

Sparse2DGS: Geometry-Prioritized Gaussian Splatting for Surface Reconstruction from Sparse Views

1. 论文内容概述

1.1 研究背景与动机

从稀疏输入视角进行高质量的表面重建是计算机视觉理解三维场景的关键挑战。通过3D高斯泼溅（3DGS）及其衍生的2D高斯泼溅（2DGS）技术，现已能分别获取高质量的辐射场渲染或更准确的表面几何近似。然而，这些方法并非稀疏视角下的完美通用解决方案，因其严重受限于对密集输入视角和良好SfM初始化点云的依赖；在数据稀疏时，往往因初始化点极其稀疏而陷入几何退化。对于仅有少量视角的场景分析，目前简单的将基于学习的多视图立体视觉（MVS）生成的密集点云与高斯泼溅结合，尚不存在能克服稀疏视角几何优化“不适定性（ill-posed nature）”并产生最优结果的方案。实现这种几何优先的高斯泼溅系统（Sparse2DGS），将把稀疏重建任务重构为由几何线索（深度与法线）引导的优化问题，为少样本表面重建开辟新的研究途径。

1.2 相关工作

1.2.1 基于新视角合成的 3D 重建

近年来，新视角合成（NVS）取得了显著进展，仅通过图像监督即可实现精确的 3D 表面重建。神经辐射场（NeRFs）已成为表示 3D 场景的重要技术，通过可微体积渲染训练参数化辐射场来提取表面。随后，Neus 和 VoISDF 通过将有符号距离函数（SDF）重新参数化为体积密度，进一步提高了表面的准确性。近期，3D Gaussian Splatting (3DGS) 利用运动结构（SfM）点云构建高斯基元，实现了快速且高质量的渲染。尽管后续工作试图利用 3DGS 进行更高效的表面重建，但在面对稀疏视角的训练数据时，提取的几何结构往往充满噪声且不完整。

* [22] NeRF (ECCV 2020) / [28] Neus (NeurIPS 2021) / [38] VoISDF (NeurIPS 2021): 该领域的经典工作。NeRF 奠定了神经辐射场的基础，而 Neus 和 VoISDF 通过引入 SDF 改进了表面学习，重建出更精确的表面，但主要针对密集视角。

- * [14] 3D Gaussian Splatting (SIGGRAPH 2023): 展示了在新视角合成方面的惊人效果，引发了大量将其用于表面重建的研究。
- * [11] 2DGS (SIGGRAPH 2024): 本文的基线方法。它用 2D 定向平面圆盘 (Disks) 取代了 3D 高斯椭球，提供了视图一致的几何表示。但在稀疏视角下，由于初始 SfM 点过于稀疏，难以生成忠实重建结果。
- * [10] SuGaR (CVPR 2024) / [43] GOF (arXiv 2024) / [4] PGSR (arXiv 2024): 近期尝试从高斯基元中提取网格的方法。GOF 构建高斯不透明度场来提取水平集，PGSR 引入几何约束。然而，在有限视角下，这些方法生成的几何结构仍存在明显的噪声和缺失。

1.2.2 基于学习的多视图立体视觉

基于深度学习的多视图立体视觉 (MVS) 旨在从多视角的姿态图像中估计参考图像的深度。MVSNet 首次引入了可微单应性变换，利用深度图像特征构建代价体积 (Cost Volume)，并使用 3D CNN 进行正则化。尽管后续工作在特征提取和代价聚合方面进行了改进，但它们通常需要真值深度图进行训练，这在实际中很难获取。相比之下，无监督 MVS 方法更适用于通用的现实场景。

- * [36] MVSNet (ECCV 2018): 开创性工作，通过构建代价体积进行深度推断，显著提高了完整性，即使在稀疏图像下也能提供较好的几何覆盖。
- * [31] CLMVSNet (ICCV 2023): 一种无监督的 MVS 方法，无需真值即可训练，具有更高的灵活性。本文选择该方法来初始化高斯点云，因为它能为每个输入视图推断密集的深度图。

1.2.3 稀疏视角表面重建

针对稀疏训练视角的 3D 表面重建是近期的研究热点。一些方法通过构建级联体素或利用表面先验来引导几何学习，取得了一定的重建效果。另一些前馈网络设计了专门的 Transformer 架构来增强特征。然而，这些前馈网络虽然能直接推断场景几何，但在未见过的场景中表现出的泛化能力有限，限制了其在现实世界场景中的应用。

- * [20] SparseNeuS (ECCV 2022): 本文的主要对比方法之一。作为首个基于 NeRF 的稀疏视角方法，它通过构建级联几何体来学习可泛化的辐射场，随后进行微调，实现了合理的网格重建。
- * [12] NeuSurf (AAAI 2024) / [40] SparseCraft (ECCV 2024): 尝试利用稀疏点上的表面先验或立体视觉线索进行几何正则化，以指导稀疏视角下的学习。
- * [25] VolRecon (CVPR 2023) / [19] ReTR (NeurIPS 2024): 设计了专门的 Transformer 架构来增强体素特征或渲染建模。尽管属于前馈网络，但依然面临泛化性不足的问题。

1.3 创新点

1. 提出了一种 **Sparse2DGS** 框架，这是一种测试时 (Test-time) 的稀疏视角 3D 表面重建方法，旨在结合 Gaussian Splatting 和多视图立体视觉 (MVS) 的优势。
2. 设计了 **几何优先的增强方案 (Geometric-Prioritized Enhancement)**，包含三个关键技术模块，以解决稀疏视角重建中常见的病态几何优化问题。
3. **几何增强监督 (Geometrically Enhanced Supervision)**: 利用 MVS 衍生的几何特征进行监督，同时在优化过程中固定特征和颜色值。

