arXiv:2601.18475v1 [cs.GR] 26 Jan 2026

用于流式视频重建的LoD结构化3D高斯泼溅

Xinhui Liu， Can Wang、Lei Liu、Zhenghao Chen、Wei Jiang、Wei Wang、Dong Xu，*IEE*会士 *E*

摘要—**自由视点视频（FVV）重建能够实现逼真且交互式的3D场景可视化；然而，实时流传输常受稀疏视角输入、高昂的训练成本以及带宽限制所制约。尽管近期的3D高斯泼溅（3DGS）凭借卓越的渲染速度推动了FVV的发展，流式自由视点视频（SFVV）仍提出了额外要求，包括快速优化、在稀疏约束下实现高保真重建，以及尽可能减少存储占用。为弥合这一差距，我们提出了StreamLoD-GS，这是一种专为SFVV设计的基于LoD的高斯泼溅框架。我们的方法融合了三项核心创新：1）基于锚点和八叉树的LoD结构化3DGS，并采用分层高斯丢弃技术，以确保高效、稳定的优化，同时保持高质量渲染；2）基于GMM的运动划分机制，将动态和静态内容分离，在细化动态区域的同时保持背景稳定性；3）量化残差细化框架，在不损害视觉保真度的前提下显著降低存储需求。大量实验表明，StreamLoD-GS在质量、效率和存储方面达到了具有竞争力或最先进的性能。**

索引词—**3DGS、稀疏视角、自由视点视频、视频流传输。**

I. 引言

CONSTRUCTING 基于多视图图像的自由视点视频（FVV）有望通过弥合我们对动态3D世界的感知与传统2D视频的局限之间的鸿沟，彻底革新视觉媒体 [2]。近年来，3D高斯泼溅（3DGS） [3] 已成为自由视点视频重建 [2], [4]–[9],的基石，这主要得益于其实现照片级真实感和实时合成的能力。随着对实时交互性和沉浸式体验的需求不断增长，流式自由视点视频（SFVV） [10]–[15]已成为研究重点，为用户提供超越传统媒体边界的动态、实时观看体验。然而，SFVV也为3DGS带来了新的挑战和要求，包括：1）快速优化和渲染：能够实时快速优化并渲染新视角；2）在稀疏视图下仍保持高保真度：即使仅有有限的相机配置，也要确保所渲染场景的视觉质量；因为在现实环境中采集和传输稠密视角通常在物流和经济方面都难以承受；3）较低的传输存储需求（带宽）：减少3DGS点的存储量，以优化带宽使用。

为应对这些挑战，3DGStream [10] 率先将 3DGS 扩展到 SFVV，并借助 Instant-NGP [16],（结合神经网络表示与多分辨率哈希）加速数据查询和渲染。除表示方法外，其他当代方法 [11]–[13], [15] 已将重点转向运动场建模，利用残差学习 [17] 分离动态区域并降低优化成本。尽管取得了这些进展，大多数基于 3DGS 的框架 [10]–[13]仍依赖稠密、同步的相机阵列来提供足够的视角，从而实现高质量重建。然而在实际应用中，带宽限制和空间限制等因素往往使系统只能使用少量相机，从而阻碍其在真实场景中的应用。在稀疏训练视图下进行 SFVV 重建时，会面临两个主要挑战。首先，随着输入视角数量减少（*e.g.*、7、5、4 个或更少），多视图一致性减弱，导致模型对观测视角过拟合，并在新视图合成中产生伪影。其次，稀疏输入常会在致密化过程中 [18], [19], 导致生成冗余或位置不当的高斯，这会增加存储和传输开销；对于内存和带宽有限的移动设备及低端设备而言，这一问题尤为突出。

受细节层次（LoD）层级 [20]–[24], 的启发，这些层级能够有效减少几何冗余并提升渲染效率，我们提出了 StreamLoD-GS，这是一种专为 SFVV 设计的基于 LoD 的高斯泼溅框架。我们的方法包含三项核心创新：首先，我们提出了一种采用锚点和八叉树表示的 LoD 结构化 3DGS。为防止过拟合并促进稳定优化，我们引入了分层高斯丢弃技术。该策略根据观察距离动态选择细节层次，并在训练期间采用感知层级的随机丢弃。最终，这种设计缓解了稀疏视角过拟合，同时保留了稳健收敛所必需的梯度流。其次，利用视频序列固有的时间连贯性，我们开发了一种基于高斯混合模型 [25] （GMM）的运动分区机制。该模块能够智能识别需要进行选择性细化的动态锚点，同时冻结静态区域，从而利用背景稳定性和运动连续性，大幅减少冗余计算。最后，为实现高带宽效率的流式传输，我们

X. Liu、C. Wang 和 L. Liu 任职于中国香港香港大学计算与数据科学学院。电子邮箱：(xhliu01, canwang, and liulei95)@hku.hk。W. Jiang 和 W. Wang 任职于美国加利福尼亚州圣克拉拉的 Futurewei Technologies Inc。电子邮箱：(wjiang, rick weiwang)@futurewei.com。Z. Chen 任职于澳大利亚新南威尔士州纽卡斯尔大学计算机与信息科学学院。电子邮箱：zhenghao.chen@newcastle.edu.au。D. Xu 任职于中国香港香港大学计算与数据科学学院。电子邮箱：dongxu@hku.hk。通讯作者。

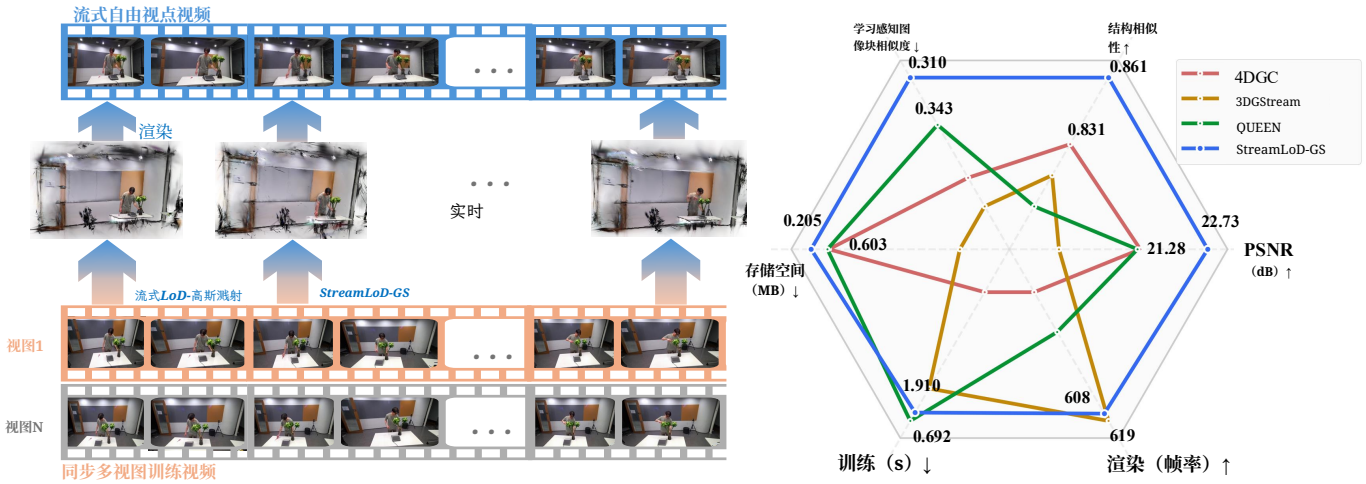


图1. StreamLoD-GS 可为流式自由视点视频实现快速、实时的高保真重建。**右侧**：在 MeetRoom [1] 数据集上，与现有方法使用 5 个训练视图进行比较。

引入了量化残差细化框架。该设计通过量化残差连接静态冻结锚点与动态更新区域，在不牺牲视觉保真度的前提下，大幅减少存储和传输开销。

我们的主要贡献总结如下。

- 我们提出了一种基于锚点和八叉树的LoD结构化3DGS表示，并结合层次化高斯随机失活技术，以稳定训练并确保稀疏视角 SFVV 中的高保真合成。
- 我们引入了一种 GMM 驱动机制，将动态内容与静态内容解耦，从而能够有针对性地细化运动区域，同时保持背景的结构稳定性。
- 我们开发了一种量化残差细化框架，用于压缩动态更新，从而在低带宽环境中实现高效的数据传输和存储。
- 大量实验表明，与当前的 SFVV 方法相比，StreamLoD-GS 在质量、效率和存储占用方面达到了有竞争力或最先进的性能。

II. 相关工作

A. 流式 3DGS

与基于完整多视图视频进行离线学习相比，通过即时逐帧训练构建可流式传输的动态场景面临着巨大挑战。早期研究尝试将神经辐射场 [16], [26]–[30] 应用于这一任务。例如，StreamRF [1] 采用增量学习范式，通过建模逐帧差异进行学习；而 ReRF [17]则对相邻帧之间的残差信息进行建模，以处理长时段场景。另一方面，NeRFPlayer [31], 将四维时空分解，以优化模型紧凑性和重建速度。3DGS 的出现标志着重大进步，其训练效率和渲染速度均更胜一筹。在此基础上，流式 3DGS 旨在通过实时逐帧训练重建动态场景，从而能够随着场景演变实时更新和调整 [7], [10]–[13], [15], [32]。3DGStream [10]提出了结合多分辨率哈希的神经变换缓存，以推进 3DGS 建模。其他方法则侧重于先进的运动建模，例如 HiCoM [11] 提出的分层相干运动机制，以及 IGS [13], 采用锚点驱动网络，通过单步学习运动残差来加快训练。为增强模型表达能力并应对数据量问题，QUEEN [12] 提出了一种框架，将高斯残差与学习式量化和稀疏性相结合，以实现压缩。为优化 SFVV 的率失真权衡，4DGC [15]引入了一种具有运动网格表示和可微量化的率感知压缩框架。尽管保真度和效率有所提升，但这些方法仍存在一个共同瓶颈：进行三维重建时，每一帧都需要稠密多视图图像。这种对同步多摄像机设置的依赖，显著限制了其实际应用。此外，这些方法通常需要大量高斯基元来建模场景，导致存储需求大幅增加。相比之下，我们的方法不仅能够快速优化和渲染，还能在稀疏视图下实现高保真度，同时保持较低的存储需求。

