unlock the full potential of 3DGS in cloud-based streaming applications, paving the way for more immersive, responsive, and efficient virtual reality experiences.

4 System Overview

Fig. 6 illustrates the workflow of our proposed framework. Initially, ① Reusable Multi-Quality Scene Training generates multi-bitrate 3DGS scenes by creating hierarchical representations with varying numbers of Gaussian primitives, enabling adaptive streaming. Subsequently, ② SRBF Color Expression transforms the traditional color representation into an SRBF-based HSL representation, facilitating divisible encoding and efficient transmission. Following this, ③ Occlusion Culling Preprocessing computes visibility information through occlusion culling, significantly reducing data transmission requirements. During real-time streaming, ④ Differential Transmission dynamically selects bitrate based on prevailing network conditions. Broadly speaking, our system comprises two principal components:

SRBF-based (Spherical Radial Basis Function) Color Expression (§5) transforms the traditional Spherical Harmonic-based RGB color representation into an SRBF-based HSL representation. This approach utilizes OMP algorithm to determine the optimal SRBF parameters, enabling efficient encoding and transmission of color information. The SRBF module significantly reduces the data volume while preserving viewport-dependent color effects, crucial for high-quality rendering in bandwidth-constrained environments.

Differentiable streaming-optimized 3DGS Framework (§6) addresses the challenges of real-time 3DGS delivery in dynamic network conditions. It incorporates a multi-bitrate generation mechanism that creates hierarchical representations of the scene with varying numbers of Gaussian primitives. This component works with an adaptive bitrate selection algorithm to ensure smooth visual experiences during navigation while reducing network throughput. The framework also leverages precomputed occlusion information to further enhance transmission and rendering efficiency.

5 SRBF Color Expression

5.1 Separating Color and Lighting using HSL Color Space

In our *SRBF-Gaussian* framework, we propose a novel approach: the intrinsic color of objects can be separated from lighting effects. This separation not only simplifies color representation but also enhances streaming efficiency. However, we recognize that employing complex lighting calculation methods (such as global illumination or ray tracing) could significantly reduce rendering speed. Consequently, we prioritized a method that separates the expression, storage, and transmission of color and lighting effects, opting for the HSL color space to achieve this objective.

The HSL (Hue, Saturation, Lightness) color space demonstrates superior performance in separating color and lighting effects. As illustrated in Fig. 7, the HSL space utilizes three components to represent color: Hue (H) denotes the dominant wavelength, Saturation (S) represents color intensity, and Lightness (L) indicates brightness. This representation enables us to encode fundamental color information in the hue component while employing saturation and lightness to express variations in lighting conditions.

We first convert colors from RGB to the HSL space [50]. Then, we define a base color for each Gaussian ellipsoid:

$$C_{base} = (H, S_{ref}, L_{ref}) \tag{1}$$

where H is the hue, and S_{ref} and L_{ref} are reference saturation and lightness values under neutral lighting conditions.

To implement viewport-dependent lighting effects, we introduce a lighting function $L(\theta,\phi)$ that modulates the saturation and lightness components based on viewing angles:

$$C_{final}(\theta,\phi) = (H, S_{ref} \cdot L_s(\theta,\phi), L_{ref} \cdot L_l(\theta,\phi)) \tag{2}$$

where L_s and L_l are lighting functions for saturation and lightness, respectively.

This methodology enables efficient transmission of base color information and computation of lighting effects based on the current viewport, thereby optimizing bandwidth utilization while maintaining visual quality. The HSL-based color representation effectively replicates the expressive power of Spherical Harmonics in RGB space, while improving streaming efficiency by transmitting only the necessary color and lighting information for the current viewpoint.

5.2 SRBF Decomposition using Orthogonal Matching Pursuit

To efficiently represent and compute the viewport-dependent lighting functions L_s and L_v introduced in the previous section, we propose using Spherical Radial Basis Functions (SRBF) with Orthogonal Matching Pursuit (OMP) algorithm. This method provides an

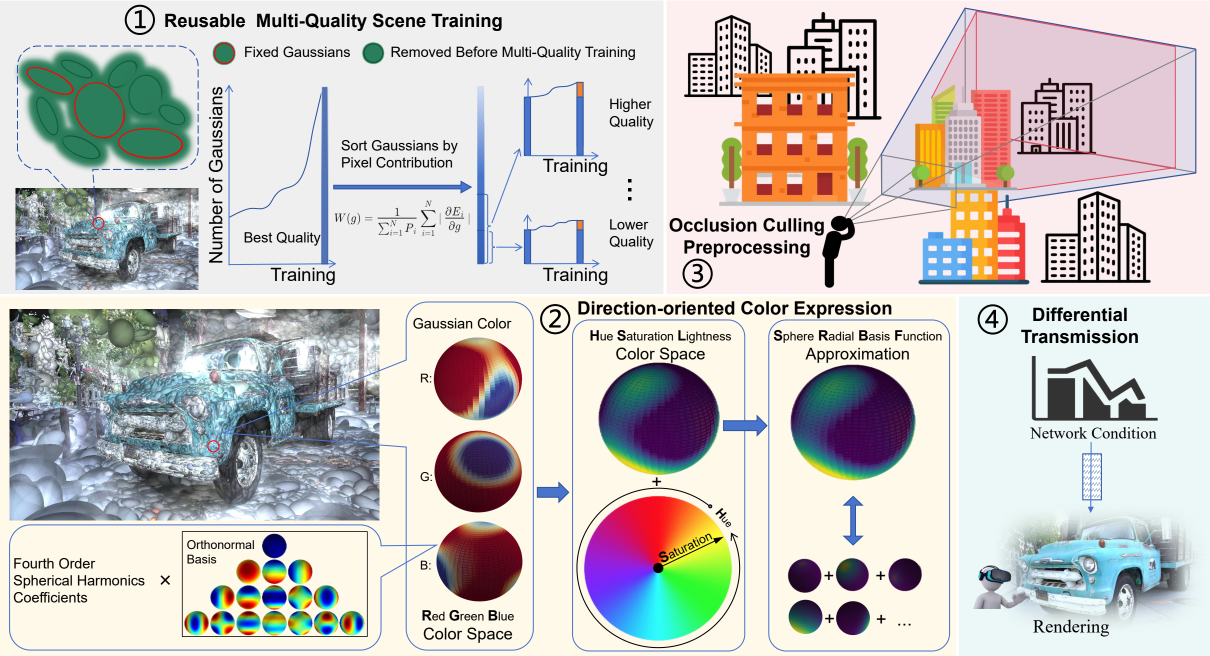
① Reusable Multi-Quality Scene Training
Fixed Gaussians
Removed Before Multi-Quality Training
Number of Gaussians
Best Quality
Training
Sort Gaussians by Pixel Contribution
$W(g)=\frac{1}{\sum_{i=1}^{N}P_i}\sum_{i=1}^{N}\left|\frac{\partial E_i}{\partial g}\right|$
Higher Quality
Training
Lower Quality
Training
Occlusion Culling Preprocessing ③
② Direction-oriented Color Expression
Gaussian Color
R:
G:
B:
Red Green Blue Color Space
Fourth Order Spherical Harmonics Coefficients ×
Orthonormal Basis
Hue Saturation Lightness Color Space
Saturation
Hue
Sphere Radial Basis Function Approximation
④ Differential Transmission
Network Condition
Rendering