抗过拟合机制: 这种策略提供了具有几何表现力的监督信号，并防止了外观 (Appearance) 过拟合，从而显著提高了几何重建的质量。

4. **直接高斯基元正则化 (Direct Gaussian Primitive Regularization)**: 引入了一种基于重参数化的圆盘采样策略 (Reparameterization-based Disk Sampling)，将高斯基元的属性 (位置、方向、缩放) 转化为点表示。

统一优化: 这一方法使得能够在一个标准的、基于点的跨视图一致性损失 (Cross-view Consistency Loss) 框架下，对不同形式的高斯属性进行统一优化。

5. **选择性高斯更新 (Selective Gaussian Update)**: 摒弃了在病态条件下直接致密化高斯基元的做法，转而利用渲染出的几何线索 (如深度和法线) 来选择性地更新 MVS 初始化的高斯基元。

自导向修正: 这种策略允许进行自我导向的几何修正 (Self-guided Geometric Corrections)，进一步提升了重建的完整性和准确性。

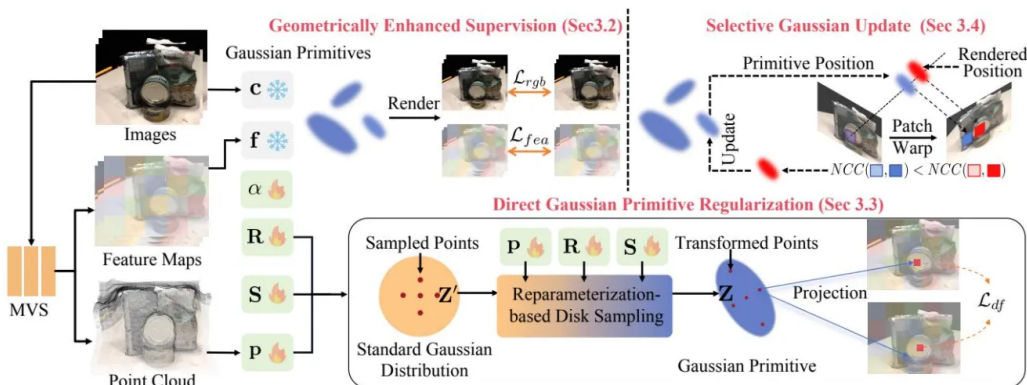
6. **卓越的性能与效率:** 在保持高质量重建的同时，实现了显著的效率提升。

量化指标: 在 DTU 数据集 (3 视图) 上，Sparse2DGS 实现了最低的倒角距离误差 (**1.13 CD error**)，优于 2DGS (2.81)、GOF (2.82) 和 PGSR (2.08)。同时，其训练速度比基于 NeRF 的微调方法 (如 SparseNeuS) 快 **2 倍**。

1.4 method

1.4.1 总体架构 (Overview)

Sparse2DGS 的核心目标是解决从稀疏视角 (Sparse Views) 进行几何优化时的病态问题 (Ill-posed nature)。系统输入为 N 张稀疏视角的姿态图像 $\{I_i\}_{i=1}^N$ ，输出为精确的 3D 场景表面。



图表说明：Figure 2 是理解本方法全貌的最重要图表。它展示了从 MVS 初始化到最终高斯更新的完整流水线。左侧展示了输入处理，中间展示了特征监督与正则化，右上角展示了独特的更新策略。

为了实现这一目标，作者设计了一个包含四个关键阶段的完整流水线：

1. 初始化阶段：利用 MVS 点云打破稀疏限制

传统的 2DGS 依赖于 Structure-from-Motion (SfM) 生成的点云进行初始化。然而，在稀疏视角下，SfM 点云非常稀疏，无法为高斯基元提供足够的初始几何信息。

- **策略：**Sparse2DGS 首先利用多视图立体视觉 (MVS) 网络（具体使用 CLMVSNet [31]）生成的点云来初始化高斯基元。
- **目的：**为系统提供一个密集的几何起点，而非从零开始或从稀疏点开始。

2. 监督阶段：几何增强与抗过拟合

在有了初始点云后，需要对高斯基元进行优化。

- **策略：**模型将 MVS 提取的几何特征图 (Feature Maps) 采样到高斯基元中进行 "Feature Splatting"。
- **关键约束：**在优化过程中，固定特征值 (Feature) 和 RGB 颜色值 (Color)。
- **目的：**这是为了防止在稀疏视角下，模型为了“凑”出正确的颜色而牺牲几何形状（即外观过拟合）。固定参数迫使模型专注于优化几何属性。

3. 正则化阶段：基于重参数化的统一优化

如何约束高斯基元的形状（旋转、缩放）符合真实几何？

- **策略：**引入了一种基于重参数化的圆盘采样策略 (Reparameterization-based Disk Sampling)。
- **机制：**将高斯基元的旋转 R、缩放 S 和位置 p 重新公式化为“采样点”。这样，就可以利用基于点的跨视图一致性损失来同时优化这些复杂的几何属性。

4. 更新阶段：选择性更新替代盲目致密化

标准 Gaussian Splatting 会不断分裂和克隆 (Densify) 高斯球。但在稀疏视角下，盲目增加高斯数量会导致伪影。

- **策略**：提出了 **选择性高斯更新 (Selective Gaussian Update)**。
- **机制**：利用渲染出的几何线索（如深度图、法线图）来指导高斯基元的位置更新，而不是简单地增加数量。

