Low-Rank Approximation for Efficient Compression of Gaussian Splatting Spherical Harmonics

Zhiwei Zhu, Sicheng Li, Yiyi Liao†, and Lu Yu†, *Senior Member, IEEE*

***Abstract*—3D Gaussian Splatting (3DGS) enables real-time, high-fidelity rendering but suffers from large model sizes, mainly due to the Spherical Harmonics (SH) coefficients used for view-dependent appearance modeling, whose parameter count scales quadratically ($\mathcal{O}(L^2)$) with the degree L. This paper presents the first systematic study that reveals and leverages the intrinsic low-rank structure of SH coefficients in 3DGS. Unlike prior approaches that truncate spectral energy, the proposed low-rank paradigm compactly preserves spectral information, achieving high visual quality with substantially reduced storage. Two complementary approaches are introduced. (1) SHAC-PCA (Principal Component Analysis) is a plug-and-play post-hoc compressor that retains principal spectral variance for high-fidelity compression. (2) SHAC-LST (Learned Subset Transformation) is a training-integrated approach that decomposes Alternating Current (AC) of SH coefficients into low-dimensional subset coefficients and a shared transformation matrix, offering superior compression and even quality improvements through regularization. Both methods effectively reduce SH coefficients storage complexity to $\mathcal{O}(L)$. Extensive experiments demonstrate that our approaches significantly reduce memory usage while maintaining or even improving rendering quality. The proposed techniques are highly versatile: SHAC-PCA can be applied to any pre-trained 3DGS model, while SHAC-LST supports end-to-end training or fine-tuning. Both methods are compatible with existing 3DGS compression pipelines, providing a practical and general solution for efficient compression of SH coefficients in 3DGS.**

***Index Terms*—Gaussian Splatting Compression, Spherical Harmonics, Low-Rank Approximation**

I. Introduction

VOLUMETRIC video, the next generation of visual media, enables free-viewpoint exploration of dynamic 3D content, offering vast potential across entertainment, education, and beyond. Owing to its photorealistic quality and real-time rendering, 3D Gaussian Splatting (3DGS) [1] has gradually become a leading representation for volumetric video. However, storage efficiency remains a critical bottleneck for its widespread adoption and deployment [2]. A key contributor to this overhead is the use of Spherical Harmonics (SH) coefficients for view-dependent appearance modeling. For the standard degree $L = 3$, SH coefficients constitute the vast majority (over 80%) of per-Gaussian parameters and scale quadratically ($\mathcal{O}\left(L^2\right)$) with L, severely limiting storage efficiency. This motivates the development of effective compression strategies specifically targeting SH coefficients.

Zhiwei Zhu, Sicheng Li, Yiyi Liao, Lu Yu are with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China. (email: zhuzhiwei21@zju.edu.cn; jasonlisicheng@zju.edu.cn; yiyi.liao@zju.edu.cn; yul@zju.edu.cn).

†Corresponding authors: Yiyi Liao and Lu Yu.

Existing approaches for compacting SH coefficients primarily focus on reducing the number of spectral dimensions per Gaussian. Representative techniques include *truncation* [3], which discards higher-degree coefficients at the cost of sacrificing high-frequency details; *distillation* [4], which transfers information to lower-degree representations but introduces additional training overhead; and *adaptive masking* [5], [6], which learns per-primitive SH degrees but increases model complexity. Alternatively, some methods replace SH coefficients with learned *latent features* [7], [8], [9], [10], increasing decoding latency and degrading rendering efficiency. While these techniques provide various trade-offs between compactness and fidelity, they generally overlook the potential for compression through the exploitation of intrinsic correlations among SH coefficients across all Gaussians within a scene.

In this work, we investigate the low-rank structure of SH coefficients in 3DGS. Both theoretical analysis and empirical observation reveal that these coefficients exhibit strong correlations, driven by factors such as directional lighting and shared material properties. This inherent redundancy suggests that SH coefficients can be effectively represented in a low-rank subspace, enabling compression strategies that preserve spectral information rather than truncating spectral dimensions.

Building upon this insight, we first demonstrate the effectiveness of low-rank approximation through a post-hoc method, **SHAC-PCA**, which applies Principal Component Analysis (PCA) to the Alternating Current (AC) components of pretrained SH coefficients. By retaining only the principal components, SHAC-PCA achieves substantial compression while preserving high fidelity. The success of SHAC-PCA motivates a subsequent investigation: can the low-rank constraint be integrated directly into training, serving simultaneously as a compression mechanism and a structural regularizer? However, traditional rank minimization techniques (e.g., nuclear norm minimization) are computationally infeasible for 3DGS training because they require expensive Singular Value Decompositions (SVDs) at each iteration. To address this, we proposed **SHAC-LST**, a Learned Subset Transformation (LST) that embeds low-rank constraints into the training process through explicit factorization. SHAC-LST explicitly decomposes the SH AC coefficients into compact, physically motivated subset coefficients and a shared, structured transformation matrix. This formulation enforces low-rank factorization without additional overhead, offering both parameter efficiency and regularization during optimization. Empirically, SHAC-LST achieves strong compression while preserving high fidelity, in some cases, even surpassing the baseline in visual fidelity.

Our study evolves from SHAC-PCA, a post-hoc compression

approach, to SHAC-LST, a training-integrated framework. In contrast to existing methods that truncate spectral energy, our strategies leverage low-rank approximation to compactly preserve information distributed across the entire SH spectrum, thereby achieving superior rendering quality with substantially reduced storage. These approaches provide practical solutions for developing compact 3DGS models. In summary, our main contributions are as follows:

- To the best of our knowledge, we are the first to present a systematic analysis and exploitation of the inherent low-rank structure of SH coefficients for efficient compression.
- We propose SHAC-PCA, a plug-and-play post-hoc method that retains principal spectral variance, achieving strong compression while maintaining high visual fidelity.
- We introduce SHAC-LST, which embeds low-rank factorization directly into 3DGS training, delivering higher compression ratios and even improving rendering quality through regularization.
- Both methods exhibit strong versatility. Extensive experiments demonstrate that both methods can be effectively combined with existing 3DGS compression frameworks, enhancing their performance.

II. Related Works

A. Gaussian Splatting Compression

The substantial storage overhead of 3DGS models has motivated extensive research on Gaussian Splatting Compression (GSC) [11], [12]. Existing approaches can be broadly categorized into two paradigms, a dichotomy that is notably mirrored in the ongoing standardization efforts by the Moving Picture Experts Group (MPEG) [13], [14].

The first category, *Vanilla-GS-based* methods (aligning with *Inria-track* of MPEG GSC), aims to compress pre-trained vanilla 3DGS models with low complexity while remaining compatible with standard rasterization pipelines. Representative techniques include pruning [4], [15], [16], [17], vector quantization [4], [15], [18], [19], point-cloud-based codecs [20], [10], and 3D-to-2D mapping that leverage standard 2D codecs [3], [21], [22]. This paradigm prioritizes engineering practicality and widespread deployment.

The second category, *Scaffold-GS-based* methods (aligning with *Alternate-track* of MPEG GSC), typically extends anchor-based 3DGS representations like Scaffold-GS [23] with tailored context models [24], [25], [26], [27], [28]. While these approaches achieve state-of-the-art rate-distortion (RD) performance through efficient entropy coding, they alter the native 3DGS format. This modification often reduces tool compatibility, degrades rendering efficiency, and increases decoding latency due to the complexity of context modeling.

Given 3DGS's potential as a universal 3D data format, we prioritize the compatibility and standardization benefits of the first paradigm. Consequently, our work follows the *Vanilla-GS-based* route, targeting the efficient compression of SH coefficients within the native 3DGS representation.

B. SH Coefficients Compaction

Existing approaches for SH coefficient compaction primarily focus on reducing the number of stored spectral dimensions per Gaussian. Representative methods include: *Truncation*, which discards high-degree coefficients [3] but degrades view-dependent fidelity; *Distillation*, which trains low-degree SH Coefficients to emulate higher-degree ones via pseudo-view supervision [4], improving quality but adding training overhead; *Adaptive Masking*, which learns per-primitive degrees through masks [5], [6], offering flexibility but increasing model complexity; and *Latent Features*, which replace SH coefficients with learned compact latent representations [7], [8], [9], require extra decoders and often exhibit unstable convergence.

Most existing approaches either truncate the SH spectrum or incur substantial computational overhead, while overlooking the intrinsic correlations present in SH coefficients. In contrast, we explicitly exploit this low-rank structure. Our methods employ low-rank approximation to decompose the full SH coefficients into compact representations, effectively preserving distributed spectral information while achieving high rendering fidelity and parameter efficiency.

C. Low-Rank Approximation.

Low-Rank Approximation (LRA) is a fundamental technique for dimensionality reduction and capturing latent structures, with widespread applications in image and video compression [29], [30], multi-view clustering [31], [32], and 3D geometry filtering [33], [34]. In machine learning, LRA is crucial for developing efficient deep neural networks by compressing weight matrices [35], [36]. More recently, low-rank updates have gained prominence in parameter-efficient fine-tuning (PEFT) of large pre-trained models, exemplified by LoRA [37] and its variants [38], [39], highlighting LRA's ability to compactly model complex functions. Motivated by these successes, to the best of our knowledge, we are the first to systematically study the low-rank structure of SH coefficients in 3DGS and leverage LRA to achieve efficient compression.

III. Preliminaries

This section provides a brief overview of the fundamental concepts underlying our work. We first describe the 3DGS representation and then discuss the key properties and characteristics of its SH coefficients.

A. Gaussian Splatting

3DGS models scenes using anisotropic 3D Gaussians, combining the differentiable properties of volumetric representations with efficient tile-based rasterization for rendering. The spatial distribution of each Gaussian i is defined by its center position $\boldsymbol{\mu}_i \in \mathbb{R}^3$ and a covariance matrix $\boldsymbol{\Sigma}_i \in \mathbb{R}^{3\times3}$:

$$G_i(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)}, \tag{1}$$

where $\mathbf{x}$ is a point in 3D scene. $\boldsymbol{\Sigma}_i$ is formulated by a scaling vector $\mathbf{s}_i \in \mathbb{R}^3$ and a rotation quaternion $\mathbf{q}_i \in \mathbb{S}^3$ (representing rotation matrix $\mathbf{R}_i \in SO(3)$) to ensure positive semi-definite during optimization:

$$\boldsymbol{\Sigma}_i = \mathbf{R}_i \mathbf{S}_i \mathbf{S}_i^\top \mathbf{R}_i^\top, \quad \text{where } \mathbf{S}_i = \mathrm{diag}(\mathbf{s}_i). \tag{2}$$

The appearance of each Gaussian i is associated with opacity $\alpha_i \in [0,1]$, and spherical harmonics (SH) coefficients $\mathbf{h}_i \in \mathbb{R}^K$, where $K = 3(L+1)^2$ for degree L.