Figure 6: Overview of SRBF-Gaussian.

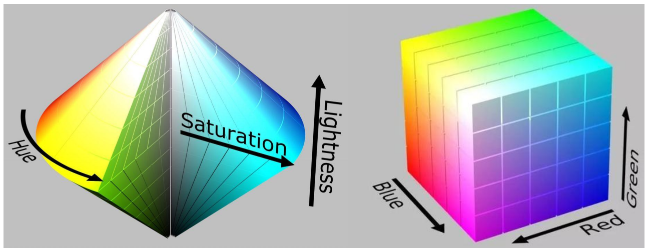

Figure 7: Comparison between HSL and RGB Color Spaces. [10]

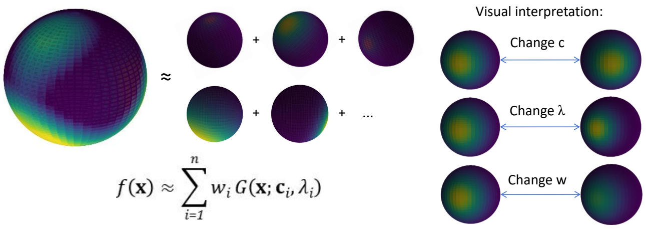
≈
+
+
+
+ ...
$f(\mathbf{x})\approx\sum_{i=1}^{n}w_i\,G(\mathbf{x};\mathbf{c}_i,\lambda_i)$
Visual interpretation:
Change c
Change λ
Change w

Figure 8: Visualization of spherical function approximation using spherical basis functions.

efficient sparse approximation of complex lighting conditions while maintaining high accuracy.

5.2.1 SRBF Definition

Spherical Radial Basis Functions (SRBFs) are a class of functions defined on the surface of a sphere, making them ideal for representing spherical data such as directional lighting. It is defined as:

$$G(\mathbf{x};\mathbf{c},\lambda)=\exp(\lambda(\mathbf{x}\cdot\mathbf{c}-1)) \tag{3}$$

where $\mathbf{x}$ is a unit vector on the sphere, $\mathbf{c}$ is the center of the SRBF (also a unit vector), and $\lambda > 0$ is the shape parameter controlling the width of the function.

A linear combination of SRBFs can approximate any smooth function on the sphere. For a given function $f(\mathbf{x})$ on the unit sphere, we can approximate it as:

$$f(\mathbf{x})\approx\sum_{i=1}^{n}w_iG(\mathbf{x};\mathbf{c}_i,\lambda_i) \tag{4}$$

where w_i are the weights, and n is the number of SRBFs used.

The theoretical foundation and practical effectiveness of SRBFs have been extensively studied in various domains. Research has demonstrated that SRBFs can achieve approximation convergence for smooth functions on the sphere [36, 35], with applications ranging from scattered data approximation to computer graphics [9, 13].

5.2.2 OMP-based SRBF Decomposition

We employ the Orthogonal Matching Pursuit algorithm to decompose our lighting functions into a sparse combination of SRBFs. As illustrated in Fig. 8, this approach iteratively selects the most relevant SRBF components to approximate the target function.

The decomposition process can be formalized as:

$$L(\theta,\phi)=\sum_{i=1}^{K}w_iG(\mathbf{x}(\theta,\phi);\mathbf{c}_i,\lambda_i) \tag{5}$$

where K is the number of selected SRBF components, determined by the desired approximation accuracy or sparsity constraints. The $L(\theta,\phi)$ here represents either L_s or L_l from the previous section, allowing us to decompose both saturation and lightness modulation functions independently.

The OMP algorithm iteratively constructs the SRBF approximation through a systematic process. Initially, the algorithm begins with the target lighting function as the residual. In each iteration, it identifies the SRBF component that exhibits the strongest correlation with the current residual signal. Upon selecting a new component, the algorithm performs a least-squares optimization to update all weights of the previously selected components, ensuring optimal approximation at each step. The residual is then recomputed by subtracting the current approximation from the target function.

This process continues iteratively until either a predetermined approximation accuracy threshold is achieved or a specified maximum number of components is reached.

This iterative selection and optimization strategy ensures that each additional SRBF component maximally reduces the approximation error, resulting in a compact yet accurate representation of the lighting function. The orthogonality property of OMP guarantees that the algorithm efficiently captures distinct features of the lighting function while avoiding redundant representations.

6 Streaming-optimized 3DGS Framework

6.1 Reusable Multi-Quality Scene Training

Inspired by scalable encoding techniques [44, 51, 7], we introduce a differential multi-bitrate training method that enables high-bitrate versions to efficiently reuse a substantial portion of low-bitrate Gaussians. This approach significantly reduces data redundancy and transmission overhead while maintaining visual quality across different bitrates.

Our approach begins with training a high-fidelity 3DGS model using conventional methods. Subsequently, we quantify each Gaussian's contribution to the rendered image across all training viewpoints using the following formula:

$$W(g) = \frac{1}{\sum_{i=1}^{N} P_i} \sum_{i=1}^{N} |\frac{\partial E_i}{\partial g}| \tag{6}$$

where $W(g)$ represents the importance weight of Gaussian g, N is the number of training views, P_i is the number of pixels in view i, and E_i is the rendering error for view i.