1.4.2 MVS 初始化与几何增强监督

这一节解决的是“从哪里开始优化”以及“如何保证优化方向正确”的问题。

A. 基于 MVS 的初始化 (MVS Initialization)

1. **遇到的问题**：标准的 2D Gaussian Splatting (2DGS) 依赖于 Structure-from-Motion (SfM) 产生的 3D 点云进行初始化。但在稀疏视角 (Sparse Views) 设定下，SfM 只能产生极其稀疏的点，导致高斯基元数量不足，无法构建出高质量的初始表面。
2. **解决方案**：Sparse2DGS 选择使用**多视图立体视觉 (MVS)** 方法来替代 SfM 初始化
 - **具体选择**：选用了无监督的 **CLMVSNet [31]**。之所以选择无监督方法，是因为它不需要真值深度图 (Ground Truth Depth) 进行训练，在现实场景中更具灵活性。
 - **操作流程**：输入稀疏图像 $\{I_i\}$ 到 MVS 网络。网络推断出每张图的密集深度图 $\{D_i\}$ 。将这些深度图反向投影 (Back-project) 并融合，形成一个密集的点云。将这个 MVS 点云直接作为高斯基元的初始位置 p 。

B. 几何增强监督 (Geometrically Enhanced Supervision)

1. **遇到的问题**：在 3D/2DGS 中，通常是通过图像的 RGB 重建误差来同时优化几何（位置、形状）和外观（颜色）。在稀疏视角下，这是一个“陷阱”。模型会为了“过拟合”仅有的几张图片，强行调整颜色来降低误差，而牺牲了几何结构的正确性（即：虽然图看起来是对的，但 3D 模型是乱的）。
2. **解决方案**：作者提出了一种“几何增强监督”策略，核心思想是：**引入 MVS 特征，并“冻结”外观参数**。

引入 MVS 特征 (Feature Splatting)：除了颜色，每个高斯基元还被赋予了一个特征向量 f_{ij} 。这个特征直接取自 CLMVSNet 的特征金字塔网络 (FPN)，具有很强的几何表达能力。公式如下，

$$f_{ij} = F(\pi(P_{ij})), \quad c_{ij} = I(\pi(P_{ij}))$$

- P_{ij} (3D 点): 这是 MVS 初始化得到的 3D 点的位置。
- $\pi(P_{ij})$ (投影): 把这个 3D 点投影回 2D 屏幕，找到它对应在哪张图片的哪个像素坐标 (u, v) 上。
- **I 和 F (查表)**:
 - $I(\dots)$ 去原始 RGB 图片里查这个坐标的颜色，赋值给 c_{ij} (颜色)。

- $F(\dots)$ 去 MVS 的特征图 (Feature Map) 里查这个坐标的特征向量, 赋值给 f_{ij} (特征)。

其中 π 是投影操作。表示从3D空间到2D图像平面的投影。本论文进一步使用tile-based的渲染器与RGB渲染一起将特征渲染到图像平面。我们展示了给定沿一条射线的 M 个高斯基元 (为简洁起见, 省略了索引 i,j) 的渲染过程。

$$\{\hat{\mathbf{c}}, \hat{\mathbf{f}}\} = \sum_{m=1}^M \{\mathbf{c}_m, \mathbf{f}_m\} \cdot \alpha_m \prod_{o=1}^{m-1} (1 - \alpha_o)$$

- \sum (累加): 想象一条光线穿过场景, 它会穿过一串高斯球 (共 M 个)。我们需要把这一串球的颜色和特征叠加起来。
- $\{\mathbf{c}_m, \mathbf{f}_m\}$: 这一串球里第 m 个球刚才被赋予的颜色和特征。
- α_m (不透明度): 第 m 个球有多不透明 (挡住多少光)。
- $\prod(1 - \alpha_o)$ (透射率): 这一项代表光线到达第 m 个球之前, 还剩多少光没被前面的球挡住。如果前面的球完全不透明, 后面的就看不到了。
- $\{\hat{\mathbf{c}}, \hat{\mathbf{f}}\}$ (结果): 最终渲染出来的 2D 像素的颜色和特征图。

“冻结”策略 (Fixing Strategy): 为了强制模型去学习几何, 论文明确指出: 在优化过程中, 特征值 \mathbf{f} 和颜色值 \mathbf{c} 是固定的 (Fixed)。

这意味着模型无法通过“调色”来作弊, 只能老老实实在地移动和变形高斯基元, 才能让渲染出的结果与输入匹配。

监督损失: 系统使用基于图块的光栅化器 (Tile-based renderer) 将这些特征渲染成特征图 $\hat{\mathbf{f}}$, 并计算其与 MVS 原始特征图之间的余弦相似度损失 (Cosine Similarity Loss):

添加 TeX 公式

\hat{f}_{ij} : 这是通过上一条公式渲染出来的 2D 特征图中, 坐标 (i, j) 处的特征向量。它是由视线路径上的那些高斯球叠加而成的。

f_{ij} : 这是直接从 MVS 网络 (CLMVSNet) 提取出来的原始特征图中, 坐标 (i, j) 处的特征向量。它代表了该像素点在几何上“应该”是什么样子。

- 分式部分 ($\frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|}$): 这是标准的余弦相似度 (Cosine Similarity) 公式。它计算两个向量夹角的余弦值。
 - 如果两个特征向量方向一致 (非常相似), 值为 1。
 - 如果方向相反或无关, 值会很小甚至是负数。

- 完美匹配： $1 - 1 = 0$ (损失为0)。
- 完全不匹配： $1 - (-1) = 2$ (损失很大)。

这个损失函数起到了至关重要的作用：

- 如果高斯基元的属性（颜色、特征）是可变的，模型可能会“作弊”：它不移动高斯球的位置，而是直接修改高斯球上的特征值来通过这个考试。
- 但是，Sparse2DGS 强制固定了高斯球上的特征值。
- 结果：为了降低这个损失 \mathcal{L}_{fea} ，模型别无选择，只能移动、旋转或缩放高斯基元。只有把带有正确特征的高斯球，移动到屏幕上正确的位置（像素 (i, j)），渲染出来的 \hat{f}_{ij} 才能和 MVS 的 f_{ij} 对上号。

