

综述：3D 高斯泼溅的发展及其在自动驾驶、SLAM 等领域的应用

周钰添

2024 年 3 月 1 日

1 引言

1.1 高斯泼溅 (3D Gaussian Splatting) 基本概述

3D 高斯泼溅 (3D Gaussian Splatting) [3] 是一种用于高效和高质量渲染 3D 场景的显式辐射场技术，由 Kerbl 等人在 2023 年 8 月提出。相比于 NeRF[4]，新技术在渲染速度上大幅胜出，兼顾了较高的训练、渲染速度和渲染质量。因此，这项技术迅速地在计算机视觉 (CV) 和计算机图形学 (CG) 领域引起热烈的研究，相关的论文数量迅速增加 [1]，仅半年的时间已经有超过 100 篇相关论文。本文基于<https://github.com/MrNeRF/awesome-3D-gaussian-splatting>所收集的相关论文和博客，对 3D 高斯泼溅的新方法产生初步的了解和认识。

3D 高斯泼溅具有很深远的研究价值，因为他同时具有高质量和快速实时渲染的特点，训练速度也相对较快。这种表示方法创新性地利用数百万个 3D 高斯函数所表示的椭球体来建立了 3D 世界的模型，其本质比较简单，每个椭球体都是一个 3D 的高斯分布，具有独特的参数。这些参数的调整可以匹配输入的图像，通过不同视角的图像输入，来建立一个 3D 的现实场景。相比于使用神经网络的 NeRF 方法，3D 高斯函数的可微性让 3D 高斯泼溅能够进行快速的优化，同时，3D 高斯泼溅的光栅化速度也要比 NeRF 的光线追踪方法更快。最终，3D 高斯泼溅通过非神经网络的技术实现了半小时内完成训练、与以前方法相媲美的质量和实时的渲染画面。

在后续的研究中，人们致力于解决 3D 高斯泼溅的一系列缺点，即使用更小的内存来训练和储存模型 (Compression) [5][6][7]、改善使用稀疏点

云初始化所呈现的最终场景质量 [9][13]。此外，人们尝试将 3D 高斯泼溅技术用于自动驾驶、即时定位与地图构建（Simultaneous Localization and Mapping，简称 SLAM）等方面，取得了良好的进展。在后文，我将分别阐述这几个方面的成果。

1.2 综述的内容范围

本文将讨论 3D 高斯泼溅（3D Gaussian Splatting）的提出和发展，并讨论这项新技术在自动驾驶、SLAM 等领域的应用和未来发展趋势。

- 在第 3 节，我会解释 3D 高斯泼溅（3D Gaussian Splatting）技术的基本原理和关键概念，探讨这项技术的创新性。
- 在第 4 节和第 5 节，我将讨论人们如何克服使用巨大内存的缺点和在稀疏点云的初始化情况下如何呈现较好的结果，以达到更低的内存使用和更高质量的成像。
- 在第 6 节和第 7 节，我将分别讨论 3D 高斯泼溅技术及其优化在自动驾驶和 SLAM 领域的探索。

2 研究背景

2.1 辐射场 (Radiance Fields)

辐射场是对三维空间中光分布的一种表示，它捕捉了光与环境中的表面和材料相互作用的方式。辐射场有两种主要表示形式：一种是隐式表示，例如 NeRF[4]，它使用神经网络进行直观但计算上较为昂贵的渲染；另一种是显式表示，例如 grid 和今天的 3D 高斯泼溅，它使用离散结构以便更快地访问，但代价是更高的内存使用。辐射场是场景渲染中的关键概念。

2.2 神经辐射场 (Neural Radiance Fields)

神经辐射场（Neural Radiance Fields，简称 NeRF）[4] 是一种用于生成高质量三维重建模型的计算机视觉技术。该技术利用深度学习从多个视角的图像中提取对象的几何形状和纹理信息，并以此生成一个连续的三维辐射场。由此产生的三维模型在任意角度和距离下呈现出高度逼真的外观。

NeRF 技术在计算机图形学、虚拟现实、增强现实等领域有着广泛的应用前景。

NeRF 技术的优势在于生成的三维模型质量高、逼真度强，能够以高质量呈现真实物体表面和纹理细节，而且可以从任意数量的输入图像中生成三维模型，无需对输入进行特定处理或标记。然而，NeRF 技术也存在一些缺点，例如对于训练需要大量的计算资源和时间，并且在处理大规模场景和复杂光照条件时面临一定挑战。另外，由于 NeRF 是基于视图合成的技术，生成模型时需要确保输入视角足够广泛和充分，否则可能导致模型中存在遮挡和空洞的问题。