Based on these importance weights, we extract hierarchical subsets of the most significant Gaussians (e.g., 25%, and 50% in our experiments). We then freeze these subsets and adjust the 3DGS training process for each bitrate level. This involves increasing the threshold for Gaussian cloning, changing split operations to cloning operations, and restricting clone operations to the top 1 percent of total Gaussians for each cloning operation.

This modified training process yields a multi-bitrate 3DGS scene representation that supports differential transmission. Our method ensures that higher bitrate versions can effectively reuse Gaussians from lower bitrate versions, significantly reducing data redundancy and transmission overhead.

6.2 Occlusion Culling Preprocessing

To optimize data transmission in 3DGS, we introduce an occlusion-aware Gaussian culling [15] preprocessing pipeline. This approach significantly reduces the number of Gaussians and their attributes that need to be rendered and transmitted, thereby improving both performance and bandwidth efficiency.

Our culling algorithm operates in two phases: visibility determination and SRBF activation. In the visibility determination phase, we discretize the navigable space within the scene into a 3D grid. For each grid cell, we compute the set of potentially visible Gaussians using a hybrid approach that combines a hierarchical Z-buffer technique optimized for 3DGS and a modified version of the Hidden Point Removal operator adapted for Gaussian primitives. This process accounts for both self-occlusion within objects and inter-object occlusion, providing a comprehensive visibility solution for complex 3DGS scenes.

In the SRBF activation phase, we determine which Spherical Radial Basis Functions (SRBFs) need to be activated to accurately represent the viewport-dependent lighting effects for each visible Gaussian from a given viewpoint. Formally, for a given viewpoint v and Gaussian g, we define the set of active SRBFs as:

$$A(v,g) = s_i | \text{contrib}(s_i, v, g) > \varepsilon \tag{7}$$

where $\text{contrib}(s_i, v, g)$ measures the contribution of SRBF s_i to the appearance of Gaussian g from viewpoint v. The adaptive threshold ε is determined through a perceptual study: we conducted experiments with participants using a staircase procedure to find the just-noticeable difference threshold. Starting with $\varepsilon = 0.1$, participants compared original and approximated renderings, adjusting ε until differences became imperceptible (75% detection threshold). The final ε values typically range from 0.01 to 0.05, depending on the scene complexity and viewing conditions.

The result of this preprocessing step is a compact, queryable data structure that maps each grid cell to its corresponding set of visible Gaussians and active SRBFs. Crucially, due to the separable nature of our SRBF approximation, we only need to transmit parameters for 1-4 SRBFs per Gaussian in most cases. Each SRBF is represented by its position (2D spherical coordinates), intensity, and decay rate, resulting in a highly compact representation.

6.3 Adaptive Bitrate Streaming

Our adaptive streaming strategy, built upon the multi-bitrate training method, optimizes content delivery based on network conditions and user behavior. This approach prioritizes visual quality within the user's FoV while ensuring smooth transitions during sudden movements or view changes. We categorize scene Gaussians into three primary groups: FoV Gaussians (within the current field of view), Peripheral Gaussians (visible from the current cell but outside the FoV), and Adjacent Gaussians (visible from neighboring cells). Each category is further divided into low, medium, and high bitrate levels.

The streaming algorithm dynamically selects Gaussians for transmission based on their category, bitrate level, and available bandwidth. Our transmission priority order balances immediate visual quality with preemptive loading. It begins with low-bitrate FoV Gaussians, progressively enhancing them to medium and high quality. This is followed by low-bitrate Peripheral Gaussians, then medium-bitrate FoV Gaussians, with subsequent transmission ordered by increasing distance from the viewer's current cell to the nearest cell where each Gaussian becomes visible.

This priority scheme ensures rapid population of the user's immediate view with at least low-quality Gaussians, followed by progressive enhancement. It also preloads lower quality versions of nearby content to prevent blank areas during sudden movements or view changes. The streaming process adapts to varying bandwidth conditions. Under high bandwidth, the algorithm aims to deliver the highest quality experience across all visible areas. With medium or low bandwidth, transmission focuses on lower and medium bitrate versions, adjusting the priority order accordingly.

7 Evaluation

7.1 Methodology

7.1.1 Implementation Details

We first construct a series of 3DGS models at different quality levels, and then decompose their color information into SRBF representations using the OMP algorithm. For Gaussian ellipsoids that appear in multiple quality levels, the SRBF decomposition is performed only once. The space is then divided into cells for computing view-dependent occlusion culling. Our framework is implemented in Python, with network conditions simulated using the Linux Traffic Control (tc) tool. We have modified the diff-gaussian-rasterization (specifically the forward.cu) to incorporate our SRBF-based color rendering process.

All training and testing procedures were executed on a high-performance computing system equipped with an Intel(R) Xeon(R) Silver 4210 CPU @ 2.20GHz and a single NVIDIA A100 GPU with 80GB of dedicated memory. *SRBF-Gaussian* renders scenes at 1080p resolution (1920×1080 pixels). While we don't impose frame rate restrictions on either the baselines or *SRBF-Gaussian*, all methods consistently maintain frame rates above 30 FPS. For visualization and evaluation, scenes are rendered on a standard computer display monitor rather than a VR headset.

7.1.2 Dataset

We conducted comprehensive evaluations using real-world scene datasets including Mip-NeRF360 [4], and Tanks&Temples [29]. By encompassing a diverse range of scenarios, these datasets enabled us to comprehensively demonstrate the effectiveness of our proposed approach. We applied various walking and viewing tracks for each scene, utilizing the participation of three volunteers.

7.1.3 Experiment Setting

In the experiment, participants were assumed to navigate within the 3D regions. Considering the general movement speed, we set a fixed duration of 30-seconds for each scene. After that, participants would proceed to load a new scene and engage in another 30-second navigation session. Participants were afforded 6DoF, enabling unrestricted movement and rotation of their viewpoint.