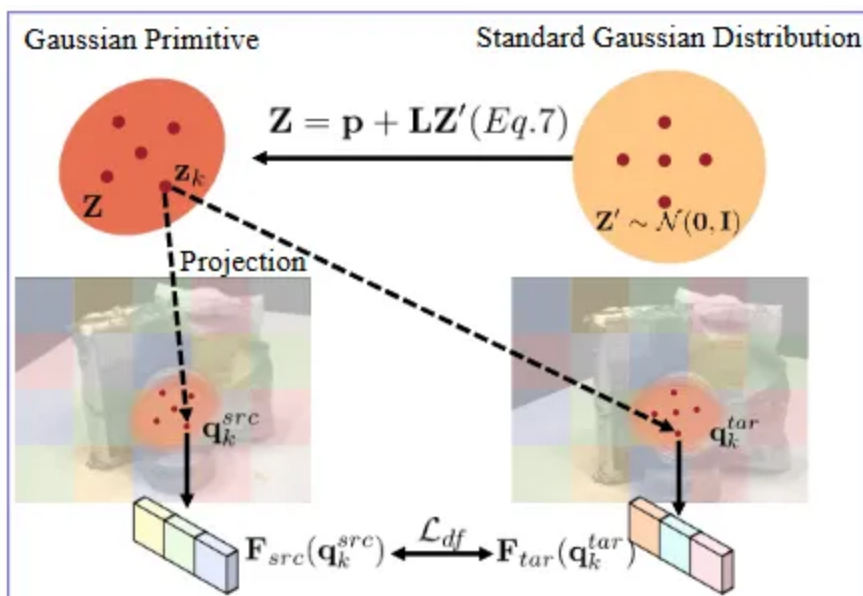
1.4.3 直接高斯基元正则化

1. 核心挑战：形状难以优化

在稀疏视角下，如果我们只约束高斯基元的中心位置 (\mathbf{p}) 是不够的。高斯基元还有旋转 (\mathbf{R}) 和缩放 (\mathbf{S}) 属性，它们决定了表面的局部朝向和覆盖范围。传统的跨视图一致性方法通常只看“点”的位置是否一致，无法告诉模型：“你的这个高斯圆盘太扁了”或者“朝向歪了”。

2. 解决方案：基于重参数化的圆盘采样

为了能同时优化 \mathbf{R} 和 \mathbf{S} ，作者提出了一种策略：把高斯的形状“翻译”成点集。



这一过程可以分为三步：

1. **标准采样**：首先，从一个标准的正态分布 $\mathcal{N}(0, I)$ 中采样出一组点 Z' 。这些点分布在一个标准的圆里，与具体的高斯参数无关。
2. **重参数化变换 (关键步骤)**：利用当前高斯基元的属性，将这些标准点“变身”到世界坐标系中：

$$Z = p + LZ', \quad \text{其中 } L = RS$$

- 这里 p 是位置， R 是旋转矩阵， S 是缩放矩阵。
 - **数学意义**：通过这个公式，生成点 Z 的位置就直接挂钩于 R 和 S 。如果后续发现点 Z 的位置不对，梯度 (Gradient) 就能顺着公式传导回去，告诉模型去修改 R 和 S 。
3. **投影与比对**：将变换后的点 Z 投影到源视图 (Source View) 和目标视图 (Target View) 上，得到像素坐标 q^{src} 和 q^{tar} 。

3. 正则化损失函数 (Regularization Loss)

基于上述采样点，系统计算两部分损失来约束几何：

- A. 跨视图特征一致性损失 (\mathcal{L}_{df})** 检查采样点在不同视角下看到的“特征”是否长得一样。

$$\mathcal{L}_{df} = \frac{1}{K} \sum_{k=1}^K (1 - \text{CosineSimilarity}(F_{src}(q_k^{src}), F_{tar}(q_k^{tar})))$$

如果特征不一致，说明高斯的形状或位置有误，需要调整。

- B. 法线监督损失 (\mathcal{L}_{dn})** 由于 2D 高斯是平面的（最后一个缩放维度为 0），其法线方向非常重要。模型利用 MVS 估计的法线 n 来监督高斯基元的旋转 R （具体取 R 的第三列作为法线向量）：

$$\mathcal{L}_{dn} = 1 - R[\dots, 2]^\top n$$

这确保了高斯圆盘的朝向与物体表面的法线一致。

总正则化损失： $\mathcal{L}_{dr} = \mathcal{L}_{df} + \mathcal{L}_{dn}$

1.4.4 选择性高斯更新

1. 核心挑战：盲目致密化导致崩溃

标准的 Gaussian Splatting 有一个核心机制叫“自适应密度控制” (Adaptive Density Control)，也就是大家熟知的“分裂 (Split)”和“克隆 (Clone)”。

- **在密集视角下**：这很好用，哪里细节不够就加点。
- **在稀疏视角下 (本论文场景)**：这很危险。因为视角太少，如果盲目增加高斯点的数量，这些新长出

来的点往往缺乏约束，会变成漂浮在空中的噪声（Floaters）或者错误的几何结构，导致重建质量严重下降。

2. 解决方案：选择性更新 (Selective Update)

Sparse2DGS 摒弃了在病态区域盲目增加点的做法，而是提出了“优胜劣汰，位置修正”的策略。核心思想是：**不增加数量，而是让现有的点跑得更准**。我们利用渲染出来的几何线索（Depth/Normal）来检查当前的高斯点是不是在正确的位置上，如果发现渲染结果比高斯点自己更好，就把高斯点挪过去。