B. 稀疏视角下的新视角合成

尽管神经辐射场 [26]和 3DGS [3] 在稠密图像下都能实现出色的渲染质量，但由于过拟合，它们在稀疏视图下的性能会显著下降。这推动了大量旨在提升其少样本性能的研究。1）稀疏视角神经辐射场 [33]–[37]。在 3DGS 出现之前，NeRF 社区率先提出了多种解决这一问题的方法。这些策略包括利用预训练图像编码器从外部先验中学习（pixelNeRF） [33] ，或[34]利用 CLIP 嵌入设计语义一致性损失 [38], ，通过基于图块的外观约束和深度正则化来增强几何一致性（RegNeRF [35],

SparseNeRF [36]），以及通过频率正则化惩罚高频噪声、提升泛化能力（FreeNeRF） [37]。2）稀疏视图 3DGS [18], [19], [39]–[43]。近年来，针对 3DGS 框架提出了多种创新方法。深度正则化：一类工作侧重于利用可学习参数（Depth-GS） [39] 或局部深度归一化（DN-Gaussian） [40]来消除深度歧义。一致性与致密化：其他方法利用光流（CoherentGS） [41] 或模型分歧（CoR-GS） [42], 来提升多视图一致性；还有一些方法通过“高斯反池化”（FSGS） [43]自适应地对稀疏区域进行致密化。随机正则化：受随机失活启发，第三类方法在训练期间随机移除高斯，以有效缓解过拟合（DropGaussian [18], 、DropoutGS [19]）。与上述方法不同，我们的方法从表示层着手解决稀疏视角问题，并提出一种 LoD 结构化 3DGS，即使在稀疏视图下也能实现稳健重建。

C. 细节层次（LoD）

在3D渲染中，LoD是指根据3D模型与相机之间的距离 [44], [45]调整其细节层次。近期已有多项研究 [21], [24], [46]–[48]探索了3D图形系统（3DGS）中的 LoD。LetsGo [46]将LoD引入为一种用于大规模场景生成的多分辨率高斯表示。CityGaussian [47] 提出了分块式LoD，在渲染过程中，同一区块内的所有高斯共享相同的细节层次，该层次由区块与相机之间的距离决定。类似地，Hierarchical-GS [48] 将大规模场景划分为不同区块，并为每个区块构建LoD，从而支持层次结构的生成与整合，以实现快速渲染。近期提出的一种更具可扩展性的方法是Octree-GS [21],，它采用具有累积式LoD策略的统一八叉树，以消除高斯冗余。不同于这些方法，我们提出了一种基于锚点和八叉树的LoD结构化3DGS表示，并结合分层高斯丢弃技术以防止过拟合。此外，我们的方法专门解决了以流式方式进行高斯存储这一关键挑战，而这些方法均未对此加以考虑。

III. 预备知识

LoD-GS [22] 通过将三维高斯组织成分层结构，实现细节层次（LoD）渲染。每一层 l包含具有位置 $\boldsymbol{\mu}_l$、协方差矩阵 $\boldsymbol{\Sigma}_l$（分解为尺度 $\boldsymbol{s}_l$ 和四元数 $\boldsymbol{q}_l$）、不透明度 $\boldsymbol{\alpha}_l$ 以及球谐系数 $\boldsymbol{c}_l$的高斯 $\mathcal{G}_l$，从而构成多分辨率结构：

$$\mathcal{G} = \{\mathcal{G}_0, \mathcal{G}_1, \ldots, \mathcal{G}_{L-1}\}, \quad |\mathcal{G}_l| < |\mathcal{G}_{l+1}| \tag{1}$$

较粗的层捕捉全局结构，较细的层则编码细节。渲染时，根据观察距离选择活动层 l^* d:

$$l^* = \min\left(\left\lfloor \log_\beta \frac{d}{d_0} \right\rfloor, L\right) \tag{2}$$

其中 d_0 为基准距离， $\beta > 1$ 控制过渡速率。只有活动层中的高斯会通过基于图块的渲染进行光栅化 [3], ，从而降低远距离视角下的计算成本，同时通过距离感知训练和渐进式稠密化保持视觉质量。

Scaffold-GS [49] 提出了一种基于锚点的结构化三维高斯表示，在保持高质量渲染的同时减少冗余高斯。该方法首先根据运动恢复结构 [50] （SfM）生成的点构建稀疏体素网格，并在每个体素中心放置一个锚点。每个锚点与一个可学习特征$\mathbf{f}$ 及 K 个相邻神经高斯相关联，这些神经高斯通过偏移量由锚点生成：

$$\{\boldsymbol{\mu}_0, \ldots, \boldsymbol{\mu}_{K-1}\} = \boldsymbol{\mu}_{\text{anchor}} + \{\mathbf{O}_0, \ldots, \mathbf{O}_{K-1}\} \tag{3}$$

其中， $\boldsymbol{\mu}_{\text{anchor}}$ 是锚点中心，$\mathbf{O}_i$ 是可学习偏移量。其他高斯属性则通过各自的多层感知机直接根据锚点特征$\mathbf{f}$、相对观察距离 Δ_c （锚点中心与相机位置之间的距离）以及$\vec{}$方向 d_c （从相机指向锚点的向量）解码，分别记为 F_α、F_c、F_q、F_s。例如，不透明度为：

$$\{\boldsymbol{\alpha}_0, \ldots, \boldsymbol{\alpha}_{K-1}\} = \mathrm{F}_\alpha(\hat{\mathbf{f}}, \Delta_c, \vec{d_c}) \tag{4}$$

IV. 方法

给定由 N同步相机 $\{\mathbf{I}_t^{(i)}\}_{i=1}^N$ 拍摄的视频序列，其中 $t = 0, 1, \ldots, T$，我们的目标是以流式方式重建视频场景。为此，我们提出了 StreamLoD-GS，这是一种分层 LoD 结构化高斯框架，如图 2 所示。首先，我们利用首帧 $\{\mathbf{I}_{t=0}^{(i)}\}_{i=1}^N$ 的多视图图像，初始化基于锚点和八叉树的 LoD 结构化 3DGS 表示（第 IV-A 节）；接着，我们设计了一种基于高斯混合模型的运动分区方法，用于识别动态锚点以进行选择性建模，同时冻结静态区域，以确保背景稳定性并降低训练成本（第 IV-B 节）；最后，我们提出了一种量化残差细化方法，对动态锚点的残差进行量化，以降低存储成本（第 IV-C 节）。

A. 带锚点和八叉树的 LoD 结构化 3DGS

收到首帧的多视图图像 $\{\mathbf{I}_{t=0}^{(i)}\}_{i=1}^N$ 后，我们首先使用这些图像初始化 LoD 结构化表示。随后，我们应用分层高斯丢弃，逐步丢弃每一层中的高斯，以防止过拟合。

锚点与 LoD 初始化。标准的点云初始化通常采用基于 COLMAP 的 SfM [51]; 然而，在稀疏视角约束下，其可靠性会降低。为解决这一问题，我们利用 VGGT [52] 从首帧的多视图图像 {估计初始点云 $\mathcal{P}\,\mathbf{I}_{t=0}^{(i)}\}_{i=1}^N$，为后续优化提供更准确的起点。关键的是，为确保实验比较的公平性，我们在第 V 节评估的所有方法中采用这一一致的初始化方式。在获得