7.1.4 Evaluation Metric

Follows previous research [34, 52, 61, 62, 64], the Quality of Experience (QoE) is determined by perceptual quality and delay in our context and formulated as:

$$QoE = \sum_{t=1}^{T} (f(t) - \lambda g(t)) \tag{8}$$

where $f(t)$ measures the distortion between the reconstructed scene and the ground truth and $g(t)$ calculates the latency stems from client-side data transmission and decoding, and λ signifies the relative importance assigned to the delay component in the overall QoE calculation. In our experiment, we employ Peak Signal-to-Noise Ratio (PSNR) [19] as the metric for $f(t)$, while $g(t)$ is measured in seconds to represent latency. The value of λ is set to 5, and both PSNR and latency are normalized.

7.1.5 Baseline

To ascertain the performance superiority of *SRBF-Gaussian*, we consider four state-of-the-art baselines:

- **Compressed-3D** [45]: This method employs sensitivity-aware clustering, quantization-aware fine-tuning, and entropy encoding to reduce memory consumption and improve efficiency.
- **SOGG** [41]: SOGG introduces a parallel sorting algorithm to arrange high-dimensional Gaussian parameters into a 2D grid, effectively preserving neighborhood structure. It incorporates a smoothness loss to enforce local grid coherence for efficient JPEG XL compression.
- **Compact-3D** [33]: This method proposes a learnable mask strategy to reduce the number of Gaussians and employs residual vector quantization for compact geometry attribute representation, and applies quantization, pruning, and entropy coding to further optimize the model size.
- **Scaffold-GS** [37]: This approach utilizes anchor points initialized from Structure from Motion (SfM) to distribute local 3D Gaussians, predicting their attributes based on viewing direction and distance. It also employs pruning strategies for improved scene coverage to enhance rendering speed.

7.2 Performance Analysis

7.2.1 Overall Performance of of *SRBF-Gaussian*:

Fig. 9, 10, 11, and 12 illustrate the trends in QoE score, average PSNR, average latency, and network simulation bandwidth, respectively. These results demonstrate that *SRBF-Gaussian* significantly outperforms the other four baselines across various metrics.

In terms of overall QoE, *SRBF-Gaussian* achieves the highest score, showing improvements of 10.45%, 25.80%, 30.12%, and 16.69% compared to Compact-3D, Scaffold-GS, SOGG, and Compressed-3D, respectively. SOGG and Compressed-3D, due to their lack of support for progressive transmission, perform the poorest in this aspect. Compact-3D ranks second, followed by Scaffold-GS, attributed to Compact-3D's slightly better compression ratio.

Visual quality, as measured by PSNR, showcases *SRBF-Gaussian*'s superior performance, especially in the initial seconds of transmission. Our method demonstrates PSNR improvements of 7.44%, 12.47%, 14.17%, and 5.63% over Compact-3D, Scaffold-GS, SOGG, and Compressed-3D, respectively. Compact-3D and Scaffold-GS exhibit gradual PSNR improvements due to their support for partial transmission, with Compact-3D showing a slightly faster increase owing to its better compression ratio. *SRBF-Gaussian*, benefiting from multi-bitrate support and viewport-dependent optimization, achieves exceptional visual quality in the first few seconds, outperforming all baselines until converging with others later. SOGG and Compressed-3D, unable to support progressive transmission, only begin rendering after complete data transfer at the fourth and fifth second, respectively, resulting in zero PSNR before these points.

Regarding latency, *SRBF-Gaussian* demonstrates substantial reductions of 7.61%, 47.31%, 59.19%, and 30.11% compared to Compact-3D, Scaffold-GS, SOGG, and Compressed-3D, respectively. Our method achieves the lowest initial latency due to its multi-bitrate support and viewport-dependent optimization. Compact-3D follows, then Scaffold-GS, reflecting their compression efficiency. Both Compact-3D and Scaffold-GS eventually reach a minimal latency primarily consisting of rendering time, with Compact-3D slightly outperforming Scaffold-GS. SOGG and Compressed-3D, limited by their transmission approach, only start rendering after complete data transfer at the fourth and fifth second, respectively, with latency treated as a timeout (1 second) before completion. *SRBF-Gaussian* occasionally experiences slight transmission delays in later stages when higher bitrates are needed for specific viewpoints to enhance visual quality.

7.3 Component Analysis

7.3.1 Effectiveness of Gaussian and Attributes Pruning

The effectiveness of our pruning approach in reducing data transmission requirements is demonstrated through experimental evaluation. Fig. 13 shows the proportion of Gaussians and SRBFs retained after applying Occlusion Culling.

Our findings reveal a significant enhancement in data representation efficiency. As users navigate through the scene, the proportion of Gaussians classified as visible exhibits a natural increase, culminating in 86.85% by the conclusion of the exploration phase. This indicates that the position, rotation, opacity, and base color information of these Gaussians require transmission. Notably, despite the high percentage of visible Gaussians, merely 22.39% of SRBFs are requisite for accurate chromatic representation at the termination of the exploration. This marked disparity between the proportion of Gaussians (86.85%) and SRBFs (22.39%) requiring transmission underscores the exceptional effectiveness of our method.

The substantially lower percentage of SRBFs, both initially (3.58%) and at the end of exploration (22.39%), provides significant evidence that our SRBF-based approach successfully minimizes the transmission of viewport-dependent color data. This empirical result demonstrates that our innovative viewport-dependent color representation scheme significantly outperforms the traditional Spherical Harmonics color expression in the context of 3DGS.

Furthermore, our comparative analysis of throughput across different baselines, as shown in Fig. 14, reveals that our method achieves comparable throughput to compression-based approaches like Scaffold-GS. It is important to note that the throughput measurement here does not directly reflect compression efficiency, but

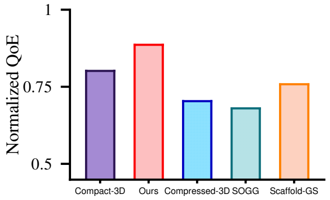

Figure 9: Performance comparison using overall QoE.

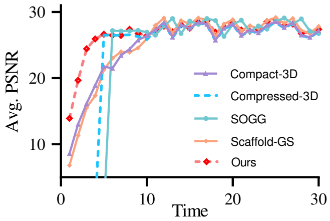

Figure 10: The Avg. value of PSNR over time.

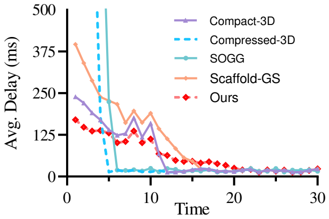

Figure 11: The Avg. latency of baselines over time.