During rendering, 3D Gaussians are projected onto the image plane as 2D Gaussians $G'_i(\mathbf{x}')$. The final color $C(\mathbf{x}')$ at pixel $\mathbf{x}'$ is obtained by α-blending the contributions from N overlapping, depth-sorted 2D Gaussians:

$$C(\mathbf{x}') = \sum_{i \in N} \mathbf{c}_i \alpha'_i \prod_{j=1}^{i-1}(1-\alpha'_j), \quad \alpha'_i = \alpha_i G'_i(\mathbf{x}'), \tag{3}$$

where $\mathbf{c}_i \in \mathbb{R}^3$ is the view-dependent color derived from SH coefficients and viewing direction. In summary, the set of learnable parameters for each Gaussian i typically comprises:

$$\mathcal{G}_i = \{\boldsymbol{\mu}_i, \mathbf{s}_i, \mathbf{q}_i, \alpha_i, \mathbf{h}_i\}. \tag{4}$$

The parameter set composition shows that **SH coefficients dominate the memory footprint**, consuming 48 of the total 59 parameters (**81.4%**) per Gaussian for default degree $L = 3$. Furthermore, the storage requirement for SH coefficients grows quadratically with the degree L, indicating that SH compression is crucial for efficient 3DGS compression.

B. Spherical Harmonics

Spherical Harmonics (SH) are orthogonal basis functions defined on the unit sphere (S^2), widely used in graphics for representing directional functions like view-dependent appearance [40]. Using spherical coordinates $\Omega = (\theta, \varphi)$ (polar angle $\theta \in [0, \pi]$, azimuthal angle $\varphi \in [0, 2\pi]$), the complex-valued SH basis functions $Y_l^m : S^2 \to \mathbb{C}$ are defined as:

$$Y_l^m(\Omega) = K_l^m P_l^m(\cos\theta) e^{jm\phi}, \tag{5}$$

where $l \in \mathbb{N}_0$ is the degree, $m \in \{-l, ..., l\}$ is the order. P_l^m are the associated Legendre polynomials and K_l^m is a normalization constant. Any square-integrable function $f : S^2 \to \mathbb{C}$ can be uniquely expanded as:

$$f(\Omega) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} h_l^m Y_l^m(\Omega). \tag{6}$$

The coefficients h_l^m are given by the projection of $f(\Omega)$ onto each basis function $Y_l^m(\Omega)$:

$$h_l^m = \int_{S^2} f(\Omega)[Y_l^m(\Omega)]^* \, d\Omega. \tag{7}$$

For real-valued signals (e.g., RGB color), a real, orthonormal SH basis y_l^m is typically derived from linear combinations of the complex basis Y_l^m [40]. In 3DGS, each Gaussian i stores learned coefficients $\mathbf{h}_i$ for this real basis, encoding its directional RGB color. During rendering, the color $\mathbf{c}_i \in \mathbb{R}^3$ for a viewing direction $\mathbf{d} \in S^2$ is computed via the real SH expansion up to degree L:

$$\mathbf{c}_i = \sum_{l=0}^{L} \sum_{m=-l}^{l} \mathbf{h}_{i,lm} y_l^m(\mathbf{d}), \tag{8}$$

where $\mathbf{h}_{i,lm} \in \mathbb{R}^3$ are coefficients corresponding to the real basis function y_l^m.

Critically, the SH evaluation Eq. (8) relies on viewing directions $\mathbf{d}$ and basis functions y_l^m defined within a consistent coordinate frame. Consequently, the coefficients $\mathbf{h}_{i,lm}$ across all Gaussians can be interpreted as learned projections of local scene appearance onto this *shared basis* relative to this *shared frame*. This observation motivates **viewing the entire set of coefficients $\{\mathbf{h}_i\}_{i=1}^N$ as samples drawn from an underlying probability distribution that characterizes the scene's anisotropic appearance field.** This perspective forms the basis of our hypothesis on the low-rank structure inherent in these coefficients.

IV. Low-Rank Analysis of SH Coefficients

Building on the interpretation of SH coefficients as samples from the appearance field, we hypothesize that SH coefficients exhibit strong correlations and inherent low-rank structure. This stems from anisotropic real-world appearance signals—driven by factors such as directional lighting and shared material properties—simultaneously excite multiple SH basis functions, inducing statistical dependencies among coefficients. This section analyzes this hypothesis through both theoretical justification (Sec. IV-A) and empirical validation (Sec. IV-B).

A. Theoretical Justification

1) Overview: The theory of random fields on the sphere (S^2) characterizes the correlation structure of SH coefficients according to the isotropy of the underlying field [41], [42]. In the case of a statistically isotropic (i.e., rotationally invariant) appearance field, the SH coefficients $\mathbf{h}_{lm}$ are mutually uncorrelated, resulting in a diagonal covariance matrix:

$$\mathbb{E}[\mathbf{h}_{l_1 m_1} \mathbf{h}_{l_2 m_2}^*] = C_l \delta_{l_1 l_2} \delta_{m_1 m_2}, \tag{9}$$

where * denotes the complex conjugate, $\mathbb{E}[\cdot]$ is expectation, C_l is the angular power spectrum, and δ_{ij} is the Kronecker delta (1 if $i = j$, 0 otherwise). However, real-world scene appearance is inherently anisotropic. In such non-isotropic fields, rotational invariance no longer holds, leading to a non-diagonal covariance structure as predicted by the theory:

$$\mathbb{E}[\mathbf{h}_{l_1 m_1} \mathbf{h}_{l_2 m_2}^*] = C_{l_1 m_1, l_2 m_2}. \tag{10}$$

This indicates that correlations ($C_{l_1 m_1, l_2 m_2} \neq 0$) naturally arise between coefficients of different degrees l or orders m.

Therefore, modeling scene appearance as an anisotropic random field sampled by the Gaussians theoretically implies that the corresponding SH coefficients $\mathbf{h}_{lm}$ will exhibit inter-coefficient correlations. This provides a principled motivation for employing low-rank approximation methods to exploit these correlations for compression.

2) Mathematical Proof: The proof that the SH coefficients of a statistically isotropic random field on the sphere are uncorrelated is as follows:

Theorem 1 ([41]): For a zero-mean, statistically isotropic random field $f(\Omega)$, the SH coefficients h_l^m are uncorrelated across different degrees l or orders m. Their covariance is given by:

$$\mathbb{E}[h_{l_1}^{m_1} (h_{l_2}^{m_2})^*] = C_{l_1} \delta_{l_1 l_2} \delta_{m_1 m_2}, \tag{11}$$

where C_l is the angular power spectrum of the field f, defined as:

$$C_l = \frac{1}{2l+1} \sum_{m=-l}^{l} \mathbb{E}[|h_l^m|^2], \tag{12}$$

and δ_{ij} is the Kronecker delta:

$$\delta_{ij} = \begin{cases} 1, & \text{if} \quad i = j, \\ 0, & \text{if} \quad i \neq j. \end{cases} \tag{13}$$

Proof: The proof proceeds in three steps:

Step 1: Covariance Function Expansion. The two-point covariance function of the field is defined as $C(\Omega_1, \Omega_2) = \mathbb{E}[f(\Omega_1) f^*(\Omega_2)]$. Substituting the SH expansion of f, we get:

$$\begin{aligned} C(\Omega_1, \Omega_2) = \mathbb{E}\Bigg[& \Big(\sum_{l_1=0}^{\infty} \sum_{m_1=-l_1}^{l_1} h_{l_1}^{m_1} Y_{l_1}^{m_1}(\Omega_1) \Big) \\ & \cdot \Big(\sum_{l_2=0}^{\infty} \sum_{m_2=-l_2}^{l_2} h_{l_2}^{m_2} Y_{l_2}^{m_2}(\Omega_2) \Big)^* \Bigg] \\ = & \sum_{l_1,m_1} \sum_{l_2,m_2} \mathbb{E}[h_{l_1}^{m_1} (h_{l_2}^{m_2})^*] \, Y_{l_1}^{m_1}(\Omega_1) \, [Y_{l_2}^{m_2}(\Omega_2)]^*. \end{aligned} \tag{14}$$

Here, summations run over all valid $l \geq 0$ and $|m| \leq l$.

To find the coefficient $\mathbb{E}[h_{l_1}^{m_1}(h_{l_2}^{m_2})^*]$, we can project $C(\Omega_1, \Omega_2)$ onto the basis $Y_{l_1}^{m_1}(\Omega_1)[Y_{l_2}^{m_2}(\Omega_2)]^*$ by integrating over Ω_1 and Ω_2:

$$\begin{aligned} \mathbb{E}[h_{l_1}^{m_1} (h_{l_2}^{m_2})^*] = & \iint_{S^2 \times S^2} C(\Omega_1, \Omega_2) \\ & [Y_{l_1}^{m_1}(\Omega_1)]^* Y_{l_2}^{m_2}(\Omega_2) d\Omega_1 d\Omega_2 \end{aligned} \tag{15}$$

Step 2: Isotropy Constraint. For a statistically isotropic field, the covariance function $C(\Omega_1, \Omega_2)$ depends only on the angular separation γ between the points Ω_1 and Ω_2, where $\cos\gamma = \Omega_1 \cdot \Omega_2$. Therefore, $C(\Omega_1, \Omega_2) = C(\gamma)$. Such a function can be expanded in terms of Legendre polynomials P_l:

$$C(\Omega_1, \Omega_2) = C(\gamma) = \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} C_l P_l(\cos\gamma). \tag{16}$$

Here, C_l are the coefficients of the expansion, which correspond to the angular power spectrum. We can use the *addition theorem* for spherical harmonics, which states:

$$P_l(\cos\gamma) = P_l(\Omega_1 \cdot \Omega_2) = \frac{4\pi}{2l+1} \sum_{m=-l}^{l} Y_l^m(\Omega_1) [Y_l^m(\Omega_2)]^*. \tag{17}$$

Substituting Eq. (17) into Eq. (16) gives the covariance function for an isotropic field in terms of SH basis functions:

$$C(\Omega_1, \Omega_2) = \sum_{l=0}^{\infty} C_l \sum_{m=-l}^{l} Y_l^m(\Omega_1) [Y_l^m(\Omega_2)]^*. \tag{18}$$

Substituting Eq. (18) into Eq. (15) gives:

$$\begin{aligned} \mathbb{E}[h_{l_1}^{m_1} (h_{l_2}^{m_2})^*] = & \iint_{S^2 \times S^2} \left(\sum_{l,m} C_l \, Y_l^m(\Omega_1) \, [Y_l^m(\Omega_2)]^* \right) \\ & \cdot [Y_{l_1}^{m_1}(\Omega_1)]^* \, Y_{l_2}^{m_2}(\Omega_2) \, d\Omega_1 \, d\Omega_2 \\ = & \sum_{l,m} C_l \left(\int_{S^2} Y_l^m(\Omega_1) \, [Y_{l_1}^{m_1}(\Omega_1)]^* d\Omega_1 \right) \\ & \cdot \left(\int_{S^2} [Y_l^m(\Omega_2)]^* \, Y_{l_2}^{m_2}(\Omega_2) d\Omega_2 \right). \end{aligned} \tag{19}$$