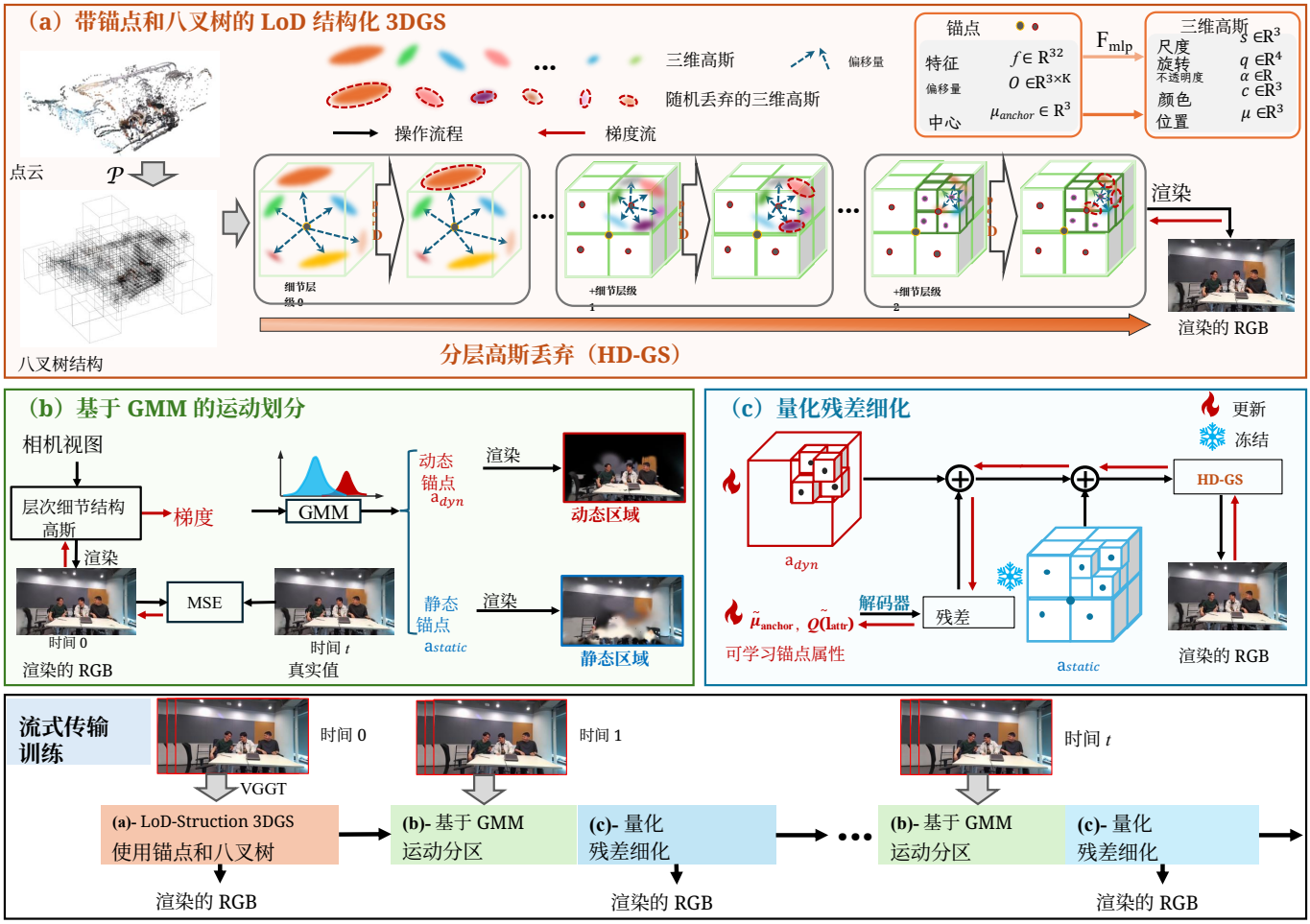


图2. 我们提出的 StreamLoD-GS 示意图。它由三个主要部分组成：**(a)** 带锚点和八叉树的 LoD 结构化 3DGS；**(b)** 基于 GMM 的运动划分；**(c)** 量化残差细化。**下方：** 流式训练流程。帧按顺序处理：初始帧（时间 0）进行 LoD 结构化 3DGS 优化，后续帧则应用运动分区和残差细化，以确保整个流中的 RGB 渲染在时间上保持连贯。

$\mathcal{P}$后，我们初始化高斯中心，并按照 Octree-GS [53]将其组织成具有 L层 LoD 层级的八叉树结构。这里的 L 由场景尺度边界（$d_{\max}$，$d_{\min}$）确定，计算为 $L = \lfloor \log_2(d_{\max}/d_{\min}) \rceil + 1$，其中$\lfloor . \rceil$ 表示舍入算子。我们不直接在每个体素中放置锚点，而是采用逐层放置。在每个层级 $l \in \{0, \ldots, L-1\}$，我们都会在大小为 $\delta_l = \delta/2^l$的体素中放置一个锚点，其中 δ 是最粗层级（第 0 层）的基础体素尺寸。随后，与 Scaffold-GS [49],类似，每个锚点都会被分配一个特征 $\mathbf{f}$，并与 K个神经高斯相关联，每个神经高斯都具有用于高斯位置的可学习偏移量。

为进一步降低计算开销，我们采用动态锚点选择策略 [21]，在渲染过程中选出最重要的锚点。对于任意锚点$\mathbf{a}_j$，其原始层索引可计算为 $L_j = \lfloor \log_2 \left(\frac{d_{\max}}{\Delta_{c_j}} \right) \rceil$，其中 Δc_j 表示锚点中心与相机位置之间的距离。在训练过程中，我们记录其相对于二维屏幕空间投影位置的梯度，记为$\mathbf{g}^{\mu}_{j}$。然后

其新的层索引可通过以下方式更新：

$$L_j^* = \lfloor \log_2 \left(\frac{d_{\max}}{\Delta_{c_j}} \right) \rceil + \Delta L, if \ \mathbf{g}_{\mu_j} > \mathbf{g}_{\text{threshold}} \tag{5}$$

其中 $\mathbf{g}_{\text{threshold}}$ 是一个阈值。在渐进式训练期间，我们将最大层级限制为 $L_{\max}$。当满足以下条件时，选取一个锚点进行渲染：

$$L_j \leq \min(\lfloor L_j^* \rceil, L_{\max}) \tag{6}$$

该函数表明，如果锚点移动到更远的粗化层，则认为它不重要。因此，我们会将其从渲染过程中移除。

分层高斯丢弃。 为缓解训练视点有限所导致的过拟合并稳定优化过程，我们通过引入层级感知丢弃策略，对 DropGaussian [18] 进行了改进。由于更高的细节层次包含更密集的高斯，我们采用逐步增加的丢弃率，其定义为 $r_m^{(l)} = \gamma^{(l)} \cdot \frac{m}{M}$，其中 m 为当前训练步，M 为训练步总数，尺度因子 $\gamma^{(l)} = 0.1 + 0.05 \cdot l$ 则随 LoD层级 l的增加而增大。

最后，不同于使用独立网络来预测高斯属性的 Scaffold-GS [49]，我们使用共享多层感知机网络

$\mathrm{F}_{\mathrm{mlp}}$ 根据这些锚点特征 $\mathbf{f}$、观察距离 Δ_c和相机信息 d_c，预测层次化高斯泼溅样本 $\mathcal{G} = \{\mathcal{G}_0, \mathcal{G}_1, \ldots, \mathcal{G}_{L-1}\}$ 的高斯属性：

$$\mathcal{G} = \{\mathcal{G}_0, \mathcal{G}_1, \ldots, \mathcal{G}_{L-1}\} = \mathrm{F}_{\mathrm{mlp}}(\hat{\mathbf{f}}, \Delta_c, \vec{d_c}) \tag{7}$$

我们发现，使用共享多层感知机有助于避免多个网络引入过多参数，从而在不降低性能的情况下加速训练和渲染。

B. 基于 *GMM* 的运动划分

按照第 IV-A 节所述，使用首帧（定义为规范空间 $\mathcal{G}_{\mathrm{cano}}$）初始化层次细节结构高斯后，我们专注于仅对后续帧中的动态区域进行建模，以降低优化成本并实现更快的训练。我们的思路是将当前帧 $\{\mathbf{I}_{t=c}^{(i)}\}_{i=1}^{N}$ 与从$\mathcal{G}_{\mathrm{cano}}$ 渲染得到的规范帧 $\{\hat{\mathbf{I}}_{t=0}^{(i)}\}_{i=1}^{N}$ 进行比较，通过计算二者差异的梯度来估计空间变化。该梯度能够将场景中的锚点分为两组：动态锚点 $\mathbf{a}^{\mathrm{dyn}}$和静态锚点$\mathbf{a}_{\mathrm{static}}$。梯度计算如下：

$$\mathbf{g}_{\boldsymbol{\mu}_{\mathrm{cano}}} = \frac{1}{N}\sum_{i=1}^{N} \nabla_{\boldsymbol{\mu}_{\mathrm{cano}}} \left\| \mathbf{I}_{t=c}^{(i)} - \hat{\mathbf{I}}_{t=0}^{(i)} \right\|^2 \tag{8}$$

其中，$\boldsymbol{\mu}_{\mathrm{cano}}$ 表示规范空间中高斯中心 $\boldsymbol{\mu}$的二维屏幕空间投影。随后，我们应用 GMM [25]，根据梯度 $\mathbf{g}^{\boldsymbol{\mu}_{\mathrm{cano}}}$对锚点进行聚类，其中均值较高的分量对应动态区域。如果锚点的 GMM后验概率超过阈值$\rho = 80\%$，则将其分类为动态锚点$\mathbf{a}^{\mathrm{dyn}}$；其余锚点则分类为静态锚点 $\mathbf{a}_{\mathrm{static}}$。随后，动态锚点 $\mathbf{a}^{\mathrm{dyn}}$ 将用于量化动态学习，如第 IV-C 节所述。

C. 量化残差细化

将动态锚点$\mathbf{a}^{\mathrm{dyn}}$ 与静态锚点$\mathbf{a}_{\mathrm{static}}$分解后，我们旨在高效压缩动态场景中的时间变化。为此，我们通过学习动态锚点$\mathbf{a}^{\mathrm{dyn}}$的量化残差 $\tilde{\mathbf{r}}$ 来实现：

$$\mathbf{A}_{\mathrm{dyn}} = \mathbf{A}_{\mathrm{dyn}} + \tilde{\mathbf{r}}, \quad \tilde{\mathbf{r}} = \begin{cases} \tilde{\boldsymbol{\mu}}_{\mathrm{anchor}} & \text{Anchor location} \\ \mathbf{Q}(\tilde{\mathbf{l}}_{\mathrm{attr}}) & \text{Other attributes} \end{cases} \tag{9}$$

其中$\mathbf{A}^{\mathrm{dyn}}$ 表示$\mathbf{a}^{\mathrm{dyn}}$的锚点属性。 $\tilde{\boldsymbol{\mu}}_{\mathrm{anchor}}$ 是一个可学习参数，表示锚点的空间坐标；由于其存储占用可以忽略不计，因此不对其进行量化。$\mathbf{l}_{\mathrm{attr}}$ 也是一个可学习参数（由锚点特征和偏移量组成），我们对其进行量化。我们将$\mathbf{Q}(\mathbf{l}_{\mathrm{attr}})$定义为量化后的潜变量，并通过直通估计器 [54]优化量化过程。通过固定静态锚点$\mathbf{a}_{\mathrm{static}}$ 并对动态锚点$\mathbf{a}^{\mathrm{dyn}}$的量化残差进行建模，我们的方法在保持视觉质量的同时，实现了约 80% 的存储量减少（见表 IV ）。

D. 流式训练

图 2 底部展示了我们的流式训练过程。第一步是使用首帧 $\{\mathbf{I}_{t=0}^{(i)}\}_{i=1}^{N}$ 训练模型，并初始化层次细节结构的三维高斯表示（图 2-(a)）。此步骤确定规范空间 $\mathcal{G}_{\mathrm{cano}}$的结构，包括层次细节层的数量、初始高斯属性，以及锚点的中心和数量。对于后续帧 $\{\mathbf{I}_{t}^{(i)}\}_{i=1}^{N}$，其中 $t = 1, 2, \ldots, T$，我们以逐帧、流式的方式优化基于 GMM 的运动划分和量化残差细化，其中仅优化与动态锚点相关的参数。所有训练均使用相同的损失函数：

$$\mathcal{L}_{\mathrm{recon}} = (1-\lambda)\mathcal{L}_1 + \lambda\mathcal{L}_{D\text{-SSIM}} \tag{10}$$