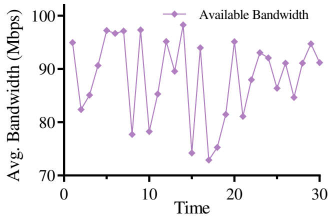

Figure 12: The Avg. Available bandwidth over time.

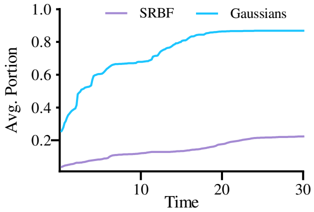

Figure 13: The Avg. Portion of Gaussians and SRBFs after Occlusion Culling.

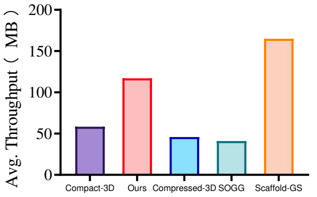

Figure 14: Comparison of Average Throughput across Different Methods.

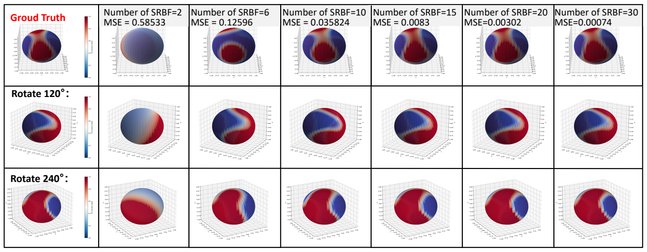

Figure 15: SRBF Approximation for Global Lightness: Illustration of spherical surface fitting using varying numbers of Spherical Radial Basis Functions (SRBFs). As the number of SRBFs increases, the approximation of the spherical surface improves. With over 20 SRBFs, the Mean Squared Error (MSE) reaches an acceptable level, demonstrating a good balance between accuracy and computational efficiency.

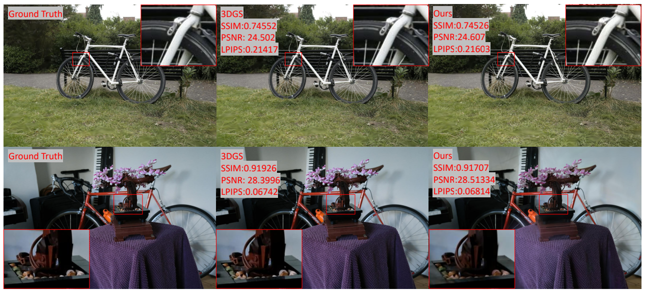

Figure 16: Qualitative comparisons between 3DGS and SRBF-based method on the Mip-NeRF 360 Dataset.

rather indicates how much data is actually transmitted based on the current viewport requirements.

7.3.2 Performance of SRBF-based Color Expression

Our novel approach utilizing Spherical Radial Basis Functions (SRBF) for light effect approximation demonstrates significant improvements over the original color representation in original 3DGS.

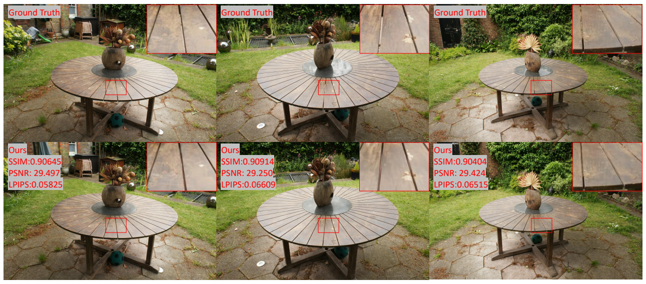

Figure 17: Visual effect of illumination. Different angle of the table shows the effect of the desktop reflecting sunlight.

Table 2: Quantitative results on Mip-NeRF 360 dataset.

Scene	Avg. SSIM ↑		Avg. PSNR ↑		Avg. LPIPS ↓	
	3DGS	Ours	3DGS	Ours	3DGS	Ours
bicycle	0.8287	0.8287	26.014	25.939	0.1472	0.1472
bonsai	0.9025	0.9051	30.543	30.886	0.0946	0.0891
counter	0.9106	0.9084	30.532	30.212	0.0919	0.0941
garden	0.9084	0.9055	30.289	29.970	0.0867	0.0900
kitchen	0.9182	0.9148	30.964	30.501	0.0782	0.0818
room	0.9228	0.9196	31.691	31.364	0.0775	0.0808
stump	0.9198	0.9167	31.576	31.268	0.0799	0.0832

Fig. 15 illustrates the approximation capability of our SRBF approximation. As the number of SRBFs on the sphere increases, the approximation accuracy improves substantially. Notably, with more than 20 SRBFs, we achieve a Mean Squared Error (MSE) below 0.003, indicating high-fidelity representation. It is worth noting that our evaluation encompasses the entire spherical surface approximation to showcase the robustness of our SRBF-based method. In practical 3D scene training scenarios, some Gaussians may be occluded in certain directions due to spatial relationships, which can further reduce the required number of SRBFs.

Fig.16 presents the visual quality achieved by our SRBF-based method on the Mip-NeRF 360 Dataset. The results demonstrate the method's ability to accurately capture and render complex scenes. Furthermore, Fig.17 provides a compelling example that highlights our method's capacity to fit illumination effects with high fidelity, showcasing its potential for realistic light modeling in diverse environments. Quantitative results for each scene in the Mip-NeRF 360 Dataset are presented in Table 2. Our SRBF-based approach achieves visual quality comparable to the original 3DGS across most scenes. Notably, in the "bonsai" scene, our method even outperforms the original 3DGS, underscoring its effectiveness in capturing the subtle lighting effects.

We evaluated different color spaces for color and lighting decomposition in our method. Table 3 compares our approach using YUV, HSV, and two variants of HSL color spaces against the original 3DGS method. While YUV and HSV color spaces offer some

Table 3: Quantitative Comparison of Different Color Space.