Step 3: Expansions using Orthogonality. Using the *orthogonality property* of spherical harmonics, which states:

$$\int_{S^2} Y_{l_1}^{m_1}(\Omega) [Y_{l_2}^{m_2}(\Omega)]^* d\Omega = \delta_{l_1 l_2} \delta_{m_1 m_2}. \tag{20}$$

Eq. (19) can be expressed as

$$\mathbb{E}[h_{l_1}^{m_1} (h_{l_2}^{m_2})^*] = \sum_{l,m} C_l \delta_{l l_1} \delta_{m m_1} \delta_{l l_2} \delta_{m m_2}, \tag{21}$$

The sums collapse due to the Kronecker deltas. For the expression to be non-zero, the first pair of delta functions requires $l = l_1$ and $m = m_1$, while the second pair requires $l = l_2$ and $m = m_2$. This simultaneously requires $l_1 = l_2$ and $m_1 = m_2$. When these conditions are met, the only term remaining in the sum is for $l = l_1 (= l_2)$ and $m = m_1 (= m_2)$, and its value is C_{l_1}. Therefore, we obtain:

$$\mathbb{E}[h_{l_1}^{m_1} (h_{l_2}^{m_2})^*] = C_{l_1} \delta_{l_1 l_2} \delta_{m_1 m_2}. \tag{22}$$

This confirms that the coefficients are uncorrelated unless $l_1 = l_2$ and $m_1 = m_2$, in which case their covariance depends only on the degree l_1 via the angular power spectrum C_{l_1}. ■

B. Empirical Validation

To empirically verify the correlations and assess the potential for low-rank approximation, we analyze the SH AC coefficients matrix $\mathbf{A} \in \mathbb{R}^{N \times K'}$ ($K' = K - 3$), constructed by stacking SH AC coefficients ($l > 0$) from N Gaussians. The Direct Current (DC) coefficients ($l = 0$) are excluded, as they dominate the overall energy. Evaluations are performed on 13 diverse scenes from standard 3DGS benchmark datasets (Mip-NeRF 360 [43], Tanks and Temples [44], and Deep Blending [45]) to ensure the results are generalizable.

1) Correlation Analysis: We compute the Pearson correlation matrix $\mathbf{C} \in \mathbb{R}^{K' \times K'}$ for the K' dimensions of $\mathbf{A}$. First, $\mathbf{A}$ is column-wise standardization:

$$\mathbf{A}' = (\mathbf{A} - \mathbf{1}\mu^\top)\mathbf{D}_\sigma^{-1}, \tag{23}$$

where $\mu \in \mathbb{R}^{1 \times K'}$ is the column-wise mean vector, and $\mathbf{D}_\sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_{K'})$ is a diagonal matrix of column-wise standard deviations σ_j. The Pearson correlation matrix is then computed as:

$$\mathbf{C} = \frac{1}{N-1} \mathbf{A'}^\top \mathbf{A}'. \tag{24}$$

The visualization of Pearson correlation is shown in Fig. 1, leading to the following observations:

- **Inter-component correlations:** Strong positive correlations are observed among the RGB components within the same SH basis y_{lm}.
- **Intra-order correlations:** Evident correlations exist among coefficients sharing the same order m.

As illustrated in Fig. 2, SH bases with the same order share the same longitude frequency, which may account for the observed intra-order correlations among the SH AC coefficients.

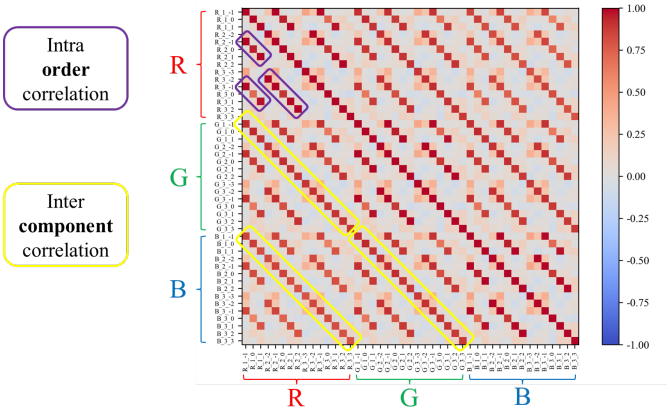

Fig. 1. Pearson correlation heatmap of SH AC coefficients. Strong off-diagonal correlations reveal both inter-component correlations (among the R, G, and B components within the same degree l and order m) and intra-order correlations (among coefficients sharing the same order m).

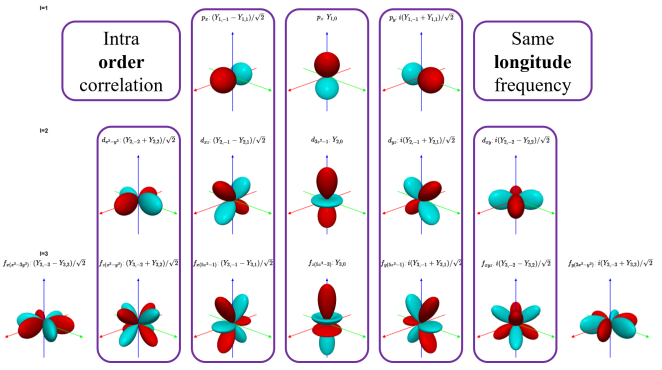

Fig. 2. The observed intra-order correlations among SH AC coefficients may likely stem from the fact that SH bases of the same order share the same longitude frequency.

2) SVD Analysis: To directly quantify the low-rank structure, we perform Singular Value Decomposition (SVD) on the normalized SH AC coefficients matrix $\mathbf{A}'$:

$$\mathbf{A}' = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top, \tag{25}$$

where $\mathbf{U}$ and $\mathbf{V}$ contain the left and right singular vectors, and $\mathbf{\Sigma}$ is a diagonal matrix containing the singular values s_k sorted in descending order. The effective rank is evaluated using the cumulative energy retention (ER) as a function of the number of retained singular values r:

$$\mathrm{ER}(r) = \frac{\sum_{k=1}^{r} s_k^2}{\sum_{k=1}^{K'} s_k^2}. \tag{26}$$

TABLE I
COMPARISON OF SH COEFFICIENT RANKS REQUIRED FOR 95% ENERGY RETENTION AND THE ENERGY CAPTURED BY 24 COMPONENTS ACROSS 13 SCENES FROM 3 BENCHMARK DATASETS (MIP-NERF 360, TANKS AND TEMPLES, DEEP BLENDING).

Scene	ER=95% (Rank)		Energy@24 (%)	
	SHAC coeff.	SVD singu.	SHAC coeff.	SVD singu.
bicycle	44	15	50.8	98.7
bonsai	44	21	49.3	96.4
counter	44	22	52.7	96.6
flowers	44	19	51.4	97.1
garden	44	26	49.9	94.1
kitchen	44	23	49.0	95.8
room	43	23	52.6	95.6
stump	44	21	51.2	97.0
treehill	44	17	51.8	98.2
train	44	22	52.3	96.4
truck	44	25	50.4	94.9
drjohnson	44	28	50.5	92.9
playroom	43	32	56.5	88.3
Average	**43.8**	**22.6**	**51.4**	**95.5**

The results summarized in Table I reveal a clear low-rank structure across diverse scenes.

- **Low effective rank:** Retaining 95% of the total energy requires only $r \approx 23$ components on average, far fewer than the full $K' = 45$ SH AC coefficients dimensions ($L = 3$), indicating the low effective rank.
- **SVD efficiency:** With $r = 24$ components, SVD retains 95.5% of the energy, whereas degree truncation to $L = 2$ (also 24 coefficients) retains only 51.4%. This demonstrates that SVD more effectively captures correlated variance across SH AC coefficients.

V. Low-Rank Approximation of SH Coefficients

Theoretical (Sec. IV-A) and empirical (Sec. IV-B) analyses confirm that SH coefficients in 3DGS exhibit significant correlations and possess a pronounced low-rank structure. This strongly motivates our proposed compression methods, which, unlike spectral truncation, **utilize low-rank approximation to preserve information distributed across the SH spectrum more effectively.** We present two approaches: SHAC-PCA, a direct post-hoc application of PCA to efficiently reducing the dimension, followed by SHAC-LST, a more integrated approach using low-rank factorization during learning.

A. SHAC-PCA: Principal Component Analysis based Approximation

SHAC-PCA applies PCA post-hoc to compress the SH coefficients of a pre-trained 3DGS model. It functions as a plug-and-play preprocessing module that operates solely on the SH AC coefficients $\mathbf{A} \in \mathbb{R}^{N \times K'}$. The DC coefficients are preserved directly. This separation is essential for two reasons: (1) it prevents the typically high variance of DC terms from dominating the PCA, and (2) it avoids rotational mixing between DC (base color) and AC (view-dependent) components, which would otherwise introduce severe base color artifacts when principal components are truncated. Detailed empirical

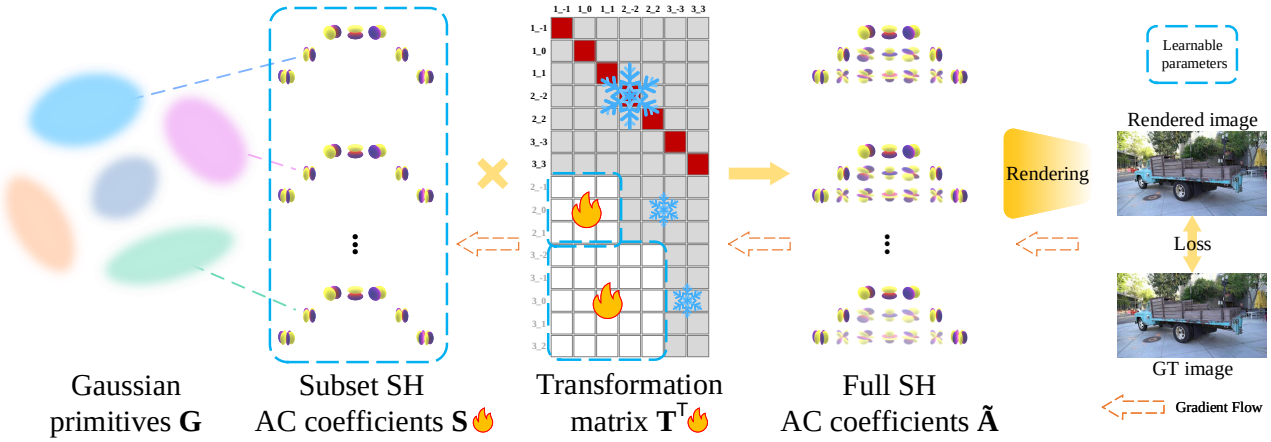

Fig. 3. Overview of the proposed SHAC-LST method. A compact subset $\mathbf{S}$ is selected using the physically motivated strategy in (32). A block-diagonal transformation matrix $\mathbf{T}$ is then learned to model correlations between the subset and the remaining coefficients, reconstructing the full set as $\tilde{\mathbf{A}} = \mathbf{ST}$. For clarity, the illustration shows a single color channel; in practice, all RGB channels are processed jointly.

validation of this design choice is provided in Sec. VII-D. Furthermore, to exploit *inter-component correlations*, the RGB components are jointly processed during PCA.