3. 具体实施步骤

这一过程主要依赖 **Patch Warping (图块变换)** 技术来进行质量评估。

第一步：两路竞技 (Two-way Evaluation) 对于一个高斯基元，系统尝试用两种方式将源视图 (Source View) 的图像块变换到目标视图 (Target View)：

1. 选手 A (高斯基元自身)：利用该高斯基元当前的位置 p 和方向 R 进行单应性变换 (Homography Warping)。得分记为： NCC_G (Normalized Cross-Correlation)
2. 选手 B (渲染几何线索)：利用该像素位置上**渲染出的深度和法线**，反投影得到一个临时的 3D 点，再用这个点进行变换。得分记为： NCC_R

通过以下公式来进行评估

$$NCC(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}$$

其中， x_l 和 y_l 分别表示源图像和目标图像的块 x 和 y 中的像素。

计算 NCC_G (Gaussian):

- 使用 **高斯基元自身** 的位置 p 和方向 R 来计算单应性变换矩阵。
- 用这个矩阵把源图像块变换过去，得到 x_G 。
- 计算 $NCC(x_G, \text{target_patch})$ ，结果记为 NCC_G 。

计算 NCC_R (Rendered):

- 使用 **渲染出来的深度和法线**（即模型目前“看”到的几何形状）来计算变换。
- 用这个几何信息把源图像块变换过去，得到 x_R 。
- 计算 $NCC(x_R, \text{target_patch})$ ，结果记为 NCC_R 。

第二步：优胜劣汰 (Update Decision) 系统比较这两个得分（NCC 越高代表图像块匹配得越完美，几何越准）：

- 规则：如果 $NCC_R > NCC_G$
- 含义：说明“渲染出来的几何位置”比“高斯点当前的位置”更靠谱。
- 动作：将该高斯基元的位置 **更新 (Move)** 到渲染深度对应的 3D 位置上。

第三步：执行频率：这种更新不是每一步都做，而是每 100 个训练步 执行一次。在更新之后，所有高斯参数继续通过正常的梯度下降进行优化。

这就像是一个“自我纠错”机制。MVS 初始化虽然好，但也不完美。系统在训练中会不断问自己：“是我现在的身体位置准，还是我眼睛看到的（渲染的）位置准？”如果是后者准，就挪过去。

1.4.5 训练损失 (Training Loss)

为了训练出既有正确外观又有准确几何的模型，Sparse2DGS 的总损失函数 \mathcal{L} 由五大部分组成。

1. 总公式

$$\mathcal{L} = \mathcal{L}_{rgb} + \lambda_1 \mathcal{L}_d + \lambda_2 \mathcal{L}_n + \lambda_3 \mathcal{L}_{dr} + \lambda_4 \mathcal{L}_{fea}$$

2. 逐项拆解

这个公式可以分为两类：**继承自 2DGS 的基础损失** 和 **本论文提出的创新损失**。

A. 基础损失 (来自原始 2DGS [11])

这部分是为了保证基本的渲染质量和几何连续性。

- \mathcal{L}_{rgb} (**图像重建损失**)：保证渲染出来的 RGB 图像和真实照片长得一样。
- \mathcal{L}_d (**深度畸变损失 Depth Distortion Loss**)：用于消除渲染深度图中不自然的伪影（如漂浮物），权重 λ_1 通常很大。
- \mathcal{L}_n (**深度-法线一致性损失 Depth-Normal Consistency Loss**)：保证由深度图推导出的几何结构与预测的法线图保持一致，使表面更平滑。

B. 创新损失 (Sparse2DGS 核心贡献)

这部分正是我们在 1.4.2 和 1.4.3 中详细讲解的内容，专门用于解决稀疏视角的病态问题。

- \mathcal{L}_{dr} (**直接高斯基元正则化损失**)：对应 1.4.3 节。
 - 它包含了跨视图特征一致性损失 \mathcal{L}_{df} 和法线监督损失 \mathcal{L}_{dn} 。

- **作用：**强迫高斯基元的形状（旋转、缩放）符合真实的几何表面。
- \mathcal{L}_{fea} (MVS 特征增强损失)：对应 1.4.2 节。
 - 它计算渲染特征与 MVS 特征的余弦相似度。
 - **作用：**配合“参数冻结”，强迫高斯基元移动到正确的 3D 位置。

3. 超参数设置 (权重)

论文在实验部分的 4.2 实施细节 中给出了这些 λ 的具体数值，这对于复现非常重要：

- $\lambda_1 = 1000$ (深度畸变损失权重，非常高，说明对深度的平滑性要求极严)
- $\lambda_2 = 0.05$ (法线一致性权重)
- $\lambda_3 = 1$ (本文提出的正则化权重)
- $\lambda_4 = 0.2$ (本文提出的特征监督权重)

1.5 实验结果与分析 (Experiments)

为了评估 Sparse2DGS 在稀疏视角重建任务中的性能，作者在 DTU [1] 和 BlendedMVS [37] 两个广泛使用的数据集上进行了实验。

- **实验设置：**遵循 SparseNeuS 的设置，在 DTU 数据集上选择了 15 个场景，仅使用 3 个视角 (views 23, 24, 33) 进行训练，图像分辨率为 576×768 。
- **评估指标：**主要使用 Chamfer Distance (CD) 来衡量重建网格与真值点云之间的几何误差，数值越低越好

1.5.1 主要结果 (Main Results)

A. 稀疏视角下的量化性能

在极具挑战性的 3 视角设置下，Sparse2DGS 展现出了显著的性能优势，大幅超越了其他高斯溅射方法。

- **总体表现：**在 DTU 数据集上，Sparse2DGS 实现了最低的平均误差，Chamfer Distance 仅为 1.13 mm。这不仅显著优于基线方法 2DGS (2.81 mm)，也优于当前的 SOTA 高斯方法 PGSR (2.08 mm) 和 GOF (2.82 mm)。
- **跨范式对比：**与基于 NeRF 的方法相比，Sparse2DGS 也表现出了极强的竞争力。它超越了首个稀疏视角 NeRF 方法 SparseNeuS (1.27 mm)，并与 VolRecon (1.38 mm) 等先进方法相当甚至更好