	SSIM ↑	PSNR ↑	LPIPS ↓
Original 3DGS	0.8679	25.7551	0.1104
Ours-HSV (Fixed H,S + SRBF-based V)	0.8588	24.6019	0.1195
Ours-YUV (Fixed U,V + SRBF-based Y)	0.8622	25.0568	0.1195
Ours-HSL (Fixed H,S + SRBF-based L)	0.8619	25.0958	0.1199
Ours-HSL (Fixed H + SRBF-based S,L)	0.8659	25.6043	0.1132

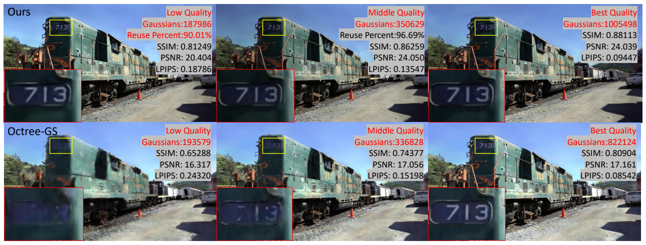

Figure 18: Qualitative performance of multi-bitrate training.

benefits in terms of color-lighting separation, they do not perform as well as either the original 3DGS or our HSL-based methods. Among all tested variants, the HSL color space, particularly with fixed H and SRBF-fitted SL, demonstrates superior performance across multiple metrics.

7.3.3 Performance of Multi-Quality Training

As demonstrated in Fig. 18, our approach offers significant advantages, particularly in the efficient reuse of Gaussians across various bitrate versions. In contrast to octree-based Gaussian Splatting (octree-GS) methods, which primarily focus on Level of Detail (LoD) layering for Gaussians and often result in a loss of scene details during low-bitrate rendering, our method successfully preserves more fine-grained information even at lower bitrates.

Both qualitative visual assessments and quantitative metrics (PSNR, SSIM [19], and LPIPS [28]) consistently show that our multi-bitrate approach surpasses the octree-GS method. This superior performance can be attributed to our method's ability to maintain critical scene details across diverse bitrate levels, thereby ensuring a more coherent and high-quality visual experience as the stream quality improves. Interestingly, we observed that the lower bitrate versions of our scenes tend to appear slightly brighter overall when rendered. This phenomenon likely stems from the freezing of a substantial portion of Gaussians and the limitation on cloning and splitting during the post-training process, which may result in reduced overlapping of Gaussians at lower bitrates.

8 Discussion

The *SRBF-Gaussian* framework represents a significant advancement in 3D Gaussian Splatting streaming technology, offering substantial improvements in transmission efficiency while maintaining high visual quality. The framework's innovative approach to viewport-dependent color encoding and adaptive streaming demonstrates considerable potential for applications in cloud-based virtual reality, remote collaboration, and interactive entertainment systems. This advancement is particularly significant given the growing demand for high-quality 3D content streaming in emerging metaverse applications and next-generation immersive telecommunications.

However, there are several limitations and opportunities for future work that should be acknowledged. First, the current implementation relies on OMP for SRBF optimization, which presents significant computational challenges due to its limited parallelization capabilities. While this preprocessing overhead does not directly impact the online user experience in video-on-demand scenarios, the substantial computational resources and time required for conversion could impede practical deployment at scale. Future research should explore more efficient optimization algorithms or parallel processing techniques to address this computational bottleneck.

Second, while the current approach leverages the same Hue of the HSL to reduce transmission requirements for view-dependent Gaussian colors, the HSL color space may not represent the optimal solution for this purpose. Future work could investigate custom color spaces specifically designed to decompose illumination and reflection-induced color variations, or explore alternative approaches such as intrinsic decomposition methods to further separate and optimize the transmission of lighting-related color changes.

The scalability and robustness of *SRBF-Gaussian* also warrant further investigation. While the experiments demonstrated promising results, they were primarily conducted using standard computer displays rather than virtual reality headsets. Future evaluations should include comprehensive testing with actual VR hardware to better understand the system's performance under real-world usage conditions and diverse network environments. This would provide more meaningful insights into the framework's practical effectiveness in immersive applications.

The personalization of the *SRBF-Gaussian* to individual users is another area for future exploration. The current implementation lacks sophisticated scene analysis and user intent prediction mechanisms. By incorporating these elements, along with advanced trajectory prediction and viewport direction estimation algorithms, the system could potentially achieve more precise bandwidth optimization and faster loading times. This user-centric approach could significantly enhance the overall streaming efficiency and QoE.

9 Conclusion

This study introduces *SRBF-Gaussian*, a novel framework for addressing challenges in 3DGS scene transmission and enhancing user experience. Our approach involves implementing viewport-dependent color encoding based on Spherical Radial Basis Functions and HSL color space, employing adaptive Gaussian pruning and transmission, and developing coherent multi-level Gaussian representations. Extensive experimental results in cloud VR scenarios demonstrate substantial improvements, including a higher reconstruction quality with a 5.63% - 14.17% increase in PSNR, a 7.61% - 59.16% reduction in latency, and a 10.45% - 30.12% enhancement in QoE compared to existing state-of-the-art solutions.

Acknowledgments

The work was supported in part by NSFC with Grant No. 62293482, the Basic Research Project No. HZQB-KCZYZ-2021067 of Hetao Shenzhen-HK S&T Cooperation Zone, NSFC with Grant No. 62471423, the Shenzhen Science and Technology Program with Grant No. JCYJ20241202124021028 and Grant No. JCYJ20230807114204010, the Shenzhen Outstanding Talents Training Fund 202002, the Guangdong Research Projects No. 2019CX01X104, the Young Elite Scientists Sponsorship Program of CAST (Grant No. 2022QNRC001), the Guangdong Provincial Key Laboratory of Future Networks of Intelligence (Grant No. 2022B1212010001) and the Shenzhen Key Laboratory of Big Data and Artificial Intelligence (Grant No. ZDSYS201707251409055). The work of Dan Wang was supported in part by RGC GRF 15200321, 15201322, 15230624, ITC ITF-ITS/056/22MX, and PolyU 1-CDKK, G-SAC8.

References

[1] J. T. Barron and J. Malik. Shape, albedo, and illumination from a single image of an unknown object. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 334–341. IEEE, 2012.

[2] J. T. Barron and J. Malik. Intrinsic scene properties from a single rgb-d image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 17–24, 2013.

[3] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE transactions on pattern analysis and machine intelligence*, 37(8):1670–1687, 2014.