1) SHAC-PCA Pipeline: PCA can be efficiently computed via Singular Value Decomposition (SVD). Following Eq. (25), the first M columns of the right singular matrix $\mathbf{V}$ are selected as the principal components:

$$\mathbf{P} = \mathbf{V}_{:,1:M} \in \mathbb{R}^{K' \times M}, \tag{27}$$

where the rank M is determined by the desired energy retention (e.g., 95%) or a target compression ratio. The compressed representation $\mathbf{F}$ is obtained by projecting the normalized SH AC coefficients matrix $\mathbf{A}'$ onto these principal components:

$$\mathbf{F} = \mathbf{A}'\mathbf{P}, \quad \mathbf{F} \in \mathbb{R}^{N \times M}. \tag{28}$$

To reconstruct an approximation $\tilde{\mathbf{A}}$ of the original SH AC coefficients, the features $\mathbf{F}$ are projected back to the original space, followed by rescaling and mean restoration:

$$\tilde{\mathbf{A}} = \left(\mathbf{F}\mathbf{P}^\top + \mathbf{1}\mu^\top\right)\mathbf{D}_\sigma, \tag{29}$$

2) Compression Analysis: The storage cost comprises: the SH DC coefficients $\mathbf{D}$ ($N \times 3$), the compressed representation $\mathbf{F}$ ($N \times M$), the principal components $\mathbf{P}$ ($K' \times M$), and the AC mean $\boldsymbol{\mu}$ and standard deviation $\boldsymbol{\sigma}$ (each of dimension K').

The resulting compression ratio (CR) relative to the full SH matrix $\mathbf{H} \in \mathbb{R}^{N \times K}$ is:

$$\mathrm{CR}_{\mathrm{PCA}} = \frac{N(3+M) + K'M + 2K'}{NK}. \tag{30}$$

For large N, the contribution of K' becomes negligible, giving the approximation:

$$\lim_{N \to \infty} \mathrm{CR}_{\mathrm{PCA}} = \frac{3+M}{K}, \tag{31}$$

highlighting substantial storage savings when $M \ll K$.

B. SHAC-LST: Learned Subset Transformation Based Approximation

SHAC-LST integrates the low-rank constraint directly into training, serving both as a compression mechanism and a structural regularizer. Unlike generic low-rank matrix factorization, the design of SHAC-LST is physically motivated. Guided by two fundamental SH properties, namely *low-degree energy concentration* and *strong intra-order correlations*, we decompose SH AC coefficients into a compact subset $\mathbf{S}$ and a structured transformation $\mathbf{T}$ (Fig. 3). We further introduce a scale regularization loss to constrain the magnitude of $\mathbf{T}$, preventing noise amplification.

1) Subset Selection: The subset selection strategy is guided by two fundamental SH properties:

- **Low-degree energy concentration:** Real-world signals typically exhibit strong energy concentration in low-degree SH coefficients due to low-frequency dominance.
- **Strong intra-order correlations:** SH coefficients sharing the same order m have identical azimuthal frequencies, leading to strong intra-order correlations.

Based on these properties, we select specific indices of SH AC coefficients to form the subset matrix $\mathbf{S} \in \mathbb{R}^{N \times d}$. This subset captures diverse azimuthal frequencies while remaining compact. Specifically, for each order m, we retain the coefficient corresponding to the lowest degree $l = |m|$:

$$\text{Subset} = \{(l,m) \mid 1 \le l \le L,\ l = 1 \text{ or } |m| = l\}. \tag{32}$$

This yields $d = 3(2L+1)$ subset dimensions (e.g., $d = 21$ for $L = 3$). Such a design ensures that every azimuthal frequency m in the full SH expansion is represented in the subset, providing a broad azimuthal footprint. The remaining coefficients ($l > \max(1, |m|)$) are predicted by the subset $\mathbf{S}$ and transformation $\mathbf{T}$.

2) Structured Transformation: The core of SHAC-LST lies in learning a transformation $\mathbf{T} \in \mathbb{R}^{d \times K'}$ that reconstructs the full SH AC coefficients from the selected subset:

$$\tilde{\mathbf{A}} = \mathbf{ST}. \tag{33}$$

Motivated by the observation that SH AC coefficients of the same order m exhibit strong correlations while inter-order dependencies are generally weak, we parameterize $\mathbf{T}$ as a block-diagonal matrix (Fig. 3). Specifically, the selected subset coefficients are preserved through a fixed identity submatrix, while *intra-order correlations* are captured by learnable square blocks $\mathbf{t}_l$. All remaining entries are fixed to zero to suppress weak inter-order interactions. This configuration effectively embeds physical priors and reduces the number of learnable parameters compared to a dense $d \times K'$ matrix. To exploit *inter-component correlations*, the RGB components are jointly predicted, making each block $\mathbf{t}_l$ a square matrix with $P_{\mathbf{t}_l} = [3\,(2l-1)]^2$ learnable parameters. The total number of learnable parameters $P_{\mathbf{T}}$ in $\mathbf{T}$ is the sum of all block parameters:

$$P_{\mathbf{T}} = \sum_{l=2}^{L} P_{\mathbf{t}_l} = \sum_{l=2}^{L} [3\,(2l-1)]^2\,. \tag{34}$$

3) Loss Function: SHAC-LST jointly optimizes the subset matrix $\mathbf{S}$ and the nonzero blocks $\mathbf{t}_l$ of the transformation matrix $\mathbf{T}$ via backpropagation. The training objective consists of the standard 3DGS rendering loss [1] and an additional scale regularization term on $\mathbf{T}$. This regularization is motivated by the scale ambiguity inherent in low-rank factorization:

$$\tilde{\mathbf{A}} = \mathbf{S}\mathbf{T} = (\mathbf{S}'\mathbf{Q})\mathbf{T} = \mathbf{S}(\mathbf{Q}\mathbf{T}'), \tag{35}$$

where scaling either $\mathbf{S}'$ or $\mathbf{T}'$ by $\mathbf{Q}$ yields an identical reconstruction. Without constraints, $\mathbf{T}$ can grow unbounded during training. Unbounded transformation magnitudes are undesirable, as they amplify noise, particularly quantization errors introduced during the compression of $\mathbf{S}$. Consequently, even minor deviations in $\mathbf{S}$ can propagate into significant distortions in the reconstructed $\tilde{\mathbf{A}}$. To ensure numerical stability and reduce error propagation, we introduce a scale regularization term on $\mathbf{T}$ to penalize large transformation magnitudes:

$$\mathcal{L}_{\mathbf{T}} = \sum_{l=2}^{L} \frac{1}{P_{\mathbf{t}_l}} \|\mathbf{t}_l\|_F^2, \tag{36}$$

where $\|\mathbf{t}_l\|_F^2$ denotes the squared Frobenius norm of the block matrix $\mathbf{t}_l$.

The final loss function is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{render}}(\Theta, \mathbf{S}\mathbf{T}) + \lambda_t \mathcal{L}_{\mathbf{T}}, \tag{37}$$

where $\mathcal{L}_{\text{render}}(\Theta, \mathbf{S}\mathbf{T})$ supervises both $\mathbf{S}$ and $\mathbf{T}$ through reconstruction, and Θ denotes the remaining learnable parameters. The scalar λ_t controls the strength of the transformation regularization. Larger values impose stronger constraints but may limit expressiveness. Empirically, $\lambda_t = 5 \times 10^{-3}$ provides a good balance between stability and reconstruction fidelity.

4) Compression Analysis: The storage consists of the original DC coefficients $\mathbf{D}$ ($N \times 3$), the subset SH AC coefficients $\mathbf{S}$ ($N \times d$), and the transformation parameters ($P_{\mathbf{T}}$). The compression ratio (CR) relative to the full $N \times K$ SH coefficients is:

$$\text{CR}_{\text{LST}} = \frac{N(3+d) + P_{\mathbf{T}}}{N \times K}. \tag{38}$$

For $d = 3(2L+1)$ and $K = 3(L+1)^2$, the asymptotic CR for large N simplifies to:

$$\lim_{N\to\infty} \text{CR}_{\text{LST}} = \frac{3+d}{K} = \frac{3+3(2L+1)}{3(L+1)^2} = \frac{2}{L+1}. \tag{39}$$

This $O(1/L)$ compression demonstrates efficient memory savings, especially as the SH degree L increases.

VI. Experiments

A. Datasets and Metrics

The experiments are conducted on 13 diverse scenes from standard 3DGS benchmark datasets (Mip-NeRF 360 [43], Tanks and Temples [44], and Deep Blending [45]). We report PSNR, SSIM [46], and LPIPS (VGG) [47] to assess view synthesis quality, along with model size (MB) for compression analysis.

B. SH AC Coefficients Compaction

We begin by comparing our approach with existing SH AC coefficients compaction methods, which can be broadly categorized into post-hoc and training-integrated strategies. Each method is denoted in the form "{method}-L{SH degree}-D{SH AC dimensions}".

1) Baselines:

- **Anchor** (*Train-L3-D45*): The original, uncompressed 3DGS model trained with full-degree SH AC coefficients ($L = 3$, $K' = 45$), serving as the reference.
- **Post-hoc methods**: These methods are applied to a pre-trained full-degree 3DGS model.
  - *Trunc.-L3-D24*: Post-hoc truncation of higher-degree SH AC coefficients.
  - *Distill.-L3-D24*: Distillation from a full-degree model to a lower-degree SH representation via fine-tuning [4].
  - *SHAC-PCA-L3-D24 (Ours)*: The proposed PCA-based post-hoc compaction method.
- **Training-integrated methods**: These methods incorporate SH compaction directly into the training process.
  - *Train-L2-D24*: Model trained from scratch with a reduced SH degree ($L = 2$, $K' = 24$).
  - *Latent-L3-D21*: Substitutes SH coefficients with a learnable latent vector and a lightweight decoder [7].
  - *SHAC-LST-L3-D21 (Ours)*: The proposed training-integrated low-rank factorization approach.

2) Settings: All methods are implemented based on the original 3DGS [1] framework. For SHAC-LST, both the structured transformation matrix $\mathcal{T}$ and the per-Gaussian subset coefficients $\mathbf{S}$ are jointly optimized using the Adam optimizer [48]. The learning rate for $\mathbf{S}$ follows the default settings for SH AC coefficients. For $\mathcal{T}$, the learning rate is linearly increased to 2.5×10^{-3} during the first 5% of training steps, and then gradually decayed to 2.5×10^{-5} following a cosine schedule. All other hyperparameters remain consistent with those in 3DGS [1]. The *Distill.-L3-D24* baseline is implemented following the publicly released code from [4]. For *Latent-L3-D21*, we employ 21-dimensional latent features and a two-layer MLP (with width twice the output dimension) to decode the full SH AC coefficients coefficients.