2.3 场景重建和渲染

场景重建是通过分析一组图像或其他数据来生成三维模型的过程。相对而言，渲染是一个更具体的术语，强调将计算机可读信息（例如场景中的 3D 对象）转换为基于像素的图像，使其可视化。在技术发展的早期，一些方法利用光场来生成逼真的图像。随着时间的推移，运动结构 (SfM) [8] 和多视图立体 (MVS) [2] 算法通过从图像序列中估计 3D 结构进一步推动了场景重建的领域。这些历史性的方法为更复杂的场景重建和渲染技术打下了基础。

3 3D 高斯泼溅 (3D Gaussian Splatting) 基本原理

3D 高斯泼溅 (3D Gaussian Splatting) 的解决方案由 3 个组件构成，分别是引入了 3D 高斯函数作为一种灵活且表达力强的场景表示方式、3D 高斯函数参数的优化方法和实时渲染方案。我将着重描述前两个步骤。

3.1 3D 高斯函数的表示方式

3D 高斯函数的形式如下：

首先，我们用 $V = [a, b, c]^T$ 来表示 3 个维度， $p(x)$ 是 $N(0, 1)$ 的一维高斯分布。那么 3D 高斯分布的概率密度函数可以这么表示

$$p(V) = p(a)p(b)p(c) = \frac{1}{(2\pi)^{3/2}} e^{(-\frac{a^2+b^2+c^2}{2})} = \frac{1}{(2\pi)^{3/2}} e^{(-\frac{1}{2}V^TV)} \quad (1)$$

以上是 3 个简单的 (0,1) 正态分布叠加。在空间中，一个一般的 3D 高斯函数的属性可以由中心点向量 μ 、3 维协方差矩阵 Σ 来表示。

$$G(x) = e^{-\frac{1}{2}x^T \Sigma^{-1} x} \quad (2)$$

其中，这个公式前面的系数被忽略了，因为在后续的步骤中还设置了高斯椭圆的不透明值 α 和表示颜色的球谐系数，不同的高斯椭圆体将乘以 α 来呈现不同的透明度。

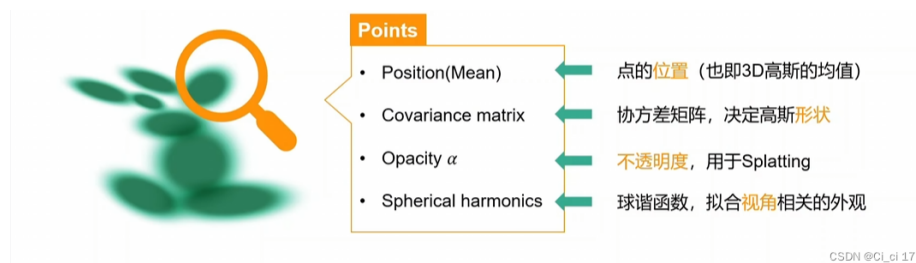


图 1: 3D 高斯泼溅技术中每个 3D 高斯函数含有的参数

当输入一组静态场景的图像，将会基于 sfM 点产生一组稀疏的点云，这就是最初被创建的一组高斯分布。3D 高斯分布可以在各个方向上呈现不同的姿态，因此可以自由地调整，也就是在优化中不断接近真实图像。

3D 高斯是一个具有连续性和可微性的函数，因此对于 3D 高斯函数的优化可以很容易地投影到二维 Splatting，允许快速的 α 混合渲染。为了避免产生无效的协方差矩阵，我们将协方差矩阵 Σ 分解为半正定的表示，即缩放矩阵 S 和旋转矩阵 R ：

$$\Sigma = R S S^T R^T \quad (3)$$

为了允许对这两个因素进行独立的优化，我们分别存储它们：一个用于缩放的三维向量 S 和一个表示旋转的四元数 q 。

3.2 3D 高斯的参数优化

优化步骤是 3D 高斯泼溅的核心。它创建了一个密集的三维高斯函数集合，精确地表示自由视图合成的场景。优化参数有：中心点位置 μ 、不透明值 α 和协方差 Σ ，此外还优化了代表每个高斯分布的颜色的球谐系数，以正确地捕捉场景中视图相关的外观。