[4] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5470–5479, 2022.

[5] H. Barrow, J. Tenenbaum, A. Hanson, and E. Riseman. Recovering intrinsic scene characteristics. *Computer Vision System*, 2(3-26):2, 1978.

[6] S. Bell, K. Bala, and N. Snavely. Intrinsic images in the wild. *ACM Transactions on Graphics*, 33(4):1–12, 2014.

[7] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian. Overview of shvc: Scalable extensions of the high efficiency video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(1):20–34, 2015.

[8] S. Cho, S.-w. Kim, J. Lee, J. Ahn, and J. Han. Effects of volumetric capture avatars on social presence in immersive virtual environments. In *Proceedings of the IEEE conference on virtual reality and 3D user interfaces*, pp. 26–34, 2020.

[9] G. Czakó, V. Szalay, and A. G. Császár. Finite basis representations with nondirect product basis functions having structure similar to that of spherical harmonics. *The Journal of chemical physics*, 124(1), 2006.

[10] Datumizer. Color solid comparison hsl hsv rgb cone sphere cube cylinder. https://commons.wikimedia.org/wiki/File:Color_solid_comparison_hsl_hsv_rgb_cone_sphere_cube_cylinder.png, 2008. Image from Wikimedia Commons, licensed under CC BY-SA 3.0.

[11] R. Du, M. Chuang, W. Chang, H. Hoppe, and A. Varshney. Montage4d: interactive seamless fusion of multiview video textures. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pp. 5–1, 2018.

[12] J. Edelmann, P. Gerjets, P. Mock, A. Schilling, and W. Strasser. Face2face—a system for multi-touch collaboration with telepresence. In *Proceedings of the IEEE International Conference on Emerging Signal Processing Applications*, pp. 159–162, 2012.

[13] N. Flyer and G. B. Wright. Transport schemes on a sphere using radial basis functions. *Journal of Computational Physics*, 226(1):1059–1084, 2007.

[14] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5501–5510, 2022.

[15] J. Gao, H.-W. Shen, J. Huang, and J. A. Kohl. Visibility culling for time-varying volume rendering using temporal occlusion coherence. In *Proceedings of the IEEE Visualization Conference*, pp. 147–154, 2004.

[16] L. Gao, T. Fan, J. Wan, Y. Xu, J. Sun, and Z. Ma. Point cloud geometry compression via neural graph sampling. In *Proceedings of the IEEE International Conference on Image Processing*, pp. 3373–3377, 2021.

[17] M. Gross, S. Würmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, S. Lang, et al. blue-c: a spatially immersive display and 3d video portal for telepresence. *ACM Transactions on Graphics*, 22(3):819–827, 2003.

[18] R. Grosse, M. K. Johnson, E. H. Adelson, and W. T. Freeman. Ground truth dataset and baseline evaluations for intrinsic image algorithms. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2335–2342, 2009.

[19] A. Hore and D. Ziou. Image quality metrics: Psnr vs. ssim. In *Proceedings of the international conference on pattern recognition*, pp. 2366–2369, 2010.

[20] B. K. Horn. Determining lightness from an image. *Computer graphics and image processing*, 3(4):277–299, 1974.

[21] T. Huang and Y. Liu. 3d point cloud geometry compression on deep learning. In *Proceedings of the ACM international conference on multimedia*, pp. 890–898, 2019.

[22] S. Hubbert, Q. T. Lê Gia, T. M. Morton, et al. *Spherical radial basis functions, theory and applications*. Springer, 2015.

[23] I. A. P. F. Imawati, M. Sudarma, I. K. G. D. Putra, and I. P. A. Bayupati. A study of lab color space and its visualization. In *Proceedings of the International Conference on Applied Mathematics, Statistics, and Computing*, pp. 17–28. Atlantis Press, 2024.

[24] Y. Jin, X. Duan, K. Hu, F. Wang, and X. Liu. 3d video conferencing via on-hand devices. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[25] Y. Jin, J. Liu, F. Wang, and S. Cui. Ebublio: Edge-assisted multiuser 360 video streaming. *IEEE Internet of Things Journal*, 10(17):15408–15419, 2023.

[26] T. Kanade, P. Rander, and P. Narayanan. Virtualized reality: Constructing virtual worlds from real scenes. *IEEE multimedia*, 4(1):34–47, 1997.

[27] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023.

[28] M. Kettunen, E. Härkönen, and J. Lehtinen. E-lpips: robust perceptual image similarity via random transformation ensembles. *arXiv preprint arXiv:1906.03973*, 2019.

[29] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017.

[30] G. Kurillo, R. Bajcsy, K. Nahrsted, and O. Kreylos. Immersive 3d environment for remote collaboration and training of physical activities. In *Proceedings of the IEEE Virtual Reality Conference*, pp. 269–270, 2008.

[31] E. H. Land and J. J. McCann. Lightness and retinex theory. *Journal of the Optical Society of America*, 61(1):1–11, 1971.

[32] J. Lawrence, D. Goldman, S. Achar, G. M. Blascovich, J. G. Desloge, T. Fortes, E. M. Gomez, S. Häberling, H. Hoppe, A. Huibers, et al. Project starline: A high-fidelity telepresence system. *ACM Transactions on Graphics*, 40(6):1–16, 2021.

[33] J. C. Lee, D. Rho, X. Sun, J. H. Ko, and E. Park. Compact 3d gaussian representation for radiance field. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21719–21728, 2024.

[34] Z. Liang, J. Liu, M. Dasari, and F. Wang. Fumos: Neural compression and progressive refinement for continuous point cloud video streaming. *IEEE Transactions on Visualization and Computer Graphics*, 2024.

[35] S. Lin. Linear and nonlinear approximation of spherical radial basis function networks. *Journal of Complexity*, 35:86–101, 2016.

[36] S. Lin, J. Zeng, and Z. Xu. Error estimate for spherical neural networks interpolation. *Neural Processing Letters*, 42(2):369–379, 2015.

[37] T. Lu, M. Yu, L. Xu, Y. Xiangli, L. Wang, D. Lin, and B. Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20654–20664, 2024.