其中，λ 是一个超参数。我们将 $\lambda = 0.2$ 设为优先实现清晰重建，同时保持感知连贯性 [10]。

V. 实验

本节首先介绍实验设置。接着，我们将定量和定性比较结果与最先进的基准进行比较，以验证模型在各种视图密度下的重建质量。此外，我们还针对各个分量、训练视图数量和 LoD-AO 架构进行了系统的消融分析，最后评估所提出的 StreamLoD-GS 的局限性。

A. 实验设置

1) 数据集：我们在两个公开数据集上进行实验：Neural 3D Video（**N3DV**） [55] 和**MeetRoom** [1] 数据集。N3DV包含六个动态室内场景，每个视频均以2704×2078 的分辨率和30帧率拍摄，每个场景包含18–21个视频。按照 [10]–[13], [15],，我们将分辨率下采样至1352×1014。我们采用3、4和6个视角进行训练，并使用其余视角进行评估。

Meet Room数据集使用13台同步的Azure Kinect相机拍摄，涵盖“Discussion”“Trimming”和“Vrheadset”三个室内场景，分辨率为1280×720 ，帧率为30。我们使用原始分辨率，并采用2、3、4、5、6、9和12视图进行训练，其余视图用于评估。

2）实现：遵循 [10]–[13],[15],中的协议，我们使用前 300 帧（约 10 秒）进行评估。根据提供的相机位姿，使用 VGGT [52]生成初始点云。为确保比较严谨且公平，对于每个场景，我们的框架和所有竞争基线方法均采用相同的初始化方式。在模型超参数方面，我们将基础体素尺寸 δ 设为 0.001，并将每个锚点对应的高斯数量 K 设为 10。根据经验观察，GMM 后验概率阈值 ρ 设为 80%。F_{mlp} 模块（详见图 2 和第 IV-A 节）由 2 层多层感知机组成，采用 ReLU 激活，隐藏层维度为 32。对于属性量化，潜在维度

表I：在 Meet Room [1] 数据集中使用3、4和5个训练视图进行的定量比较。每个类别中的最佳结果以**粗体**标记，次优结果以下划线突出显示。

视角	方法	峰值信噪比（dB）↑	结构相似性↑	学习感知图像块相似度↓	存储空间（MB）↓	训练（秒）↓	渲染（帧率）↑
3视图	StreamRF [1]	15.82	0.641	0.510	22.81	7.854	14.9
	3DGStream [10]	17.24	0.687	0.481	3.822	5.526	588
	HiCoM [11]	16.92	0.665	0.431	0.913	1.762	517
	4DGC [15]	17.36	0.713	0.416	0.653	17.41	418
	QUEEN [12]	17.01	0.670	0.443	0.860	**0.455**	486
	StreamLoD-GS⋆	18.50	0.783	0.400	**0.245**	0.643	**677**
	StreamLoD-GS	**18.85**	**0.793**	**0.383**	0.253	1.073	547
4视图	StreamRF [1]	16.75	0.748	0.452	19.53	8.210	17.1
	3DGStream [10]	18.36	0.797	0.413	3.850	5.510	609
	HiCoM [11]	18.58	0.809	0.391	0.894	2.012	524
	4DGC [15]	19.45	0.810	0.385	0.794	18.61	441
	QUEEN [12]	19.26	0.753	0.386	0.823	**0.581**	484
	StreamLoD-GS⋆	20.67	0.823	0.370	**0.209**	0.823	**689**
	StreamLoD-GS	**21.40**	**0.832**	**0.351**	0.215	1.581	625
5 个视角	StreamRF [1]	17.57	0.780	0.432	16.46	8.921	19.3
	3DGStream [10]	19.53	0.817	0.400	3.841	5.440	619
	HiCoM [11]	19.69	0.826	0.391	0.819	2.225	518
	4DGC [15]	21.28	0.831	0.380	0.639	19.52	425
	QUEEN [12]	21.22	0.803	0.343	0.603	**0.692**	485
	StreamLoD-GS⋆	21.95	0.843	0.343	**0.198**	0.973	**621**
	StreamLoD-GS	**22.73**	**0.861**	**0.310**	0.205	1.910	608

锚点特征和偏移量均统一设为 12。如第 IV-A 节所述，最大 LoD层级 $L_{\max}$ 在优化过程中逐步增加。对于锚点细化，我们对初始帧的 200 次迭代和后续帧的 30 次迭代分别计算平均梯度。每个细化周期结束后，若某个锚点关联的神经高斯的累积不透明度低于 0.05，则将该锚点剔除。其余所有超参数均遵循 Octree-GS [21]中指定的配置。

模型在初始时间步训练 500 个训练轮次，随后每个时间步训练 10 个训练轮次，以实现快速适应。遵循 [10]–[13]，中确立的评估协议，我们使用逐帧峰值信噪比（PSNR）、结构相似性指数（SSIM） [56], 和学习感知图像块相似度（LPIPS） [57]评估视觉质量。这些指标在所有视角上取平均值，以确保全面评估。此外，我们报告平均存储占用、每个时间步的训练延迟以及渲染吞吐量（帧率）。所有实验均在单个 NVIDIA A800-SXM4-80GB GPU 上进行。

3) 基线方法： 我们将所提出的 StreamLoD-GS 与多个最先进的 SFVV框架进行比较，包括 StreamRF [1]， 3DGStream [10],HiCoM [11], 4DGC [15],和 QUEEN [12]。

为确保评估严谨且公正，所有基线方法均在标准化实验环境中按照官方配置实现。具体而言，我们对每个场景中的所有方法统一采用点云初始化（使用 VGGT [52]）。对于采用与我们的方法类似训练协议的 QUEEN，我们严格遵循其官方设置。所有模型均使用相同的视图输入，并在相同硬件上重新训练，以消除计算差异导致的性能偏差。此外，我们还评估了一种专门变体，StreamLoD-GS⋆，它基于我们提出的 StreamLoD-GS。该变体每四帧执行一次静态锚点更新和动态锚点识别（如第 IV-B 节所述），以积累更丰富的梯度信息。此外，鉴于短时间间隔内的动态锚点数量有限，当某个锚点的后验概率（如第 IV-B 节所定义）超过$\rho = 90\%$时，StreamLoD-GS⋆ 会将其归类为动态锚点。

B. 稀疏视图下的定量比较

我们在不同视图密度下进行了全面的定量评估，并在表 I 和表 II 中将我们的方法与最先进的基线方法进行了比较。表 I 列出了在 Meeting Room [1] 数据集上的对比结果。结果表明，在所测试的各类视图配置中，StreamLoD-GS 的表现始终优于几乎所有基线方法。具体而言，在极具挑战性的 3 视图设置下，StreamLoD-GS 达到了最高的质量指标，而 StreamLoD-GS⋆ 变体则保持了更快的渲染速度、更精简的存储空间占用和有竞争力的训练效率。在 4 视图下，StreamLoD-GS 的峰值信噪比达到 21.40 dB，比 4DGC 高出 1.95 dB，而存储占用仅为其 27%。在 5 视图配置下，我们的方法再次取得了最佳质量指标，且仅需 0.205 MB 存储空间；与 QUEEN 相比，存储空间减少了 66%，尽管后者采用了专门的基于高斯的学习量化框架。

同样，N3DV 数据集 [55](表 II) 上的定量结果进一步验证了我们模型的效率。在3个训练视角下，我们的方法达到 20.54 dB 的峰值信噪比，优于排名第二的方法（4DGC），而存储空间仅为其 15%。在 4 个视角下，StreamLoD-GS 达到 21.30 dB，比最接近的竞争方法高出 1.14 dB。值得注意的是，我们的存储占用仍然极为精简，仅为

表 II N3DV [55] 数据集在 3、4 和 5 个训练视图下的定量比较。每个类别中的最佳结果以**粗体**标记。第二佳结果以带下划线的形式突出显示。

视角	方法	峰值信噪比（dB）↑	结构相似性↑	学习感知图像块相似度 ↓	存储空间（MB）↓	训练（秒）↓	渲染（帧率）↑
3视图	StreamRF [1]	18.79	0.732	0.401	41.23	13.614	19.5
	3DGStream [10]	19.89	0.747	0.373	6.301	9.162	**662**
	HiCoM [11]	19.21	0.741	0.352	1.201	1.83	512
	4DGC [15]	19.82	0.763	0.339	0.871	17.49	484
	QUEEN [12]	19.25	0.743	0.330	0.962	**0.452**	460
	StreamLoD-GS⋆	<u>20.50</u>	<u>0.785</u>	<u>0.320</u>	**0.112**	<u>0.793</u>	<u>651</u>
	StreamLoD-GS	**20.54**	**0.790**	**0.316**	<u>0.126</u>	1.336	530
4视图	StreamRF [1]	19.08	0.747	0.390	39.68	16.46	22.3
	3DGStream [10]	20.16	0.780	0.363	6.610	8.830	**687**
	HiCoM [11]	19.81	0.767	0.326	1.024	2.36	501
	4DGC [15]	20.12	0.779	0.305	0.837	18.20	478
	QUEEN [12]	19.64	0.773	0.293	0.710	**0.568**	482
	StreamLoD-GS⋆	<u>21.14</u>	<u>0.812</u>	<u>0.283</u>	**0.243**	<u>0.982</u>	<u>679</u>
	StreamLoD-GS	**21.30**	**0.817**	**0.280**	<u>0.256</u>	1.696	525
6个视角	StreamRF [1]	20.95	0.762	0.380	12.78	17.25	26.1
	3DGStream [10]	21.85	0.800	0.351	6.624	8.965	**666**
	HiCoM [11]	21.17	0.812	0.306	0.847	4.519	499
	4DGC [15]	22.54	0.833	0.298	0.784	19.07	472
	QUEEN [12]	22.88	0.841	0.289	0.718	**0.816**	466
	StreamLoD-GS⋆	<u>23.36</u>	<u>0.842</u>	<u>0.271</u>	**0.211**	<u>1.401</u>	<u>524</u>
	StreamLoD-GS	**23.85**	**0.852**	**0.256**	<u>0.223</u>	2.313	506