优化的步骤是迭代地将生成的图像与训练图像进行比较，而初始化来自于原始图像基于 sfM 给出的一组稀疏点云。因此我们需要在优化这一步骤中创建高斯体、删除高斯体或移动高斯体。于是，3D 高斯泼溅使用了随机梯度下降进行优化，损失函数使用 L_1 与 L_{D-SSIM} 项相结合，其中 L_{D-SSIM} 是一种用于衡量图像相似性的指标。D-SSIM 损失则是基于 SSIM 的一个损失函数版本，通常用于深度学习任务中。SSIM 考虑了亮度、对比度和结构三个方面的相似性。而 D-SSIM 损失可以量化模型生成的图像与真实图像之间的结构相似性，帮助模型生成更加真实、保持结构相似性的图像。

$$L = (1 - \lambda)L_1 + \lambda L_{D-SSIM} \quad (4)$$

由于一些高斯体存在位置错误，在迭代中将使得 α 趋向于 0。因此必须设置阈值 ϵ_0 ，当不透明值 α 小于这个阈值时，该 3D 高斯函数被去除。在图像上某些结构复杂的位置，初始化的稀疏点云难以良好地表示，因此 3D 高斯泼溅运用了复制 (Clone) 和分裂 (Split)，当小尺度的几何图形没有被充分覆盖时，我们克隆一份相同的高斯分布；当小尺度的几何图形被一个庞大的高斯函数覆盖，就将其一分为二。

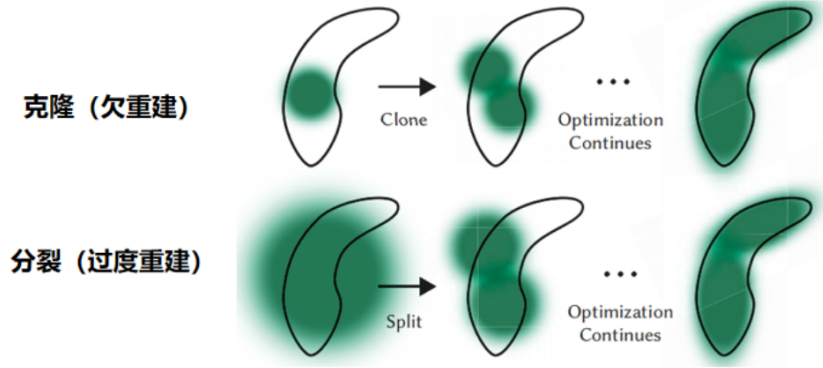


图 2: 重建的迭代过程中，3D 高斯函数进行克隆或分裂来适应图像轮廓

这种方法能够帮助 3D 高斯泼溅技术进行精确的重建。优化增加了高斯分布的 α ，同时允许我们的剔除方法去除 α 小于阈值的高斯分布。高斯分布可能会缩小或增长，并与其他分布有相当大的重叠，但我们会周期性地去除在非常大的高斯分布和那些在视图空间中有很大大足迹的高斯分布。这种策略可以很好地控制高斯分布的总数。我们模型中的高斯函数在欧几里得

空间中始终保持原型；与其他方法不同，我们对遥远或大的高斯分布不需要空间压缩、扭曲或投影策略。

4 在内存压缩方面的研究

4.1 目前情况

3D 高斯泼溅 (3D Gaussian Splatting) 使用数百万个 3D 高斯函数来还原 3 维空间场景。最初的技术方法为了实时渲染的高质量，对每一个 3D 高斯函数都储存了一组球谐系数，从而使得每一个场景都具有超过 1GB 的内存使用和更庞大的存储占用。庞大的内存使用是 3D 高斯泼溅技术应用的一个缺点，因为一台电脑一般只有不到 100GB 的内存和有限的硬盘，如果被用于大量渲染图片，那么很快就会内存溢出。

于是，人们很快开展了压缩 3D 高斯泼溅技术内存使用的研究，并且卓有成效。2023 年 9 月，就有人将一个场景的表示由 1.4GB 压缩至 100MB 以下 [6][7]，他在两篇博客中阐述了压缩的主要思路，即对原始的 3D 高斯函数的球谐系数进行**聚类**，将每个 3D 高斯函数原来存储的球谐系数改为指向聚类结果的指针。就这样，数百万个球谐系数的表示可以压缩至 4000 个到 16000 个，也就是说，去除原有的大部分重复的色彩表示，并且保留了大部分原有的图像光泽，达到了相似的质量。此时，在可以接受的质量范围内，压缩的倍率达到了 18x。

人们的压缩尝试具有相同的目标，因此我从最新的文章中了解进度。2024 年 1 月，Niedermayr 等人利用一系列方法 [5] 使得原数据集中的“bike”图像压缩了 31x[图 3]。他们主要的工作如下：