[38] X. Lu, J. Shen, S. Perugini, and J. Yang. An immersive telepresence system using rgb-d sensors and head mounted display. In *Proceedings of the IEEE International Symposium on Multimedia*, pp. 453–458, 2015.

[39] Y. Ma, X. Feng, X. Jiang, Z. Xia, and J. Peng. Intrinsic image decomposition: A comprehensive review. In *Proceedings of the International Conference of Image and Graphics*, pp. 626–638. Springer, 2017.

[40] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[41] W. Morgenstern, F. Barthel, A. Hilsmann, and P. Eisert. Compact 3d scene representation via self-organizing gaussian grids. *arXiv preprint arXiv:2312.13299*, 2023.

[42] A. Mossel and M. Kroeter. Streaming and exploration of dynami-

cally changing dense 3d reconstructions in immersive virtual reality. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, pp. 43–48, 2016.

[43] T. Narihira, M. Maire, and S. X. Yu. Learning lightness from human judgement on relative reflectance. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2965–2973, 2015.

[44] A. T. Nasrabadi, A. Mahzari, J. D. Beshay, and R. Prakash. Adaptive 360-degree video streaming using scalable video coding. In *Proceedings of the ACM international conference on Multimedia*, pp. 1689–1697, 2017.

[45] S. Niedermayr, J. Stumpfegger, and R. Westermann. Compressed 3d gaussian splatting for accelerated novel view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10349–10358, 2024.

[46] J. Pang, M. A. Lodhi, and D. Tian. Grasp-net: Geometric residual analysis and synthesis for point cloud compression. In *Proceedings of the 1st International Workshop on Advances in Point Cloud Compression, Processing and Analysis*, pp. 11–19, 2022.

[47] M. Podpora, G. P. Korbas, and A. Kawala-Janik. Yuv vs rgb-choosing a color space for human-machine interaction. In *FedCSIS (Position Papers)*, pp. 29–34. Citeseer, 2014.

[48] D. L. Post. Color and human-computer interaction. In *Handbook of human-computer interaction*, pp. 573–615. Elsevier, 1997.

[49] K. Ren, L. Jiang, T. Lu, M. Yu, L. Xu, Z. Ni, and B. Dai. Octree-gs: Towards consistent real-time rendering with lod-structured 3d gaussians. *arXiv preprint arXiv:2403.17898*, 2024.

[50] G. Saravanan, G. Yamuna, and S. Nandhini. Real time implementation of rgb to hsv/hsi/hsl and its reverse color space models. In *Proceedings of the International Conference on Communication and Signal Processing*, pp. 0462–0466. IEEE, 2016.

[51] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the scalable video coding extension of the h. 264/avc standard. *IEEE Transactions on circuits and systems for video technology*, 17(9):1103–1120, 2007.

[52] K. Shen, D. Zhang, Z. Zhu, L. Zhang, F. Wang, and D. Wang. Sja: Server-driven joint adaptation of loss and bitrate for multi-party realtime video streaming. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*, pp. 1–10. IEEE, 2023.

[53] P. Stotko, S. Krumpen, M. B. Hullin, M. Weinmann, and R. Klein. Slamcast: Large-scale, real-time 3d reconstruction and streaming for immersive multi-client live telepresence. *IEEE transactions on visualization and computer graphics*, 25(5):2102–2112, 2019.

[54] P. Stotko, S. Krumpen, R. Klein, and M. Weinmann. Towards scalable sharing of immersive live telepresence experiences beyond room-scale based on efficient real-time 3d reconstruction and streaming. In *Proceedings of the CVPR Workshop on Computer Vision for Augmented and Virtual Reality*, vol. 3, 2019.

[55] P. Stotko, S. Krumpen, M. Schwarz, C. Lenz, S. Behnke, R. Klein, and M. Weinmann. A vr system for immersive teleoperation and live exploration with a mobile robot. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3630–3637, 2019.

[56] C. Sun, M. Sun, and H.-T. Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5459–5469, 2022.

[57] S. Süsstrunk, R. Buckley, and S. Swen. Standard rgb color spaces. In *Color and imaging conference*, vol. 7, pp. 127–134. Society of Imaging Science and Technology, 1999.

[58] L. Van Holland, P. Stotko, S. Krumpen, R. Klein, and M. Weinmann. Efficient 3d reconstruction, streaming and visualization of static and dynamic scene parts for multi-client live-telepresence in large-scale environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4258–4272, 2023.

[59] L. Wei, Y. Liu, F. Wang, D. Zhang, and D. Wang. Vsas: Decision transformer-based on-demand volumetric video streaming with passive frame dropping. *IEEE Internet of Things Journal*, 2023.

[60] Y. Xu, T. Yang, Z. Tan, and H. Lan. Fov-based coding optimization for 360-degree virtual reality videos. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1640–1644, 2022.

[61] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli. A control-theoretic approach for dynamic adaptive video streaming over http. In *Proceedings of the ACM Conference on Special Interest Group on Data Communication*, pp. 325–338, 2015.

[62] D. Zhang, K. Shen, F. Wang, D. Wang, and J. Liu. Towards joint loss and bitrate adaptation in realtime video streaming. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 1–6, 2022.

[63] D. Zhang, L. Wei, K. Shen, H. Zhu, D. Wang, and F. Wang. Trimstream: Adaptive realtime video streaming through intelligent frame retrospection in adverse network conditions. *IEEE Transactions on Mobile Computing*, 2024.

[64] D. Zhang, H. Zhu, K. Shen, D. Wang, and F. Wang. Dsja: Distributed server-driven joint route scheduling and streaming adaptation for multi-party realtime video streaming. *IEEE Transactions on Mobile Computing*, 2023.

[65] S. Zhang and W. C. Ho. Tele-immersive interaction with intelligent virtual agents based on real-time 3d modeling. *Journal of Multimedia*, 7(1):57, 2012.

[66] Q. Zhao, P. Tan, Q. Dai, L. Shen, E. Wu, and S. Lin. A closed-form solution to retinex with nonlocal texture constraints. *IEEE transactions on pattern analysis and machine intelligence*, 34(7):1437–1444, 2012.

[67] T. Zhou, P. Krahenbuhl, and A. A. Efros. Learning data-driven reflectance priors for intrinsic image decomposition. In *Proceedings of the IEEE international conference on computer vision*, pp. 3469–3477, 2015.