TABLE II
QUANTITATIVE COMPARISON WITH VARIOUS SH AC COEFFICIENTS COMPACTION TECHNIQUES ON THREE BENCHMARK DATASETS(MIP-NERF 360, TANKS AND TEMPLES, DEEP BLENDING). EACH METHOD IS LABLED AS "{METHOD}-L{SH DEGREE}-D{SH AC DIMENSIONS.}". OUR METHODS, SHAC-PCA AND SHAC-LST, ACHIEVE HIGHER RENDERING QUALITY WITH LOWER STORAGE. ALL EVALUATIONS ARE BASED ON THE 3DGS [1] FRAMEWORK. BEST AND SECOND-BEST RESULTS ARE HIGHLIGHTED AS 1ST AND 2ND, RESPECTIVELY.

Category	Method	Mip-NeRF360				Tanks & Temples				Deep Blending			
		PSNR↑	SSIM↑	LPIPS↓	Size (MB)↓	PSNR↑	SSIM↑	LPIPS↓	Size (MB)↓	PSNR↑	SSIM↑	LPIPS↓	Size (MB)↓
Anchor	Train-L3-D45	27.53	0.816	0.215	612	23.82	0.853	0.169	354	29.88	0.908	0.238	559
Post-hoc	Trunc.-L3-D24	26.31	0.802	0.227	394	23.10	0.842	0.179	228	29.48	0.906	0.241	360
	Distill.-L3-D24	27.35	0.814	0.219	394	23.63	0.849	0.176	228	29.76	0.908	0.241	360
	SHAC-PCA-L3-D24	27.41	0.814	0.221	394	23.76	0.852	0.173	228	29.79	0.906	0.239	360
Training	Train-L2-D24	27.42	0.815	0.217	397	23.65	0.851	0.172	228	29.90	0.908	0.237	360
	Latent-L3-D21	27.45	0.815	0.217	374	23.71	0.853	0.170	211	29.83	0.907	0.239	334
	SHAC-LST-L3-D21	27.48	0.816	0.216	366	23.92	0.854	0.168	210	29.94	0.908	0.237	333

3) Results: As shown in Tables II, SHAC-PCA achieves the best post-hoc performance, delivering substantial compression with minimal quality degradation. SHAC-LST consistently outperforms other training-integrated methods and often matches or surpasses the anchor. This improvement highlights its structural regularization effect: by explicitly enforcing a low-rank constraint, SHAC-LST constrains the SH space and suppresses redundant or noisy high-frequency components that can cause overfitting in vanilla 3DGS optimization, thereby enhancing overall rendering quality despite using fewer parameters.

C. 3DGS Compression Enhancement

Existing 3DGS compression methods often face a bottleneck in encoding SH AC coefficients, limiting their RD performance. To evaluate the compatibility and practical benefit of our proposed low-rank approaches, we integrate them into two representative 3DGS *Vanilla-GS-based* pipelines.

1) Baselines:

- **gsplat** [19]: A leading Vanilla-GS-based method in the *3DGS.zip* [11] compression benchmark. It encodes SH AC coefficients using vector quantization (VQ). The remaining Gaussian attributes are mapped to 2D grids using PLAS [3], scalar-quantized, and then compressed losslessly with PNG.
- **GSCodec Studio** [22]: A representative video-based vanilla 3DGS compression method serves as the official video-based anchor in the MPEG GSC activity [49]. All Gaussian attributes are mapped to 2D grids using PLAS and compressed with standard video codecs.

2) Settings:

- **gsplat**: Uses the *MCMC* [50] densification strategy to precisely control the number of Gaussians, training models with 0.36M, 0.49M, and 1M Gaussians to obtain multiple rate points. Cluster centers in VQ are quantized to 8 bits, while other settings follow the default configuration in gsplat. SHAC-PCA is applied after vector quantization with an energy retention ratio of 99% (denoted as *gsplat+SHAC-PCA*), while SHAC-LST is integrated into the training process (denoted as *gsplat+SHAC-LST*).
- **GSCodec Studio**: Uses 3DGS models trained by gsplat (1M Gaussians) as input. Positions are quantized to 16 bits and coded losslessly; other attributes are quantized to 10 bits and coded lossily. The mapped 2D grids are compressed with HEVC (HM v18.0 [51]). The QP values for each rate point are summarized in Table III. SHAC-PCA is applied as a preprocessing step before quantization with an energy retention ratio of 99% (denoted as *gscodec+SHAC-PCA*), while SHAC-LST uses the Gaussians trained from *gsplat+SHAC-LST* (denoted as *gscodec+SHAC-LST*).

TABLE III
QUANTIZATION PARAMETER (QP) SETTINGS OF GSCODEC STUDIO FOR EACH GAUSSIAN ATTRIBUTE ACROSS FOUR RATE POINTS.

RP	Opacity	Rotation	Scale	SH DC	SH AC
1	22	17	12	7	22
2	17	7	12	7	17
3	17	2	7	7	12
4	7	2	7	7	7

3) Results: Fig. 4 and Fig. 5 present the RD curves for *gsplat* based and *GSCodec Studio* based methods, respectively. Across all datasets and metrics, SHAC-LST consistently achieves the best RD performance, followed by SHAC-PCA, with both surpassing the baseline. Notably, the performance gain of our methods is driven by the relative bitrate dominance of SH AC coefficients within the compression framework: the larger their dominance, the greater the rate–distortion improvement. Furthermore, as shown in Fig. 6, *gscodec+SHAC-LST* preserves colors and fine details more faithfully, particularly in highlights and textured regions, demonstrating consistent improvements in both objective and perceptual quality.

VII. DEEP ANALYSIS AND ABLATION STUDIES

A. Analysis of 3DGS Compression Enhancement

To position our work within the broader landscape of 3DGS compression, we perform a comprehensive RD comparison with recent methods. To contextualize these results, existing approaches can be broadly categorized into two directions:

- **Vanilla-GS-based methods** (e.g., SOG [3], RDO-Gaussian [6], gsplat [19], GSCodecStudio [22]): These approaches compress vanilla 3DGS representations with low complexity and remain compatible with standard rasterization pipelines. This paradigm prioritizes engineering practicality and widespread deployment.

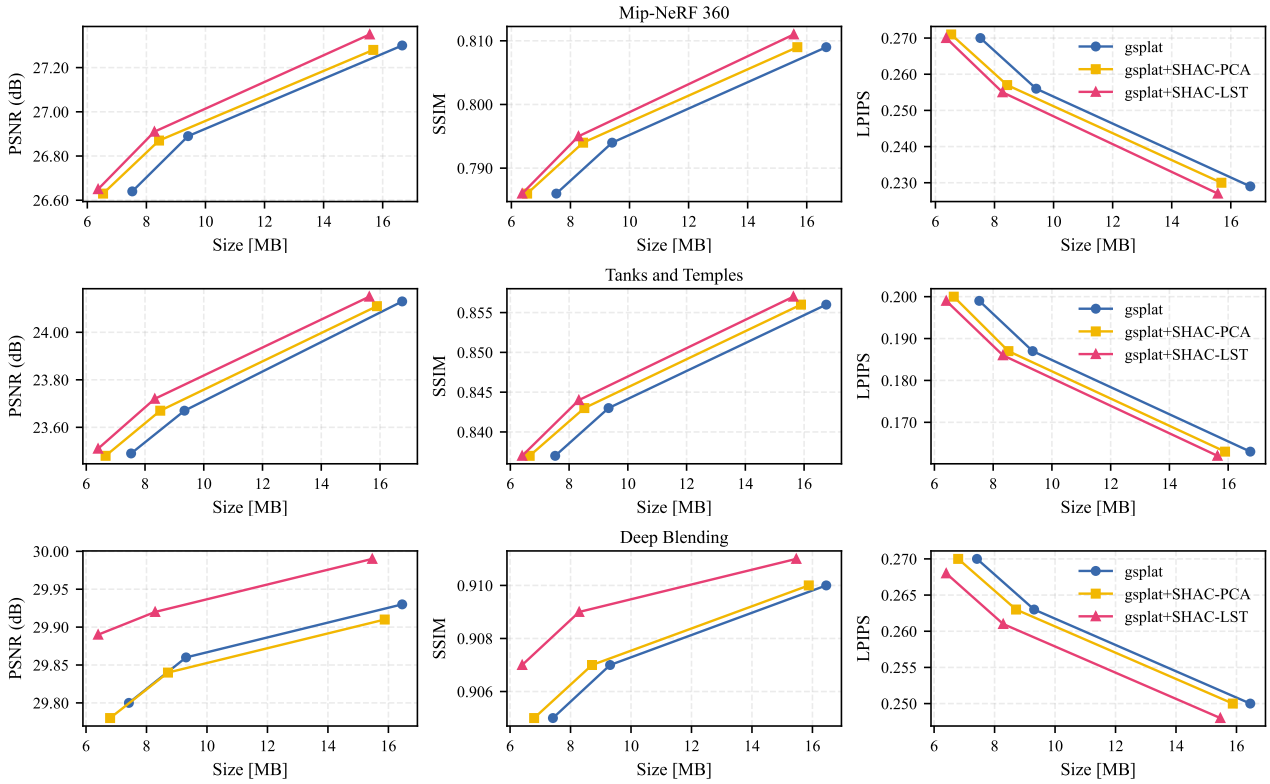

Fig. 4. Comparison of RD performance for *gsplat* based methods across three benchmark datasets (Mip-NeRF 360, Tanks and Temples, Deep Blending). Each row corresponds to a dataset, displaying PSNR, SSIM, and LPIPS metrics from left to right. The results show that *gsplat+SHAC-LST* consistently outperforms *gsplat+SHAC-PCA*, with both methods significantly surpassing the *gsplat* baseline.