。

- **具体数据：**根据 Table 1，Sparse2DGS 在多个场景（如 Scan 24, 69, 114 等）中均取得了极低的误差值，证明了其几何增强策略的有效性。

Scan	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
Colmap [26]	0.90	2.89	1.63	1.08	2.18	1.94	1.61	1.30	2.34	1.28	1.10	1.42	0.76	1.17	1.14	1.52
CLMVSNet [31]	1.17	2.52	1.68	0.89	1.51	1.82	0.99	1.29	1.52	0.78	0.88	1.10	0.64	1.04	1.12	1.26
Neus [28]	4.57	4.49	3.97	4.32	4.63	1.95	4.68	3.83	4.15	2.50	1.52	6.47	1.26	5.57	6.11	4.00
Volsdf [38]	4.03	4.21	6.12	0.91	8.24	1.73	2.74	1.82	5.14	3.09	2.08	4.81	0.60	3.51	2.18	3.41
VolRecon [25]	1.20	2.59	1.56	1.08	1.43	1.92	1.11	1.48	1.42	1.05	1.19	1.38	0.74	1.23	1.27	1.38
SparseNeus [20]	1.29	2.27	1.57	0.88	1.61	1.86	1.06	1.27	1.42	1.07	0.99	0.87	0.54	1.15	1.18	1.27
Retr [19]	1.05	2.31	1.44	0.98	1.18	1.52	0.88	1.35	1.30	0.87	1.07	0.77	0.59	1.05	1.12	1.17
C2F2Neus [34]	1.12	2.42	1.40	0.75	1.41	1.77	0.85	1.16	1.26	0.76	0.91	0.60	0.46	0.88	0.92	1.11
Sparsecraft [40]	1.17	1.74	1.80	0.70	1.19	1.53	0.83	1.05	1.42	0.78	0.80	0.56	0.44	0.77	0.84	1.04
NeuSurf [12]	0.78	2.35	1.55	0.75	1.04	1.68	0.60	1.14	0.98	0.70	0.74	0.49	0.39	0.75	0.86	0.99
2DGS [11]	3.14	3.79	2.31	1.51	5.34	2.22	1.85	2.79	3.62	1.63	3.05	4.11	1.05	3.35	2.44	2.81
GOF [43]	3.27	5.05	2.77	2.04	3.84	2.61	2.23	2.65	3.30	1.65	2.58	4.44	1.16	2.29	2.39	2.82
PGSR [4]	3.36	3.28	2.75	1.27	5.15	1.84	0.88	1.79	3.49	1.19	1.86	1.11	0.61	1.09	1.52	2.08
Ours	1.05	2.35	1.38	0.83	1.37	1.45	0.84	1.16	1.43	0.74	0.85	0.84	0.57	0.95	1.01	1.13

Table 1. **Quantitative results on DTU dataset.** Our method achieves the best performance among Gaussian Splatting surface reconstruction approaches and demonstrates competitive performance to previous state-of-the-art NeRF-based methods.

B. 视觉质量与完整性

除了数值上的优势，Sparse2DGS 在重建的视觉质量上也解决了此前方法的痛点。

- **结构完整性：**如图 4 (Figure 4) 所示，在稀疏视角下，普通的 Gaussian Splatting 方法（如 2DGS, GOF）往往会出现严重的噪声 (Noisy) 和 区域缺失 (Missing areas)。相比之下，Sparse2DGS 能够恢复出完整且连贯的场景结构。
- **细节保留：**与隐式重建方法（如 SparseNeuS）相比，Sparse2DGS 捕捉到了更多的表面细节。例如在 BlendedMVS 数据集的定性结果中 (Figure 5)，Sparse2DGS 重建的网格在真实世界场景中展现出了更高的准确度和细节丰富度。