图 3. N3DV [55]和 Meet Room [1] 数据集在 3个训练视角下的定性结果。我们展示了 3DGStream [10], 、QUEEN [12], 以及我们的方法生成的新视角结果，以供比较。

0.256 MB，比 QUEEN 小 64%。在 6视图场景中，我们的方法以紧凑的 0.223 MB 存储空间，在质量指标方面继续保持领先。在这些测试中，StreamLoD-GS⋆ 变体的质量始终排名第二，同时提供接近最优的渲染速度，并显著缩短训练时间。

表III 使用 12 个视角进行训练并保留一个视角用于测试时，Meet Room [1] 数据集上的定量比较。每个类别中的最佳结果以**粗体**标记，次佳结果以<u>下划线</u>突出显示。

方法	PSNR（dB）↑	存储空间（MB）↓	训练（s）↓	渲染（帧率）↑
3DGStream [10]	26.36	4.51	4.95	<u>309</u>
HiCoM [11]	26.73	0.60	3.92	289
4DGC [15]	26.87	0.63	21.3	273
QUEEN [12]	<u>27.35</u>	<u>0.35</u>	**1.31**	254
StreamLoD-GS	**27.84**	**0.19**	<u>2.93</u>	**312**

C. 稠密视角下的定量比较

为进一步评估 StreamLoD-GS 的鲁棒性，我们在稠密视角条件下，将其与最先进的基线方法进行比较分析，使用 12视图进行训练，并使用一个视角进行测试。如表 III 所示，StreamLoD-GS 始终优于所有竞争方法，在峰值信噪比、渲染吞吐量和存储占用方面均达到最佳，同时保持了极具竞争力的训练效率。这些结果凸显了我们框架出色的可扩展性和计算效率，表明 StreamLoD-GS 不仅在稀疏视角重建方面表现出色，在信息丰富的稠密视角场景中也保持着显著的性能优势。

D. 定性分析

图3展示了各种方法重建结果的视觉对比。在 N3DV 数据集上 [55]

表 IV Meet Room [1] 数据集上采用3个训练视角并保留全部数据用于测试的消融研究结果。LoD-AO：带锚点和八叉树的 LoD 结构化 3DGS。GMM-Part：基于 GMM 的运动划分。Q：量化残差细化；各类别中的最佳结果以**粗体**标出。

LoD-AO	GMM-Part	Q	峰值信噪比 (dB)↑	结构相似性↑	学习感知图像块相似度↓	存储空间 (MB)↓	训练 (s)↓	渲染（帧率）↑
✓	✓		22.51	0.852	0.320	1.683	1.673	582
✓		✓	**23.11**	0.857	0.326	0.283	2.197	425
	✓	✓	20.02	0.778	0.337	0.633	**0.902**	487
✓	✓	✓	22.73	**0.865**	**0.310**	**0.205**	1.910	**608**

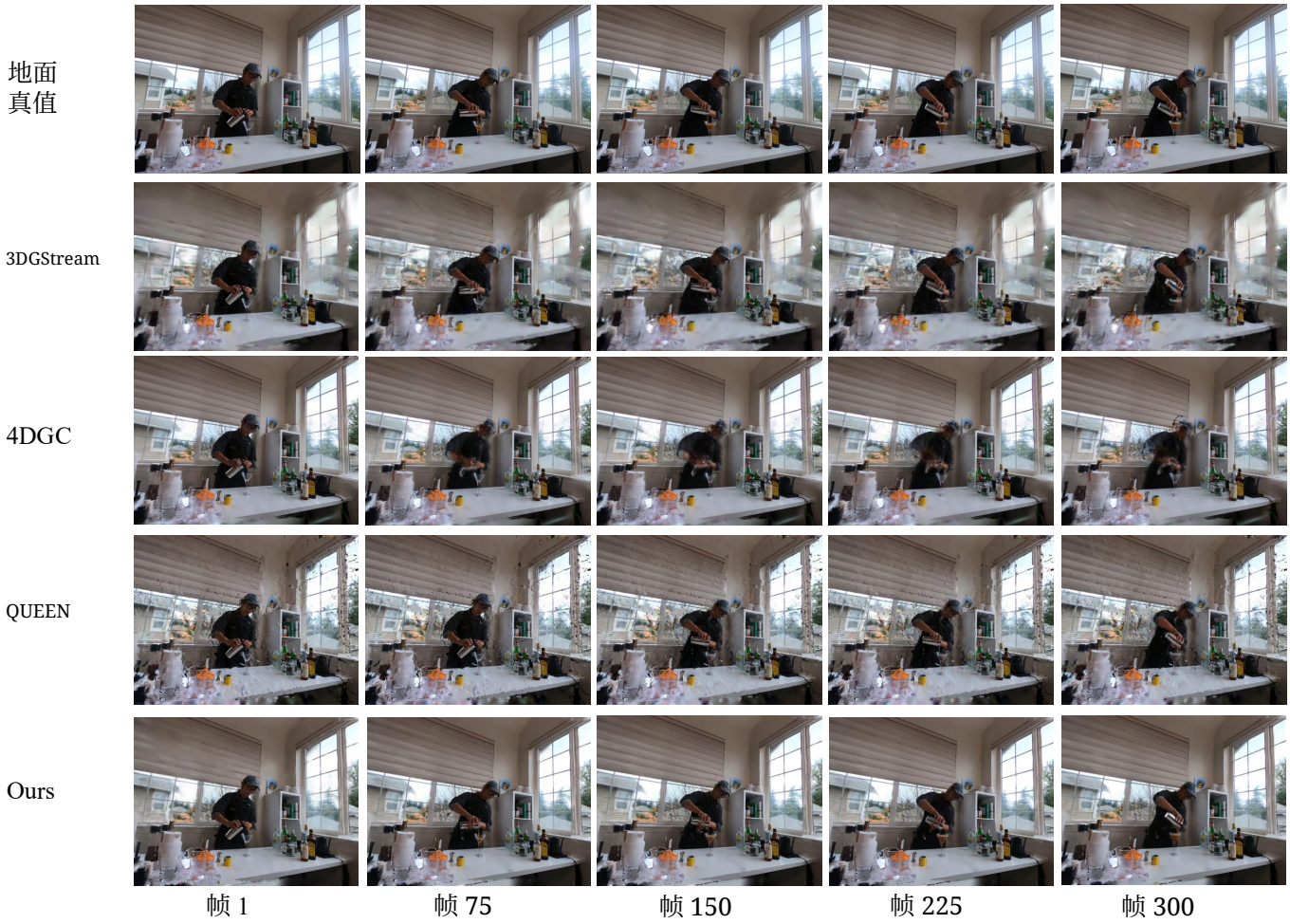


图 4. N3DV（场景：咖啡马提尼）数据集上的定量比较 [55] 使用3个训练视角。我们比较了在若干个按相等时间间隔采样的帧上的新视角结果，这些结果由 3DGStream [10], 4DGC [15], QUEEN [12], 以及我们的方法生成。

（场景：咖啡马提尼，第一行），与 3DGStream [10] 和 QUEEN [12]，例如相比，我们基于 LoD 的方法能够重建出更精细、结构更完整的细节，例如窗户，以及人物头部与背景墙之间的过渡。窗框更加清晰，主体头部与背景墙之间的边界过渡也更加锐利，改进十分明显。具体而言，我们的方法有效减少了 3DGStream 中常见的明显漂浮伪影，以及 QUEEN 中出现的结构侵蚀。同样，对于会议室数据集 [1]（场景：Discussion，第二行），我们的方法在前景和背景中都保留了更出色的细节：前景中有椅子和麦克风的复杂结构等，背景中则有玻璃墙上的反射等；相比之下，基线方法表现出明显的模糊或几何失真。

此外，我们还比较了“Discussion”和“Coffee Martini”场景中当前最先进方法对同一帧图像细节的重建效果。为进行更全面的比较，在图4和图5中，我们比较了两种场景——N3DV [55](Coffee Martini) 和 Meet Room [1] (Discussion)——中不同方法在若干按相等时间间隔采样的帧上的重建结果。如图4和图5所示，在这些采样帧中，我们提出的 StreamLoD-GS 始终能够生成更高质量的重建结果。与 3DGStream [10]，中明显的伪影、QUEEN [12]，中精细细节的丢失以及 4DGC [15]，中观察到的时间过拟合相比，我们的方法能够提供更稳定的结果，且