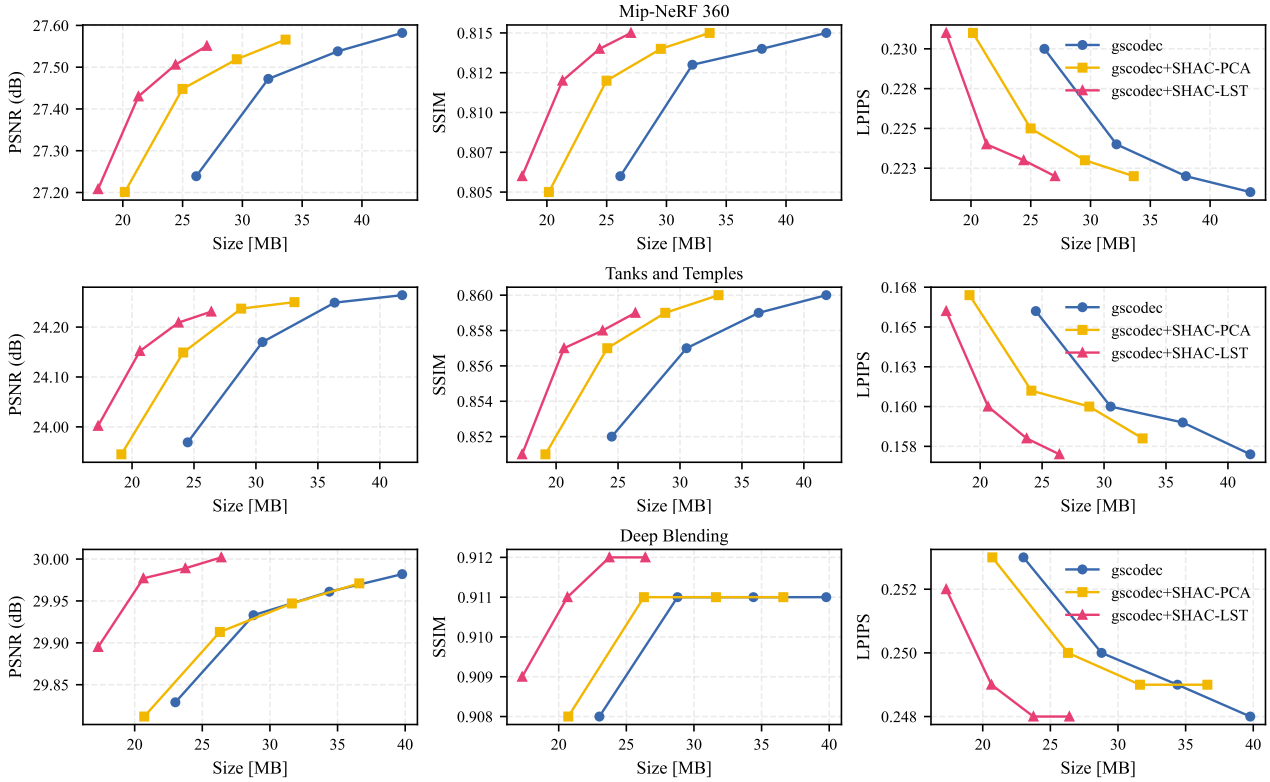

Fig. 5. Comparison of RD performance for *GSCodec Studio* based methods across three benchmark datasets (Mip-NeRF 360, Tanks and Temples, Deep Blending). Each row corresponds to a dataset, displaying PSNR, SSIM, and LPIPS metrics from left to right. The results show that *gscodec+SHAC-LST* consistently outperforms *gscodec+SHAC-PCA*, with both methods significantly surpassing the *GSCodec Studio* baseline.

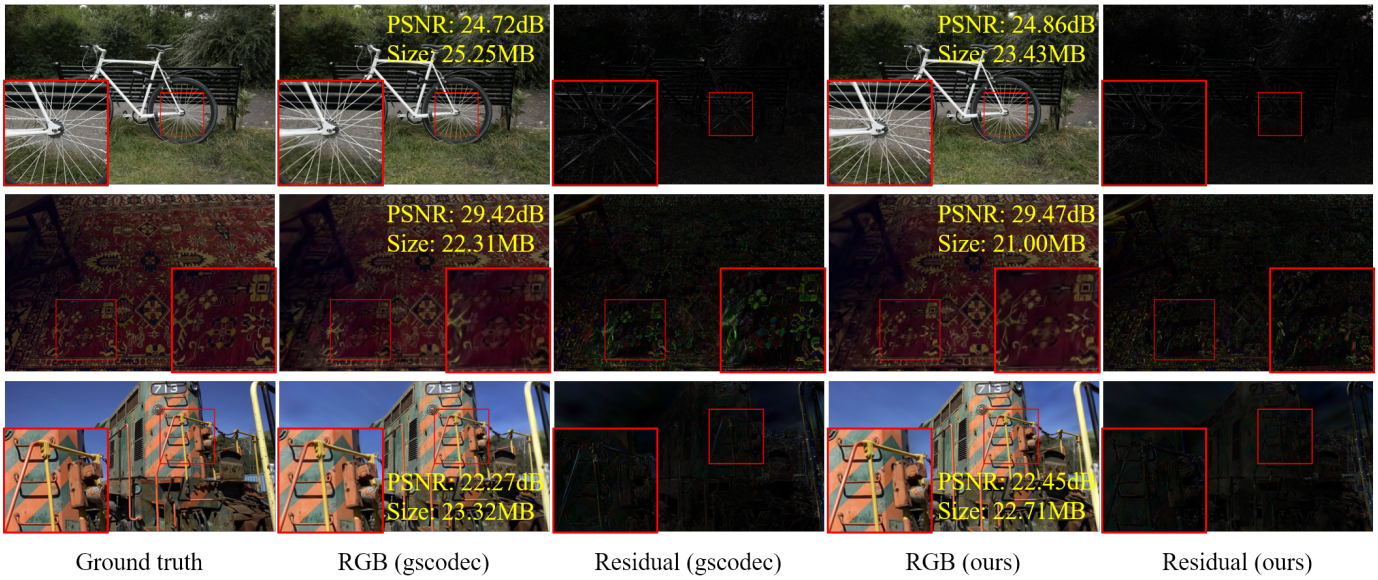

Fig. 6. Qualitative comparison between *gscodec+SHAC-LST* (ours) and the baseline *gscodec* across three scenes: *bicycle* (Mip-NeRF 360), *drjohnson* (Deep Blending), and *train* (Tanks and Temples). Our method achieves superior rendering quality with a lower storage footprint, with residual maps demonstrating the improved preservation of specular highlights and intricate textures.

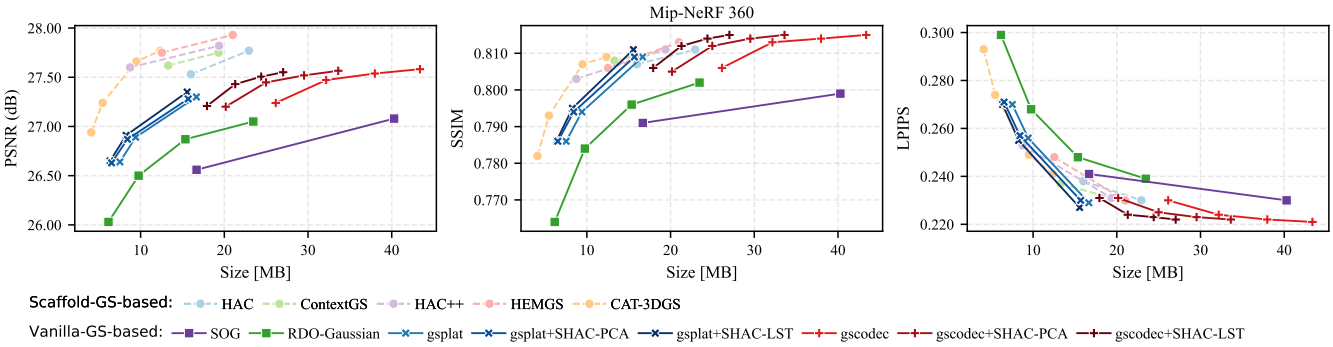

Fig. 7. Comparison of RD performance against various 3DGS compression methods on the Mip-NeRF 360 dataset. Our method significantly enhances *Vanilla-GS-based* baselines as a plug-and-play module. Consequently, our method effectively narrows the performance gap between these two categories—particularly in terms of LPIPS and SSIM—offering a compelling trade-off among efficiency, compatibility, and visual quality.

- **Scaffold-GS-based methods** (e.g., HAC [24], ContextGS [27], HAC++ [26], HEMGS [25], CAT-3DGS [28]): These methods typically extend Scaffold-GS [23] with tailored context models. While these approaches achieve state-of-the-art RD performance through efficient entropy coding, they alter the native 3DGS format. This modification often reduces tool compatibility, degrades rendering efficiency, and increases decoding latency due to the complexity of context modeling.

Fig. 7 provides a comprehensive RD comparison against these recent works. The results demonstrate that while *Scaffold-GS-based* methods currently define the performance upper bound, our method significantly enhances *Vanilla-GS-based* baselines as a plug-and-play module. Consequently, our method effectively narrows the performance gap between these two categories—particularly in terms of LPIPS and SSIM—offering a compelling trade-off among efficiency, compatibility, and visual quality.

B. Generalizability Analysis

The low-rank structure of SH coefficients is intrinsic to the SH representation of natural scenes and is therefore expected to generalize across various datasets and different 3DGS variants.

Beyond large-scale benchmarks, we additionally evaluate our method on both a small-scale dataset (Synthetic-NeRF [52]) and an extremely large-scale dataset (BungeeNeRF [53]). For both datasets, we compress *gsplat*-trained 3DGS models using *GSCodec Studio* as the baseline, following the settings in Sec. VI-C2. In particular, we use 0.36M Gaussians per scene for Synthetic-NeRF and 1M Gaussians per scene for BungeeNeRF. As shown in Fig. 8 and Fig. 9, SHAC-LST consistently achieves the best RD performance, followed by SHAC-PCA, with both clearly outperforming the baseline. These results demonstrate the generality of our method across both small-scale and extremely large-scale scenes.

The low-rank hypothesis can also extend to dynamic scenes and dynamic 3DGS variants [54], [55]. We apply our method

to *Deformable-3DGS* [54] and evaluate it on the D-NeRF [56] dataset, setting the SHAC-PCA energy retention ratio to 96%. As shown in Table IV, our approach achieves consistent compression gains with negligible impact on rendering quality, mirroring trends observed in static 3DGS. These results confirm the generality of our method beyond vanilla 3DGS.

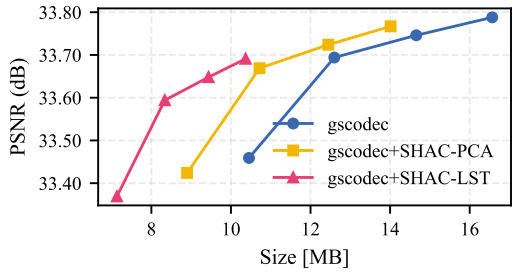

Fig. 8. Generalization of the proposed methods on Synthetic-NeRF, a small-scale dataset.

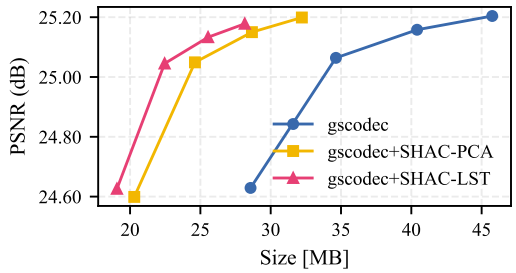

Fig. 9. Generalization of the proposed methods on BungeeNeRF, an extremely large-scale dataset.

TABLE IV
GENERALIZATION OF THE PROPOSED METHODS TO DEFORMABLE-3DGS, EVALUATED ON D-NERF DATASET.

Method	SSIM ↑	PSNR ↑	LPIPS ↓	Size (MB) ↓
Deformable-3DGS	0.991	40.09	0.013	12.7
+ SHAC-PCA	0.991	39.99	0.013	7.7
+ SHAC-LST	0.991	40.04	0.013	7.7

C. Complexity Analysis

We evaluate the time overhead on *truck* (Tanks and Temples) using a single NVIDIA RTX 3090 GPU.

1) Training Overhead of SHAC-LST: SHAC-LST incurs a moderate training overhead due to the additional matrix operations. As reported in Table V, the overhead ranges from approximately 26% to 34% relative to the *gsplat* baseline. This additional cost is considered acceptable and could be further reduced through low-level optimizations, such as CUDA kernel fusion or operator-level acceleration.