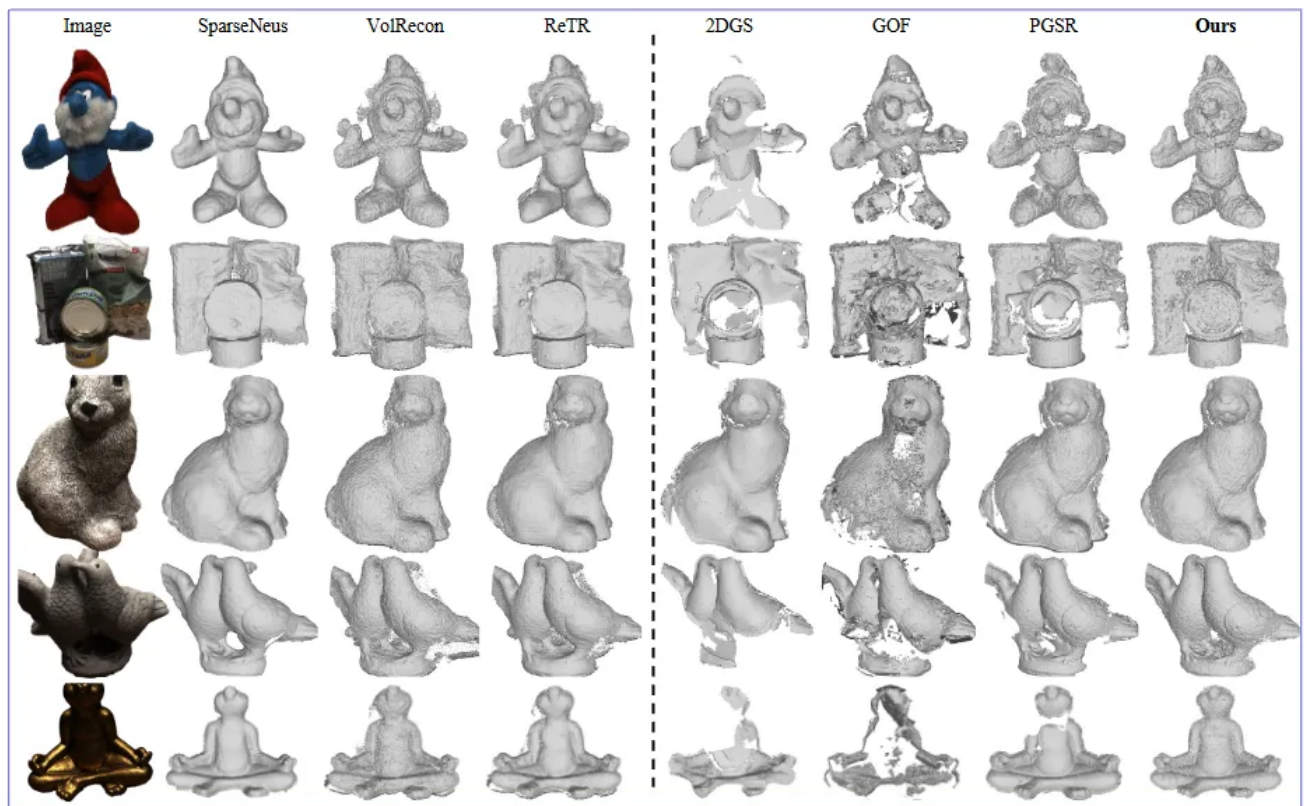


Figure 4. DTU surface reconstruction results. Our method achieves more complete reconstructions with finer details.

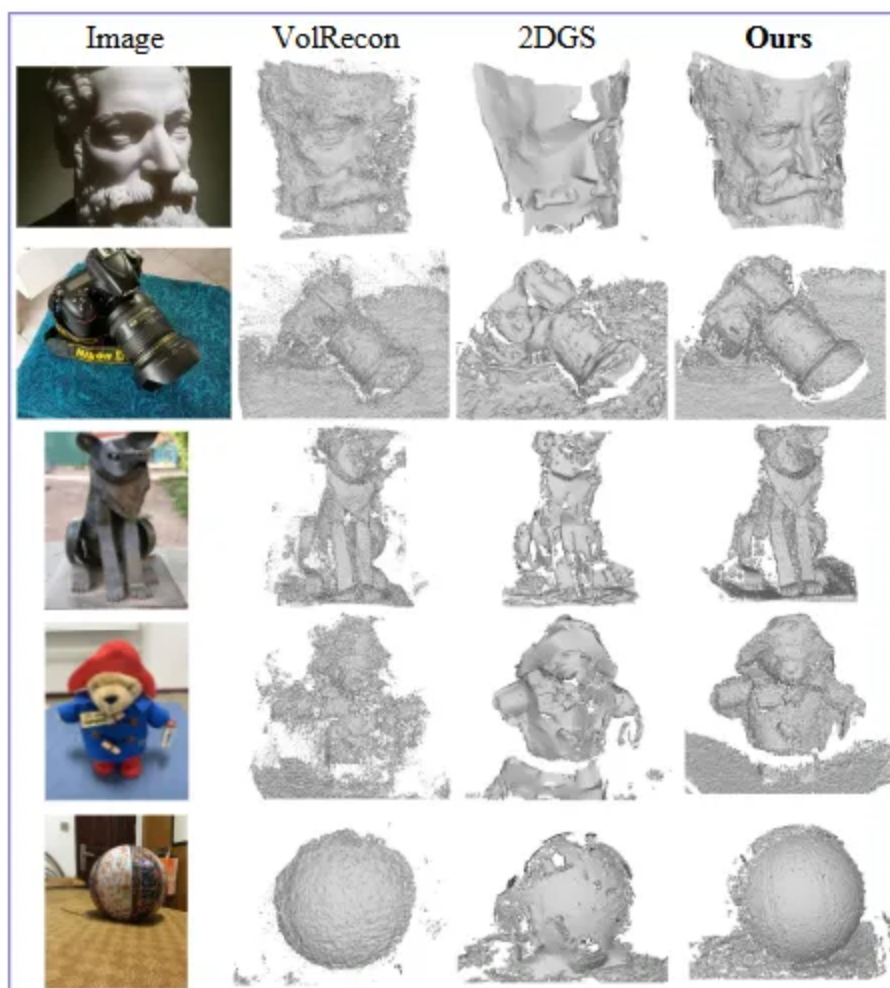


Figure 5. **BlendedMVS reconstruction results.** Our method achieves more complete and detailed reconstructions.

C. 训练效率

这是 Sparse2DGS 的核心优势之一。它在提升几何质量的同时，保持了 3DGS 家族的高效特性。

- **训练时间：**根据 Table 7 的对比，Sparse2DGS 的训练时间仅需 **10 分钟**。
- **速度对比：**这比基于 NeRF 的微调方法 SparseNeuS（25 分钟）快了 **2 倍**。与 NeuSurf（10 小时）相比，更是快了 **60 倍**。这证明了该方法在测试时优化（Test-time optimization）场景下的实用价值。

Methods	SparseNeus [20]	NeuSurf [12]	Ours
Chamfer Distance	1.27	0.99	1.13
Training Time	25mins	10 hours	10mins

Table 7. **Performance and efficiency.** Our method achieves comparable performance while requiring the least training time.