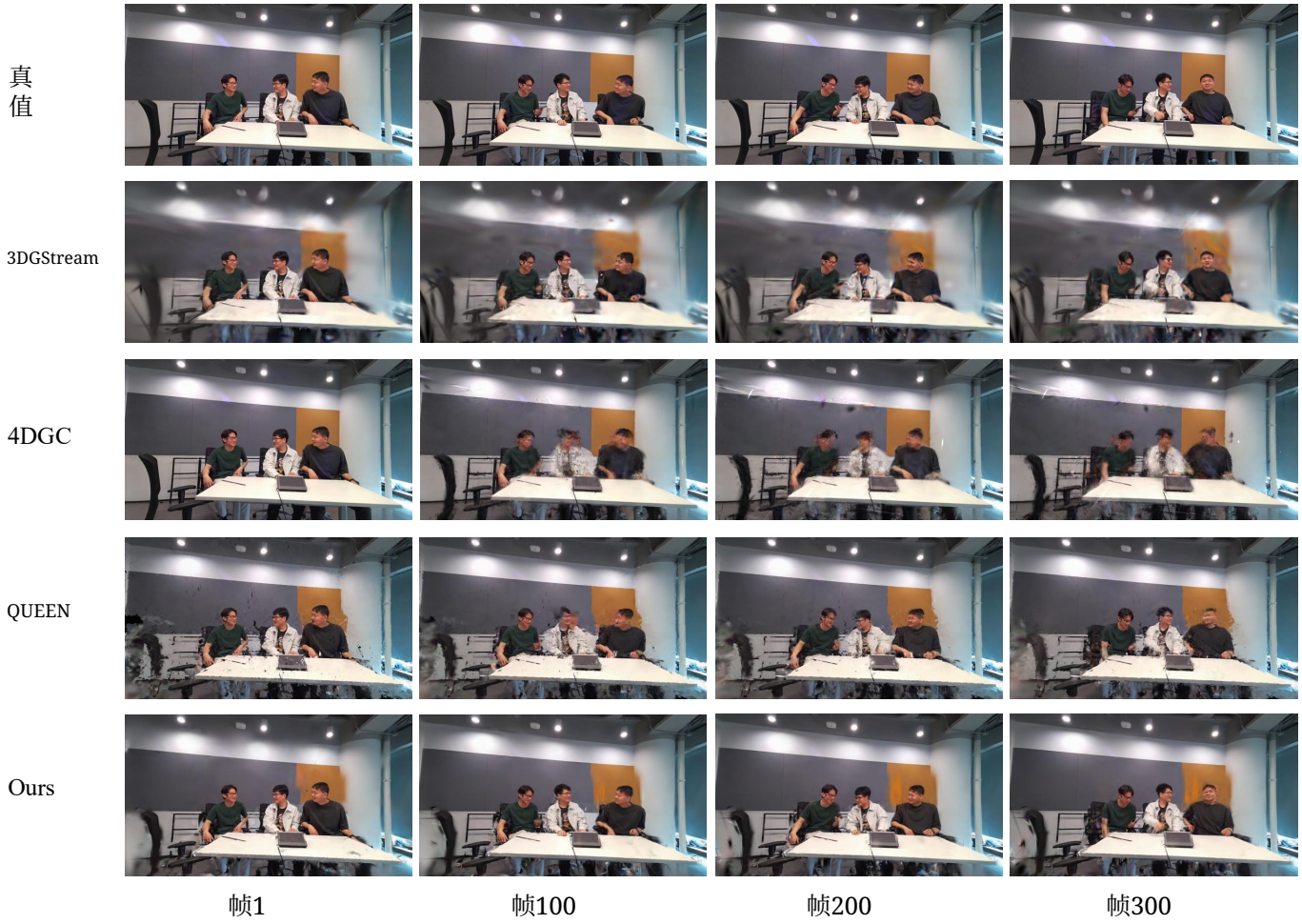


图5. 在包含3个训练视角的 Meet Room（场景：Discussion）数据集 [1] 上进行的定量比较。我们比较了在若干帧上生成的新视角结果，这些帧按相等的时间间隔采样，结果由 3DGStream [10], 4DGC [15], QUEEN [12], 以及我们的方法生成。

更丰富、更准确的场景细节。

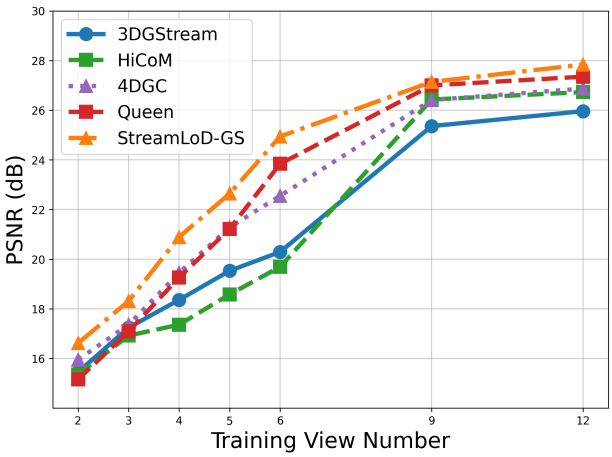


图6. 在训练视图数量不同的 Meet Room [1] 数据集上的定量比较。

表 V 采用 3个训练视角对 Meet Room [1] 数据集进行定量比较。HD-GS：分层高斯丢弃。F_{SEPS}表示一种变体，该变体使用独立网络来估计高斯属性，以替换统一属性预测多层感知机（F_{MLP}）。

方法	PSNR（dB）↑	存储空间（MB）↓	训练（s）↓	渲染（帧率）↑
w F_{seps}	18.68	0.290	1.330	425
不使用 HD-GS	17.99	0.256	1.096	498
LoD-AO	18.85	0.253	1.073	547

E. 消融研究

1）模块贡献分析：我们评估了三个核心组件各自的贡献：带锚点和八叉树的 LoD 结构化 3DGS（LoD-AO）、基于 GMM 的运动划分（GMM-Part）以及量化残差细化（Q）。如表 IV 所示，StreamLoD-GS 的完整配置在各项指标上实现了最佳平衡，包括结构相似性、学习感知图像块相似度、存储效率和渲染吞吐量。具体而言，移除量化残差细化模块会导致重建质量略有下降，同时使存储占用膨胀至 1.683 MB。移除 GMM-Part 则会导致存储空间增加，且

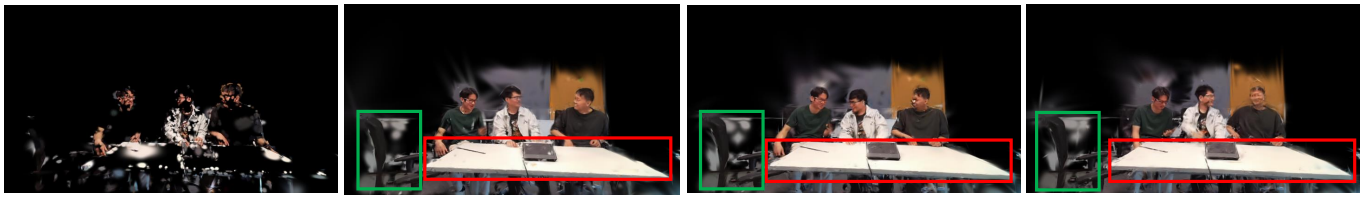


图 7. 使用我们的基于 GMM 的运动划分策略分离动态锚点后，在 Meet Room（场景：Discussion）数据集 [1] 中一组等间距帧上的渲染结果。

训练延迟上升，渲染速度降至 425 帧率。值得注意的是，省略 LoD-AO 可将训练时间缩短至 0.902 s，但代价是峰值信噪比大幅下降 2.71 dB（从 22.73 dB 降至 20.02 dB），且存储空间增加一倍以上。这些结果表明，每个组件都不可或缺，并且它们通过协同工作来维持高性能。

2) 分层丢弃与共享多层感知机下的 *LoD-AO* 分析：在 LoD-AO框架中，我们引入了分层高斯丢弃（HD-GS）一种具备层级感知能力的策略，旨在缓解稀疏视角约束下的过拟合，并稳定优化过程。此外，为避免多个网络带来的参数爆炸，我们采用共享多层感知机（F_{mlp}）来预测分层高斯的属性。与采用独立网络完成这项任务的 Scaffold-GS [49]，不同，我们的共享架构在不影响保真度的前提下，加快了训练和渲染速度。为验证这些设计选择，我们将共享 F_{mlp} 与采用独立网络的变体（F_{seps}）进行比较，并评估移除 HD-GS 模块的影响。如表 V 所示，F_{seps} 变体的峰值信噪比明显下降，渲染吞吐量也更低，证实了共享网络的效率与有效性。尤为重要的是，禁用 HD-GS 模块会导致峰值信噪比大幅下降，这表明分层丢弃对于维持重建保真度、确保稀疏视角设置下的稳健优化至关重要。这些发现表明，结合 F_{mlp} 与 HD-GS，LoD-AO 能够在最大限度提升训练效率的同时，实现峰值渲染速度（547 帧率）和最低存储成本。

3）不同训练视图的分析：图 6 展示了不同方法在 Meet Room [1] 数据集上，使用不同数量训练视图（2、3、4、5、6、9 和 12）时的重建效果。对于每种配置，所有可用视图均保留用于测试，以严格评估新视图合成效果。结果表明，所有受评方法的峰值信噪比均与训练视图数量呈正相关，证实额外的视点为场景重建提供了必要的几何和光度约束。值得注意的是，在所有视图密度下，StreamLoD-GS 的表现始终优于 3DGStream [10]，HiCoM [11],4DGC [15], 和 QUEEN [12] 。这一持续的性能差距凸显了我们的框架在稀疏视角约束下更出色的鲁棒性和泛化能力。这些发现彰显了 StreamLoD-GS 即使在训练数据极其有限的情况下，也能有效重建高质量动态场景的卓越能力即使提供的训练数据严重受限，也能实现这一点。

F. 局限性

如图 7 所示，随着视频序列的推进，我们基于 GMM 的运动划分策略能够有效地将静态区域与动态区域分离。然而，在处理具有镜面反射的静态、视觉上均匀的表面时，仍存在一个显著局限，例如具有反光效果的椅子表面、中央的笔记本电脑以及纯白色桌子（图 7 中红色和绿色方框所示）。具体而言，反射造成的视角相关的外观变化会引入时间不一致性，这是近期文献中公认的一项挑战 [58], [59]。连续帧之间这些细微的光度变化会产生虚假梯度，导致 GMM 将静态区域误分类为动态区域。因此，提升视觉均匀区域的运动分割鲁棒性，尤其是在复杂光照和反光条件下，仍是一项关键目标。在未来的工作中，我们计划引入颜色一致性约束和三维几何先验，以更好地区分真实运动与镜面反射伪影，从而提高动态锚点识别的精度。