2) Encoding and Decoding Speedup of SHAC-LST: SHAC-LST significantly accelerates both encoding and decoding by reducing the SH AC coefficients dimensionality from $O(L^2)$ to $O(L)$. As summarized in Table VI, integrating SHAC-LST with *gsplat* yields a 2–4× speedup in SH AC coefficients processing for both encoding and decoding stages.

TABLE V
TRAINING OVERHEAD OF SHAC-LST ON TRUCK (TANKS AND TEMPLES).

Method	#GS	PSNR↑	SSIM↑	LPIPS↓	Time [min]↓
gsplat	0.36M	25.27	0.875	0.159	6.5
gsplat+SHAC-LST	0.36M	25.29	0.875	0.159	8.2
gsplat	1.00M	26.02	0.891	0.126	11.6
gsplat+SHAC-LST	1.00M	26.07	0.891	0.125	15.6

TABLE VI
ENCODING AND DECODING SPEEDUP OF SHAC-LST ON TRUCK (TANKS AND TEMPLES).

Method	#GS	Enc. [s]	Speedup	Dec. [s]	Speedup
gsplat	0.36M	30.52	-	0.15	-
gsplat+SHAC-LST	0.36M	9.24	3.3×	0.04	3.8×
gsplat	1.00M	85.39	-	0.38	-
gsplat+SHAC-LST	1.00M	46.22	1.8×	0.12	3.2×

D. Ablation for SHAC-PCA

1) Ablation on Including SH DC coefficients: We study the effect of including DC coefficients in PCA. Experiments are conducted on *truck* (Tanks and Temples). Fig. 10 shows that including DC coefficients in PCA causes visible color artifacts. It confirms that isolating DC coefficients preserves base color while allowing PCA to capture AC coefficient structures.

Fig. 10. Ablation study of SHAC-PCA on including DC coefficients in PCA on *truck* (Tanks and Temples).

2) Ablation on Principal Components Counts: We study the effect of principal components counts by varying the energy retention ratio (ER = 95%, 97%, 99%). Experiments are conducted on the Tanks and Temples dataset. We compress *gsplat*-trained 3DGS models (1M Gaussians per scene) using *GSCodec Studio* as the baseline, following the settings in Sec. VI-C2. As shown in Fig. 11, higher ERs are preferred at high bitrates, while lower ERs yield better efficiency at low bitrates.

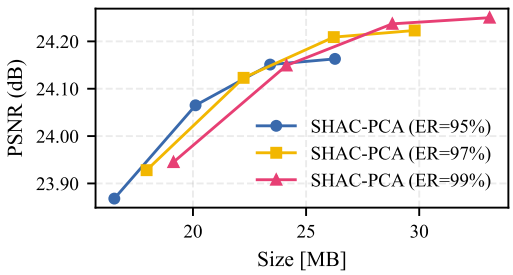

Fig. 11. Ablation study of SHAC-PCA on principal components counts by varying energy retention ratios (ER) on Tanks and Temples dataset.

E. Ablations for SHAC-LST

1) Ablation on Subset Selection Strategy: The proposed strategy (Sec. V-B) selects, for each order m, the coefficient with the lowest degree l. We also evaluate an alternative strategy that selects the highest-degree coefficient for each m. However, this choice violates the energy concentration principle and, as shown in Table VII, performs worse than our proposed strategy.

TABLE VII
ABLATION STUDY OF SHAC-LST ON SUBSET SELECTION STRATEGY ON THE TANKS AND TEMPLES DATASET.

Subset Selection Strategy	PSNR↑	SSIM↑	LPIPS↓
Highest degree l per order m	23.73	0.851	0.170
Lowest degree l per order m **(ours)**	23.92	0.854	0.168

2) Ablation on Subset Dimension: We analyze the effect of subset dimension ($d = (2L+1)^3$) by varying the SH degree ($L = 2, 3, 4$). Experiments are conducted on the Mip-NeRF 360 dataset. We compress *gsplat*-trained 3DGS models (1M Gaussians per scene) using *GSCodec Studio* as the baseline, following the settings in Sec. VI-C2. As shown in Fig. 12, higher subset dimensions are favored at high bitrates, while lower dimensions are sufficient at low bitrates.

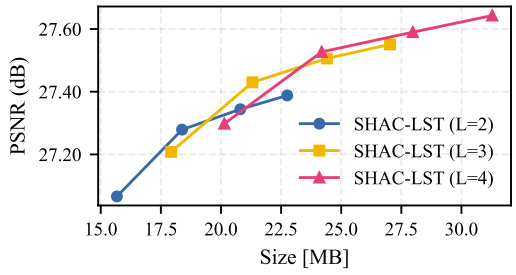

Fig. 12. Ablation study of SHAC-LST on subset dimension by varying SH degree L on Mip-NeRF 360 dataset.

3) Ablation on Scale Regularization Loss: We study the effect of the scale regularization loss on $\mathbf{T}$. Experiments are conducted on the Mip-NeRF 360 dataset, where 3DGS models with 0.36M, 0.49M, and 1.0M Gaussians are trained and encoded using *gsplat*, following the settings in Sec. VI-C2. As shown in Fig. 13, removing the regularization leads to a clear performance degradation, underscoring its role in ensuring numerical stability and reducing error propagation.

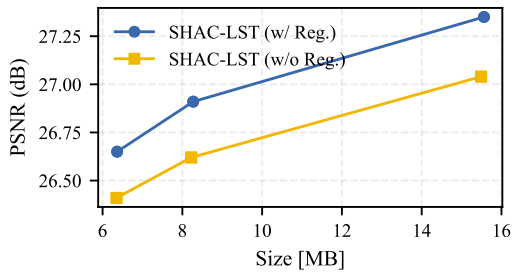

Fig. 13. Ablation study of SHAC-LST on scale regularization loss on Mip-NeRF 360 dataset.

4) Ablation on Transformation Operator: Various operators for mapping the subset $\mathbf{S}$ to the full AC coefficients $\tilde{\mathbf{A}}$ are evaluated, with results summarized in Table VIII.

- **Vector Operator**: Predicts each coefficient independently from subset elements of the same order m. This strict independence prevents information sharing and ignores cross-order correlations, leading to poor reconstruction.
- **Linear Operator**: Uses a single linear layer to learn a global mapping. While the bias term provides slight flexibility, it also introduces unwanted perturbations, resulting in inferior performance.
- **MLP Operator**: Employs a shallow MLP (two hidden layers, with width twice the output dimension) to model non-linear mappings. However, it provides no clear advantage over linear operators, suggesting that SH correlations are predominantly linear.
- **Matrix Operator (ours)**: Adopts a structured matrix composed of small submatrices to explicitly model dominant intra-order and essential cross-order SH correlations. This design achieves the best trade-off between parameter efficiency and reconstruction quality.

TABLE VIII
ABLATION STUDY OF SHAC-LST ON TRANSFORMATION OPERATORS ON TANKS AND TEMPLES DATASET.

Method	PSNR↑	SSIM↑	LPIPS↓
Vector	24.13	0.858	0.159
Linear	24.24	0.860	0.156
MLP	24.23	0.860	0.157
Matrix **(ours)**	24.31	0.862	0.155

5) Ablation on Transformation Matrix Design Choices: The key design choices of the transformation matrix are analyzed, with results summarized in Table IX.

- **Structure (Block-diagonal vs. Dense):** The block-diagonal matrix outperforms a dense matrix by capturing strong intra-order correlations while suppressing weak inter-coefficient interference.
- **Initialization (Zero vs. Identity):** Zero initialization outperforms identity initialization, as it allows the model to learn correlations freely rather than inheriting potentially misleading priors.
- **RGB Processing (Joint vs. Separate):** Jointly processing RGB components yields superior performance compared to separate processing by effectively leveraging inter-component correlations

TABLE IX
ABLATION STUDY OF SHAC-LST ON TRANSFORMATION MATRIX DESIGN CHOICES ON TANKS AND TEMPLES DATASET.

Configuration	PSNR↑	SSIM↑	LPIPS↓
w/o Block-diagonal Structure	24.20	0.859	0.157
w/o RGB Joint Processing	24.15	0.858	0.159
w/o Zero Initialization	24.18	0.859	0.158
Full Model **(ours)**	24.31	0.862	0.155

VIII. Conclusion

The high storage cost of SH coefficients remains a crucial bottleneck for 3DGS compression. In this work, we revealed that SH coefficients exhibit an intrinsic low-rank structure due to the anisotropy of real-world scenes and leveraged this property through two complementary compression approaches. SHAC-PCA retains principal spectral variance to efficiently compress SH AC coefficients. SHAC-LST decomposes the SH AC coefficients into compact, physically motivated subset coefficients and a shared, structured transformation matrix. Beyond strong compression, SHAC-LST can even enhance rendering fidelity via regularization. Comprehensive experiments demonstrate that both methods significantly reduce model size without compromising visual quality, with SHAC-LST achieving superior rate–distortion performance. This work establishes a principled foundation for SH coefficient compression, enabling more compact, efficient 3DGS representations.

IX. Limitations and Future Work

SHAC-LST adopts a fixed subset selection strategy together with a global transformation matrix as a pragmatic trade-off between complexity and efficiency. Nevertheless, in atypical scenarios—such as scenes dominated by sharp specular lighting, extremely sparse view coverage, or strong spatial appearance variations, adaptive subset selection or spatially varying transformations could further improve reconstruction fidelity. Exploring such adaptive mechanisms while controlling training complexity and storage overhead is an interesting direction for future work. Finally, extending the proposed low-rank approximation framework to other 3DGS variants, such as anchor-based 3DGS, is a promising avenue to broaden its applicability.

References

[1] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, 2023.

[2] Y. Bao, T. Ding, J. Huo, Y. Liu, Y. Li, W. Li, Y. Gao, and J. Luo, "3d gaussian splatting: Survey, technologies, challenges, and opportunities," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 35, no. 7, pp. 6832–6852, 2025.

[3] W. Morgenstern, F. Barthel, A. Hilsmann, and P. Eisert, "Compact 3d scene representation via self-organizing gaussian grids," in *European Conference on Computer Vision*. Springer, 2024, pp. 18–34.

[4] Z. Fan, K. Wang, K. Wen, Z. Zhu, D. Xu, and Z. Wang, "Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps," *arXiv preprint arXiv:2311.17245*, 2023.

[5] P. Papantonakis, G. Kopanas, B. Kerbl, A. Lanvin, and G. Drettakis, "Reducing the memory footprint of 3d gaussian splatting," *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, vol. 7, no. 1, pp. 1–17, 2024.

[6] H. Wang, H. Zhu, T. He, R. Feng, J. Deng, J. Bian, and Z. Chen, "End-to-end rate-distortion optimized 3d gaussian representation," in *European Conference on Computer Vision*. Springer, 2024, pp. 76–92.

[7] S. Girish, K. Gupta, and A. Shrivastava, "Eagles: Efficient accelerated 3d gaussians with lightweight encodings," *arXiv preprint arXiv:2312.04564*, 2023.

[8] J. C. Lee, D. Rho, X. Sun, J. H. Ko, and E. Park, "Compact 3d gaussian representation for radiance field," *arXiv preprint arXiv:2311.13681*, 2023.