1.5.2 消融实验与分析

作者在 DTU 数据集上进行了一系列严格的消融实验，以验证架构设计的有效性，并回答了关于初始化策略和参数设置的关键问题。

A. 核心组件有效性验证

如 Table 2 所示，论文采用逐步叠加模块的方式，验证了每个组件对最终性能（CD Error 越低越好）的贡献：

- **基线 (Baseline):** 仅使用 MVS 点云初始化，CD 误差为 1.273。
- **几何增强监督 (+Fixed Color & Feat. Splat.):** 通过固定颜色参数和引入 MVS 特征监督，模型的准确性 (Accuracy) 和完整性 (Completeness) 均得到提升，CD 误差降至 1.201 (Row e)。
- **直接高斯基元正则化 (+DGPR):** 引入重参数化采样正则化后，增强了对高斯形状的约束，CD 误差进一步降至 1.142 (Row f)。
- **选择性高斯更新 (+SGU):** 最后加入基于渲染线索的更新策略，模型达到了最佳性能，CD 误差为 1.125 (Row h)。

	2DGS	MVS Init.	Fixed Color	Feat. Splat.	DGPR	SGU	CD ↓
(a)	✓						2.813
(b)	✓	✓					1.273
(c)	✓	✓	✓				1.212
(d)	✓	✓		✓			1.224
(e)	✓	✓	✓	✓			1.201
(f)	✓	✓	✓	✓	✓		1.142
(g)	✓	✓	✓	✓		✓	1.152
(h)	✓	✓	✓	✓	✓	✓	1.125

Table 2. Ablation on DTU [1] dataset. DGPR and SGU represent proposed Direct Gaussian Primitive Regularization and Selective Gaussian Update respectively. Each module improves upon the baseline, leading to the best final results.

B. 为什么简单的 "MVS + 2DGS" 不够好？

这是一个非常关键的发现。论文在 Table 3 中对比了不同的初始化与组合策略：

- **简单的组合 (Plain Combination):** 如果直接把 CLMVSNet 的点云扔给 2DGS 跑，最终结果 (CD: 1.27) 甚至略微差于 CLMVSNet 原始点云 (CD: 1.26)。这说明在稀疏视角下，简单的组合

不仅没进步，反而因为几何优化的病态性质导致了退步。

- **Sparse2DGS 的突破**：本文的方法通过几何优先的增强策略，成功打破了这一瓶颈，最终结果 (CD: 1.13) 显著优于 MVS 原始点云和简单的组合。

Methods	2DGS [11]	PGSR [4]	CLMVS. [31]	Ours
3 view SfM Point Cloud	2.81	2.08	-	-
49/64 views SfM Point Cloud	1.65	1.53	-	-
CLMVSNet Point Cloud	1.27	1.38	1.26	1.13

Table 3. Comparison of initialization strategies. Initializing with MVS point cloud improves reconstruction performance. However, a plain combination (Chamfer Distance, CD:1.27) between 2DGS and CLMVSNet fails to surpass CLMVSNet results (CD: 1.26). Our method enhances both MVS and 2DGS through geometry-prioritized optimization.

C. 关键设计选择 (Key Design Choices)

论文进一步分析了外观学习策略和更新策略对性能的影响。

- **固定颜色的必要性 (Fixed Color)**：在 Table 4 中，作者尝试了使用可学习的球谐函数 (SH)。结果表明，只要让颜色参与优化（即使 SH 阶数很低），几何质量就会下降。**固定颜色** 策略被证明最有利于学习正确的几何结构。
- **更新策略对比 (SGU vs. Density Control)**：Table 5 显示，在稀疏视角下，原始 2DGS 的“自适应密度控制”（即分裂和克隆）会导致性能恶化（CD 从 1.20 变差为 1.23），因为产生了错误的几何结构。而本文提出的 **选择性高斯更新 (SGU)** 成功提升了重建质量。

Settings	Accuracy↓	Completion↓	Average↓
SH=3	0.83	1.71	1.27
SH=2	0.84	1.68	1.26
SH=1	0.84	1.64	1.24
Fixed Color	0.81	1.61	1.21

Table 4. Ablation study on appearance learning. Fixing the color helps in learning better geometry.

Methods	Accuracy↓	Completeness↓	Average↓
Baseline	0.790	1.612	1.201
Adaptive Density Control [14]	0.814	1.654	1.234
Selective Gaussian Update	0.760	1.544	1.152

Table 5. Ablation on DTU [1] dataset. The proposed Selective Gaussian Update strategy enhances reconstruction accuracy and completeness compared to the generic Adaptive Density Control in sparse views.

1.6 Limitations

尽管 Sparse2DGS 在稀疏视角重建上取得了显著进展，但其性能仍受到初始化和监督源的制约：

- **对 MVS 性能的依赖 (Dependence on MVS Performance):** Sparse2DGS 的核心策略是利用 MVS 进行初始化和几何监督。然而，MVS 方法在处理 **遮挡 (Occlusions)** 区域时，不可避免地会遇到性能下降的问题。
- **误差传播 (Error Propagation):** 由于上述依赖性，MVS 在遮挡区域产生的几何错误会 **传播 (Propagate)** 到 Sparse2DGS 的优化过程中。这意味着如果 MVS 在某些区域（如深度不连续或被遮挡处）估算错误，Sparse2DGS 可能会继承这些错误，从而导致潜在的重建性能下降。

1.7 conclusion

在本文中，我们提出 Sparse2DGS，通过几何优先增强方案解决稀疏视角下病态几何优化的挑战。我们的方法不仅能重建准确完整的场景表面，还比基于 NeRF 的方法实现了更高的效率。局限性：尽管多视图立体（MVS）方法在密集匹配中效果显著，但在遮挡情况下不可避免地会出现退化。遮挡区域的误差可能传播到我们的方法中，导致潜在的性能下降。