VI. 结论

本文提出 StreamLoD-GS，这是一种新颖的分层细节层次（LoD）高斯泼溅框架，旨在高效重建流式自由视点视频（SFVV）。为缓解过拟合并稳定优化过程，我们采用了分层高斯随机失活技术，有选择地丢弃不必要的高斯。StreamLoD-GS 还包含一个基于 GMM 的运动分区模块，可将 3DGS 划分为动态区域和静态区域，分别由动态锚点和静态锚点表示。在后续训练中，静态锚点保持冻结，而动态锚点则随着新帧到达以流式方式更新。此外，我们还引入了量化残差细化模块，对动态锚点的残差进行量化，从而在不影响性能的前提下有效降低存储成本。为评估 StreamLoD-GS 的有效性，我们在各种视点设置下进行了广泛实验。结果表明，我们的方法优于现有方法，在存储效率、渲染速度和视觉质量方面均有显著提升。

参考文献

[1] L. Li、Z. Shen、Z. Wang、L. Shen 和 P. Tan，《用于三维视频合成的流式辐射场》，载于《神经信息处理系统进展》，第 35 卷，2022 年，第 13 485–13 498 页。[2] Y. Chen、Q. Wang、H. Chen、X. Song、H. Tang 和 M. Tian，《增强现实技术概述》，载于《物理学杂志：会议系列》，第 1237 卷，第 2 期，IOP 出版社，2019 年，第 022082 页。[3] B. Kerbl、G. Kopanas、T. Leimk¨uhler 和 G. Drettakis，《用于实时辐射场渲染的3D高斯泼溅》，《ACM 图形学汇刊》，第 42 卷，第 4 期，第 139–1 页，2023 年。[4] A. Collet、M. Chuang、P. Sweeney、D. Gillett、D. Evseev、D. Calabrese、H. Hoppe、A. Kirk 和 S. Sullivan，《高质量可流式传输的自由视点视频》，《ACM 图形学汇刊》（ToG），第 34 卷，第 4 期，第 1–13 页，2015 年。[5] Y. Chen、Y. Liang、Z. Wang、D. Wang、C. Xie、S. Jiao 和 L. Zhang，《LiveGS：借助面向移动设备的高性能高斯泼溅实现实时自由视点视频》，载于 ACM SIGGRAPH 2025 新兴技术，2025 年，第 1–2 页。[6] X. Zhang、Z. Liu、Y. Zhang、X. Ge、D. He、T. Xu、Y. Wang、Z. Lin、S. Yan 和 J. Zhang，《Mega：面向动态场景的内存高效四维高斯泼溅》，载于 IEEE/CVF 国际计算机视觉会议论文集，2025 年，第 27 828–27 838 页。[7] L. Tang、J. Yang、R. Peng、Y. Zhai、S. Shen 和 R. Wang，《将可流式传输的自由视点视频压缩至每帧 0.1 MB》，载于 AAAI 人工智能会议论文集，第 39 卷，第 7 期，2025 年，第 7257–7265 页。[8] M. Kim、J. Lim 和 B. Han，《通过不确定性感知正则化实现真实场景中的四维高斯泼溅》，《神经信息处理系统进展》，第 37 卷，第 129 209–129 226 页，2024 年。[9] G. Wu、T. Yi、J. Fang、L. Xie、X. Zhang、W. Wei、W. Liu、Q. Tian 和 X. Wang，《用于实时动态场景渲染的四维高斯泼溅》，载于 IEEE/CVF 计算机视觉与模式识别会议论文集，2024 年，第 20 310–20 320 页。[10] J. Sun、H. Jiao、G. Li、Z. Zhang、L. Zhao 和 W. Xing，《3DGStream：即时训练三维高斯，实现照片级真实感自由视点视频的高效流式传输》，载于 IEEE/CVF 计算机视觉与模式识别会议论文集，2024 年，第 20 675–20 685 页。[11] Q. Gao、J. Meng、C. Wen、J. Chen 和 J. Zhang，《HiCoM：通过 3D高斯泼溅实现动态可流式传输场景的分层相干运动》，载于《神经信息处理系统进展》，第 37 卷，2024 年，第 80 609–80 633 页。[12] S. Girish、T. Li、A. Mazumdar、A. Shrivastava、S. De Mello 等，《QUEEN：用于流式自由视点视频的动态高斯量化高效编码》，《神经信息处理系统进展》，第 37 卷，第 43 435–43 467 页，2024 年。[13] J. Yan、R. Peng、Z. Wang、L. Tang、J. Yang、J. Liang、J. Wu 和 R. Wang，《Instant Gaussian Stream：通过高斯泼溅实现动态场景重建的快速、可泛化流式传输》，载于计算机视觉与模式识别会议论文集，2025 年，第 16 520–16 531 页。[14] H. Han、J. Li、H. Wei 和 X. Ji，《Event-3DGS：使用 3D高斯泼溅进行基于事件的三维重建》，《神经信息处理系统进展》，第 37 卷，第 128 139–128 159 页，2024 年。[15] Q. Hu、Z. Zheng、H. Zhong、S. Fu、L. Song、X. Zhang、G. Zhai 和 Y. Wang，《4DGC：面向高效可流式传输自由视点视频的码率感知四维高斯压缩》，载于计算机视觉与模式识别会议论文集，2025 年，第 875–885 页。[16] T. M¨uller、A. Evans、C. Schied 和 A. Keller，《采用多分辨率哈希编码的即时神经图形基元》，《ACM 图形学汇刊》（TOG），第 41 卷，第 4 期，第 1–15 页，2022 年。[17] L. Wang、Q. Hu、Q. He、Z. Wang、J. Yu、T. Tuytelaars、L. Xu 和 M. Wu，《用于可流式传输自由视点视频的神经残差辐射场》，载于 IEEE/CVF 计算机视觉与模式识别会议论文集，2023 年，第 76–87 页。[18] H. Park、G. Ryu 和 W. Kim，《DropGaussian：面向稀疏视图高斯泼溅的结构正则化》，载于计算机视觉与模式识别会议论文集，2025 年，第 21 600–21 609 页。[19] Y. Xu、L. Wang、M. Chen、S. Ao、L. Li 和 Y. Guo，《DropoutGS：丢弃高斯以实现更佳的稀疏视图渲染》，载于计算机视觉与模式识别会议论文集，2025 年，第 701–710 页。[20] D. Luebke、M. Reddy、J. D. Cohen、A. Varshney、B. Watson 和 R. Huebner，《三维图形细节层次》，爱思唯尔，2002 年。[21] K. Ren、L. Jiang、T. Lu、M. Yu、L. Xu、Z. Ni 和 B. Dai，《Octree-GS：通过基于细节层次结构的三维高斯实现一致的实时渲染》，《IEEE 模式分析与机器智能汇刊》，2025 年。[22] J. Seo、Y. S. Choi、H. S. Son 和 Y. Uh，“Flod：将灵活的细节层次融入3D高斯泼溅以实现可定制渲染”，*arXiv* 预印本 *arXiv:2408.12894*，2024年。[24] J. Kulhanek、M.-J. Rakotosaona、F. Manhardt、C. Tsalicoglou、M. Niemeyer、T. Sattler、S. Peng 和 F. Tombari，“Lodge：面向大规模高斯泼溅的细节层次与高效渲染”，*arXiv* 预印本 *arXiv:2505.23158*，2025年。[25] H. Permuter、J. Francos 和 I. Jermyn，“用于图像分类与分割的颜色和纹理特征高斯混合模型研究”，模式识别，第39卷，第4期，第695–706页，2006年。[26] B. Mildenhall、P. P. Srinivasan、M. Tancik、J. T. Barron、R. Ramamoorthi 和 R. Ng，“NeRF：将场景表示为用于视图合成的神经辐射场”，*ACM*通讯，第65卷，第106期，第99页，2021年。[27] J. T. Barron、B. Mildenhall、M. Tancik、P. Hedman、R. Martin-Brualla 和 P. P. Srinivasan，“Mip-NeRF：用于抗锯齿神经辐射场的多尺度表示”，载于*IEEE/CVF* 国际计算机视觉会议论文集，2021年，第5855–5864页。[28] J. T. Barron、B. Mildenhall、D. Verbin、P. P. Srinivasan 和 P. Hedman，“Mip-NeRF 360：无界抗锯齿神经辐射场”，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2022年，第5470–5479页。[29] A. Cao 和 J. Johnson，“HexPlane：动态场景的快速表示”，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2023年，第130–141页。[30] S. Fridovich-Keil、G. Meanti、F. R. Warburg、B. Recht 和 A. Kanazawa，“K-Planes：空间、时间与外观中的显式辐射场”，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2023年，第12 479–12 488页。[31] L. Song、A. Chen、Z. Li、Z. Chen、L. Chen、J. Yuan、Y. Xu 和 A. Geiger，“NeRFPlayer：一种由分解式神经辐射场构成的可流式传输动态场景表示”，*IEEE* 可视化与计算机图形学汇刊，第29卷，第5期，第2732–2742页，2023年。[32] L. Liu、Z. Chen 和 D. Xu，“基于先验混合的3D高斯泼溅数据压缩”，载于第*33*届 *ACM*国际多媒体会议论文集，2025年，第8341–8350页。[33] A. Yu、V. Ye、M. Tancik 和 A. Kanazawa，“pixelNeRF：从一张或少量图像生成神经辐射场”，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2021年，第4578–4587页。[34] A. Jain、M. Tancik 和 P. Abbeel，“精简NeRF：语义一致的少样本视图合成”，载于*IEEE/CVF* 国际计算机视觉会议论文集，2021年，第5885–5894页。[35] M. Niemeyer、J. T. Barron、B. Mildenhall、M. S. Sajjadi、A. Geiger 和 N. Radwan，“RegNeRF：对神经辐射场进行正则化，以便从稀疏输入合成视图”，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2022年，第5480–5490页。[36] G. Wang、Z. Chen、C. C. Loy 和 Z. Liu，“SparseNeRF：通过蒸馏深度排序实现少样本新视角合成”，载于*IEEE/CVF* 国际计算机视觉会议论文集，2023年，第9065–9076页。[37] J. Yang、M. Pavone 和 Y. Wang，“FreeNeRF：利用免费频率正则化改进少样本神经渲染”，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2023年，第8254–8263页。[38] A. Radford、J. W. Kim、C. Hallacy、A. Ramesh、G. Goh、S. Agarwal、G. Sastry、A. Askell、P. Mishkin、J. Clark 等，“通过自然语言监督学习可迁移的视觉模型”，载于国际机器学习会议。PMLR，2021年，第8748–8763页。[39] J. Chung、J. Oh 和 K. M. Lee，“面向少样本图像的3D高斯泼溅深度正则化优化”，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2024年，第811–820页。[40] J. Li、J. Zhang、X. Bai、J. Zheng、X. Ning、J. Zhou 和 L. Gu，“DNGaussian：通过全局-局部深度归一化优化稀疏视角三维高斯辐射场”，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2024年，第20 775–20 785页。[41] A. Paliwal、W. Ye、J. Xiong、D. Kotovenko、R. Ranjan、V. Chandra 和 N. K. Kalantari，“CoherentGS：利用一致性三维高斯进行稀疏新视角合成”，载于欧洲计算机视觉会议。施普林格，2024年，第 19–37 页。