[9] X. Sun, J. C. Lee, D. Rho, J. H. Ko, U. Ali, and E. Park, "F-3dgs: Factorized coordinates and representations for 3d gaussian splatting," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 7957–7965.

[10] Q. Yang, L. Yang, G. Van Der Auwera, and Z. Li, "Hybridgs: High-efficiency gaussian splatting data compression using dual-channel sparse representation and point cloud encoder," *arXiv preprint arXiv:2505.01938*, 2025.

[11] M. T. Bagdasarian, P. Knoll, Y.-H. Li, F. Barthel, A. Hilsmann, P. Eisert, and W. Morgenstern, "3dgs. zip: A survey on 3d gaussian splatting compression methods," *arXiv preprint arXiv:2407.09510*, 2024.

[12] M. S. Ali, C. Zhang, M. Cagnazzo, G. Valenzise, E. Tartaglione, and S.-H. Bae, "Compression in 3d gaussian splatting: A survey of methods, trends, and future directions," *arXiv preprint arXiv:2502.19457*, 2025.

[13] WG 02 MPEG Technical Requirements, "Draft gaussian splat coding requirements," ISO/IEC JTC1/SC29/WG02 MPEG Technical Requirements, Geneva, Working Draft MDS25816_WG02_N00487, Oct. 2025, meeting 152.

[14] WG 04 MPEG Video Coding, "Common test conditions for gaussian splat coding," ISO/IEC JTC1/SC29/WG04 MPEG Video Coding, Geneva, Explorations Document MDS25782_WG04_N00751, Oct. 2025, meeting 152.

[15] S. Xie, W. Zhang, C. Tang, Y. Bai, R. Lu, S. Ge, and Z. Wang, "Mesongs: Post-training compression of 3d gaussians via efficient attribute transformation," in *European Conference on Computer Vision*. Springer, 2024.

[16] M. S. Ali, M. Qamar, S.-H. Bae, and E. Tartaglione, "Trimming the fat: Efficient compression of 3d gaussian splats through pruning," *arXiv preprint arXiv.2406.18214*, 2024.

[17] Y. Wang, M. Liu, Q. Yang, H. Huang, L. Yang, and Y. Xu, "On the efficient adaptive streaming of 3d gaussian splatting over dynamic networks," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2025.

[18] S. Niedermayr, J. Stumpfegger, and R. Westermann, "Compressed 3d gaussian splatting for accelerated novel view synthesis," *arXiv preprint arXiv:2401.02436*, 2024.

[19] V. Ye, R. Li, J. Kerr, M. Turkulainen, B. Yi, Z. Pan, O. Seiskari, J. Ye, J. Hu, M. Tancik *et al.*, "gsplat: An open-source library for gaussian splatting," *Journal of Machine Learning Research*, vol. 26, no. 34, pp. 1–17, 2025.

[20] C. Wang, S. N. Sridhara, E. Pavez, A. Ortega, and C. Chang, "Adaptive Voxelization for Transform Coding of 3D Gaussian Splatting Data," *arXiv preprint arXiv:2506.00271*, 2025.

[21] P. Wang, Z. Zhang, L. Wang, K. Yao, S. Xie, J. Yu, M. Wu, and L. Xu, "V^ 3: Viewing volumetric videos on mobiles via streamable 2d dynamic gaussians," *ACM Transactions on Graphics (TOG)*, vol. 43, no. 6, pp. 1–13, 2024.

[22] S. Li, C. Wu, H. Li, X. Gao, Y. Liao, and L. Yu, "Gscodec studio: A modular framework for gaussian splat compression," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2026.

[23] 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*, 2024, pp. 20 654–20 664.

[24] Y. Chen, Q. Wu, W. Lin, M. Harandi, and J. Cai, "Hac: Hash-grid assisted context for 3d gaussian splatting compression," in *European Conference on Computer Vision*. Springer, 2024.

[25] L. Liu, Z. Chen, W. Jiang, W. Wang, and D. Xu, "Hemgs: A hybrid entropy model for 3d gaussian splatting data compression," *arXiv preprint arXiv:2411.18473*, 2024.

[26] Y. Chen, Q. Wu, W. Lin, M. Harandi, and J. Cai, "Hac++: Towards 100x compression of 3d gaussian splatting," *arXiv preprint arXiv:2501.12255*, 2025.

[27] Y. Wang, Z. Li, L. Guo, W. Yang, A. Kot, and B. Wen, "Contextgs: Compact 3d gaussian splatting with anchor level context model," *Advances in neural information processing systems*, vol. 37, pp. 51 532–51 551, 2024.

[28] Y.-T. Zhan, C.-Y. Ho, H. Yang, Y.-H. Chen, J. C. Chiang, Y.-L. Liu, and W.-H. Peng, "Cat-3dgs: A context-adaptive triplane approach to rate-distortion-optimized 3dgs compression," *arXiv preprint arXiv:2503.00357*, 2025.

[29] A. M. Rufai, G. Anbarjafari, and H. Demirel, "Lossy image compression using singular value decomposition and wavelet difference reduction," *Digital signal processing*, vol. 24, pp. 117–123, 2014.

[30] X. Jiang, M. Le Pendu, R. A. Farrugia, and C. Guillemot, "Light field compression with homography-based low-rank approximation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 7, pp. 1132–1145, 2017.

[31] Y. Wang, L. Wu, X. Lin, and J. Gao, "Multiview spectral clustering via structured low-rank matrix factorization," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 10, pp. 4833–4843, 2018.

[32] Z. Xue, J. Du, D. Du, and S. Lyu, "Deep low-rank subspace ensemble for multi-view clustering," *Information Sciences*, vol. 482, pp. 210–227, 2019.
[33] K. Sarkar, F. Bernard, K. Varanasi, C. Theobalt, and D. Stricker, "Structured low-rank matrix factorization for point-cloud denoising," in *2018 International conference on 3D vision (3DV)*. IEEE, 2018, pp. 444–453.
[34] X. Lu, S. Schaefer, J. Luo, L. Ma, and Y. He, "Low rank matrix approximation for 3d geometry filtering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 4, pp. 1835–1847, 2022.
[35] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Speeding up convolutional neural networks with low rank expansions," *arXiv preprint arXiv:1405.3866*, 2014.
[36] D. Povey, G. Cheng, Y. Wang, K. Li, H. Xu, M. Yarmohammadi, and S. Khudanpur, "Semi-orthogonal low-rank matrix factorization for deep neural networks." in *Interspeech*, 2018, pp. 3743–3747.
[37] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=nZeVKeeFYf9
[38] R. Karimi Mahabadi, J. Henderson, and S. Ruder, "Compacter: Efficient low-rank hypercomplex adapter layers," *Advances in Neural Information Processing Systems*, vol. 34, pp. 1022–1035, 2021.
[39] S.-Y. Liu, C.-Y. Wang, H. Yin, P. Molchanov, Y.-C. F. Wang, K.-T. Cheng, and M.-H. Chen, "Dora: Weight-decomposed low-rank adaptation," in *Forty-first International Conference on Machine Learning*, 2024.
[40] R. Green, "Spherical harmonic lighting: The gritty details," Sony Computer Entertainment Europe (SCEE) R&D (often associated, though presented at GDC), Tech. Rep., 2003, based on GDC 2003 Presentation.
[41] D. Marinucci and G. Peccati, *Random fields on the sphere: representation, limit theorems and cosmological applications*. Cambridge University Press, 2011, vol. 389.
[42] I. J. Schoenberg, "Positive definite functions on spheres," *Duke Mathematical Journal*, vol. 9, no. 1, pp. 96–108, 1942.
[43] 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*, 2022, pp. 5470–5479.
[44] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun, "Tanks and temples: Benchmarking large-scale scene reconstruction," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, pp. 1–13, 2017.
[45] P. Hedman, J. Philip, T. Price, J.-M. Frahm, G. Drettakis, and G. Brostow, "Deep blending for free-viewpoint image-based rendering," *ACM Transactions on Graphics (ToG)*, vol. 37, no. 6, pp. 1–15, 2018.
[46] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
[47] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.
[48] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc 3rd Int Conf on Learning Representations*, 2015.
[49] WG 04 MPEG Video Coding, "Jee 6.2 on anchor generation," ISO/IEC JTC1/SC29/WG04 MPEG Video Coding, Geneva, Explorations Document MDS25784_WG04_N00753, Oct. 2025, meeting 152.
[50] S. Kheradmand, D. Rebain, G. Sharma, W. Sun, Y.-C. Tseng, H. Isack, A. Kar, A. Tagliasacchi, and K. M. Yi, "3d gaussian splatting as markov chain monte carlo," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024, spotlight Presentation.
[51] J. V. E. T. (JVET), "Hm-18.0: Hevc reference software," 2023. [Online]. Available: https://vcgit.hhi.fraunhofer.de/jvet/HM/-/tree/HM-18.0
[52] 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*, vol. 65, no. 1, pp. 99–106, 2021.
[53] Y. Xiangli, L. Xu, X. Pan, N. Zhao, A. Rao, C. Theobalt, B. Dai, and D. Lin, "Bungeenerf: Progressive neural radiance field for extreme multi-scale scene rendering," in *European conference on computer vision*. Springer, 2022, pp. 106–122.
[54] Z. Yang, X. Gao, W. Zhou, S. Jiao, Y. Zhang, and X. Jin, "Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 20 331–20 341.
[55] Z. Guo, W. Zhou, L. Li, M. Wang, and H. Li, "Motion-aware 3d gaussian splatting for efficient dynamic scene reconstruction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 35, no. 4, pp. 3119–3133, 2025.
[56] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 10 318–10 327.

Zhiwei Zhu received the B.Eng. degree in electronic information engineering from Tianjin University, Tianjin, China, in 2021. He is now pursuing a Ph.D. at Zhejiang University, Hangzhou, China. His research interests include 3D computer vision and visual data compression.

Sicheng Li received the B.Eng. degree in electronic information engineering from Hunan University, Changsha, China, in 2019. He is now pursuing a Ph.D. at Zhejiang University, Hangzhou, China. His research interests include 3D computer vision, visual data compression, and immersive video coding.

Yiyi Liao is an assistant professor at Zhejiang University, leading the X-Dimensional Representations Lab. She received her Ph.D. in Control Science and Engineering from Zhejiang University in June 2018 and her B.S. degree from Xi'an Jiaotong University in 2013. Her research interests include 3D computer vision and immersive media compression.

Lu Yu (Senior Member, IEEE) received the B.Eng. degree in radio engineering and the Ph.D. degree in communication and electronic systems from Zhejiang University. She is currently a Distinguished Professor with Zhejiang University. She authored more than 150 peer-reviewed papers on image and video coding, quality assessment, and hardware architecture design, and contributed more than 100 adopted technical proposals to define a series of IEEE, ISO/IEC, as well as ISO/IEC and ITU-T joint standards. She is the Convenor of ISO/IEC JTC 1/SC 29/WG 4 MPEG Video Coding, one of the voting members of the Standard Activities Sub Division of IEEE Circuits and Systems Society, and on the Editorial Board of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.