[42] J. Zhang、J. Li、X. Yu、L. Huang、L. Gu、J. Zheng 和 X. Bai，《CoR-GS：通过协同正则化实现稀疏视图3D高斯泼溅》，载于欧洲计算机视觉会议。施普林格，2024年，第335–352页。[43] Z. Zhu、Z. Fan、Y. Jiang 和 Z. Wang，《FSGS：使用高斯泼溅实现实时少样本视图合成》，载于欧洲计算机视觉会议。施普林格，2024年，第145–163页。[44] S. Nam、D. Rho、J. H. Ko 和 E. Park，《Mip-grid：用于神经辐射场的抗混叠网格表示》，神经信息处理系统进展，第36卷，第2837–2849页，2023年。[45] J. T. Barron、B. Mildenhall、D. Verbin、P. P. Srinivasan 和 P. Hedman，《Zip-nerf：抗混叠的基于网格的神经辐射场》，载于*IEEE/CVF* 国际计算机视觉会议论文集，2023年，第19 697–19 705页。[46] J. Cui、J. Cao、F. Zhao、Z. He、Y. Chen、Y. Zhong、L. Xu、Y. Shi、Y. Zhang 和 J. Yu，《Letsgo：通过激光雷达辅助高斯基元实现大规模车库建模与渲染》，*ACM* 图形学汇刊（*TOG*），第43卷，第6期，第1–18页，2024年。[47] Y. Liu、C. Luo、L. Fan、N. Wang、J. Peng 和 Z. Zhang，《CityGaussian：使用高斯实现实时高质量大规模场景渲染》，载于欧洲计算机视觉会议。施普林格，2024年，第265–282页。[48] B. Kerbl、A. Meuleman、G. Kopanas、M. Wimmer、A. Lanvin 和 G. Drettakis，《用于超大型数据集实时渲染的分层三维高斯表示》，*ACM* 图形学汇刊（*TOG*），第43卷，第4期，第1–15页，2024年。[49] T. Lu、M. Yu、L. Xu、Y. Xiangli、L. Wang、D. Lin 和 B. Dai，《Scaffold-GS：用于视图自适应渲染的结构化三维高斯》，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2024年，第20 654–20 664页。[50] N. Snavely、S. M. Seitz 和 R. Szeliski，《照片旅游：在三维空间中探索照片集》，载于*ACM SIGGRAPH 2006* 论文集，2006年，第835–846页。[51] J. L. Schonberger 和 J.-M. Frahm，《重新审视运动恢复结构》，载于*IEEE*计算机视觉与模式识别会议论文集，2016年，第4104–4113页。[52] J. Wang、M. Chen、N. Karaev、A. Vedaldi、C. Rupprecht 和 D. Novotny，《VGGT：视觉几何基础变换器》，载于计算机视觉与模式识别会议论文集，2025年，第5294–5306页。[53] K. Ren、L. Jiang、T. Lu、M. Yu、L. Xu、Z. Ni 和 B. Dai，《Octree-GS：迈向基于细节层次结构的三维高斯一致实时渲染》，*arXiv*预印本 *arXiv:2403.17898*，2024年。[54] Y. Bengio、N. L´eonard 和 A. Courville，《针对条件计算，通过随机神经元估计或传播梯度》，*arXiv*预印本 *arXiv:1308.3432*，2013年。[55] T. Li、M. Slavcheva、M. Zollhoefer、S. Green、C. Lassner、C. Kim、T. Schmidt、S. Lovegrove、M. Goesele、R. Newcombe 等，《基于多视图视频的神经三维视频合成》，载于*IEEE/CVF* 计算机视觉与模式识别会议论文集，2022年，第5521–5531页。[56] Z. Wang、A. C. Bovik、H. R. Sheikh 和 E. P. Simoncelli，《图像质量评估：从误差可见性到结构相似性》，*IEEE*图像处理汇刊，第13卷，第4期，第600–612页，2004年。[57] R. Zhang、P. Isola、A. A. Efros、E. Shechtman 和 O. Wang，《深度特征作为感知度量的惊人效力》，载于*CVPR*，2018年，第586–595页。[58] S. Yao、X. Zhang、X. Liu、M. Liu 和 Z. Cui，《Stdd：用于视频生成的时空双重扩散》，载于计算机视觉与模式识别会议论文集，2025年，第12 575–12 584页。[59] Y. Liu、H. Zhao、K. C. Chan、X. Wang、C. C. Loy、Y. Qiao 和 C. Dong，《通过深度特征传播和自正则化学习实现时间一致的视频着色》，《计算视觉媒体》，第10卷，第2期，第375–395页，2024年。



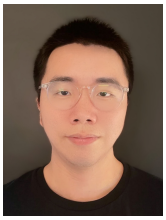
Xinhui Liu 于2015年和2018年分别获得西安工业大学电子信息工程专业的理学学士和理学硕士学位。她于2024年获得西安交通大学博士学位。2022年9月至2024年5月，她在澳大利亚新南威尔士州悉尼大学进行访问学习，并与Luping Zhou教授合作。她目前是中国香港香港大学计算与数据科学学院的博士后研究员。她目前的研究兴趣包括计算机视觉和计算机图形学。



Can Wang 目前是香港大学的研究员。他获得了中华人民共和国香港城市大学计算机科学系的博士学位。他于2020年获得中国科学技术大学计算机科学与技术专业的硕士学位。他目前的研究兴趣包括计算机图形学和计算机视觉。



Lei Liu Lei Liu 是香港大学的博士后研究员。她于 2025 年在中国北京航空航天大学获得博士学位。她分别于 2017 年和 2020 年在吉林大学仪器科学与电气工程学院获得学士学位和硕士学位。她的研究兴趣包括视觉数据压缩和三维高斯泼溅编辑。



Zhenghao Chen Zhenghao Chen 博士是纽卡斯尔大学的助理教授。他于 2017 年和 2022 年分别获得悉尼大学的信息技术学士荣誉一级学位和博士学位。此前，他曾在 TikTok 担任研究工程师，在悉尼大学担任研究员，并在微软研究院和迪士尼研究院担任访问研究科学家。Chen 博士的主要研究兴趣包括计算机视觉和机器学习。

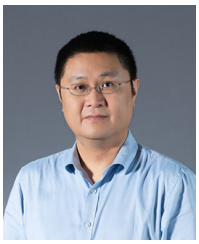


Wei Jiang 她于 2010 年在哥伦比亚大学获得电气工程博士学位，并于 2005 年和 2002 年分别在清华大学获得自动化专业理学硕士学位和工学学士学位。她在计算机视觉和机器学习领域发表了 40 多篇同行评审论文，并拥有 40 多项已授权专利，还持有 10+ IEEE 和联合视频专家组/动态图片专家组/联合图像专家组标准中的标准采纳。她是 IEEE 高级会员，并担任多个顶级会议和期刊的技术程序委员会委员及审稿人。Wei Jiang 目前是高级首席研究员任职于 Futurewei Technologies。她的研究兴趣广泛，涵盖计算机视觉和人工智能，包括基于人工智能的图像和视频压缩、图像和视频复原与生成、视觉计算以及多媒体内容分析。目前，她的研究重点是下一代基于人工智能的图像/视频压缩与复原的研究和标准化、视觉语言模型生成图像/视频，以及推理加速。



Wei Wang 于1998年和1995年分别获得中国复旦大学电气工程系的理学硕士学位和学士学位。他目前是 Futurewei Technologies 的首席研究员，曾任阿里巴巴集团高级研究员。他参与了国际标准化活动，包括为 ISO/IEC动态图像专家组的H.265/HEVC屏幕内容编码和神经网络表示工作项目作出贡献，以及为 IEEE 数据压缩标准委员会的神经图像工作项目作出贡献

编码。他曾担任动态图像专家组神经网络表示组的联合主席。目前的研究兴趣包括高性能计算、图像和视频压缩、神经网络压缩与加速，以及将算法转化为桌面、嵌入式或专用集成电路平台上的产品。



Dong Xu（IEEE 会士）于2001年和2005年分别获得中国合肥中国科学技术大学的工学学士学位和博士学位。在攻读博士学位期间，他曾在中国北京的微软亚洲研究院实习，并在香港的香港中文大学担任研究助理，历时两年多。他曾在美国纽约州纽约市的哥伦比亚大学担任博士后研究科学家一年。他还曾在新加坡的南洋理工大学担任教员，

新加坡，以及澳大利亚新南威尔士州悉尼大学的计算机工程讲席教授。他目前是中国香港香港大学计算与数据科学学院的教授。他曾合著一篇论文，该论文于2010年荣获IEEE计算机视觉与模式识别会议（CVPR）最佳学生论文奖；他还合著了另一篇论文，该论文于2014年荣获《IEEE多媒体汇刊》最佳论文奖。他目前的研究兴趣包括人工智能、计算机视觉、多媒体和机器学习。