- 敏感性感知聚类：我们通过计算每个场景参数对训练图像的贡献来导出每个场景参数的敏感性度量。颜色信息和高斯参数通过敏感性感知的矢量量化编码到紧凑的码书中。
- 量化感知微调：为了恢复在聚类过程中丢失的信息，我们使用量化感知训练以降低比特率微调场景参数。
- 熵编码：3D 高斯函数沿空间填充曲线线性化，以利用场景参数的空间一致性进行熵编码和游程编码。

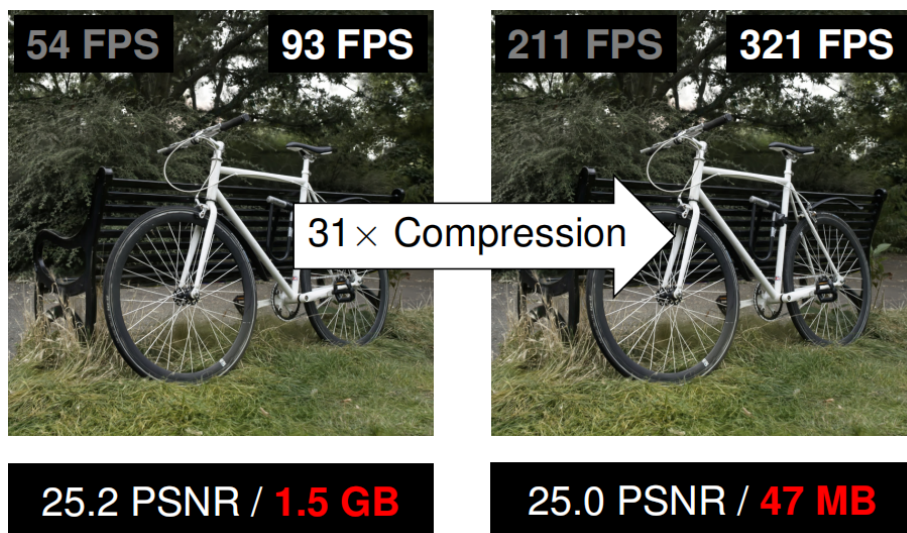


图 3: 最新的论文 [5] 将“bike”图像的结果压缩了 31x, 使得 3D 高斯泼溅技术的可用性大大增强

4.2 局限和未来展望

现有的压缩工作仍然具有一定的局限。首要的是, 当前技术无法对 3D 高斯函数的位置参数进行更强的压缩; 其次, 对于训练阶段, 目前的方法仍然需要很大的内存。

此外, 人们认为, 3D 高斯泼溅在体积场景, 即物体内部信息的表示上还有潜力。未来, 通过实现 3D 高斯泼溅技术更小的内存和储存使用, 我们能够用更多 3D 高斯函数来表示更大、更精细的场景。在开放场景的领域, 例如自动驾驶, 或许还有更多突破。

5 稀疏点云的初始化问题

在少样本视图合成问题中, 通过 2D 图像学习 3D 结构的困难性显著增加: 在少量视图中, 许多场景区域几乎没有覆盖。因此, 许多可能的不正确的 3D 表示可以正确表示 2D 训练视图, 但在渲染新视图时会产生像“floaters” (场景中高密度区域的浮动区域, 位置不规律) 之类的伪影, 或“background collapse” (当背景由接近相机的相似外观伪影表示时) 或表示中的空洞。以往的工作通过引入对渲染深度图的变化进行正则化的约束来解决这个问题。

根据描述，这个领域最近的先前工作大多基于 NeRF 构建，并因此受到长时间训练和神经网络固有的黑盒性的限制。透明度的缺乏构成了一个显著的约束，阻碍了对问题的直接处理。相反，通过精心设计的损失函数和约束间接地处理这些挑战。[9]

Xiong 等人构建了一个基于 3D 高斯斑点的少样本新视图合成技术 [9]。3D 高斯泼溅底层表示的显式性使得能够引入一个关键的新操作：直接的浮动区修剪。该操作识别由于 floaters 或背景崩溃而受到影响的渲染图像的部分，并允许直接编辑 3D 表示以消除这些伪影。除了这个新操作，他们利用图像生成扩散模型的最新进展，在训练视图覆盖较少的区域提供监督，并应用类似于先前工作的深度约束。由此，在初始点少于 20 个的情况下，他们仍然通过 3D 高斯泼溅的方法重建了较高质量的图像。

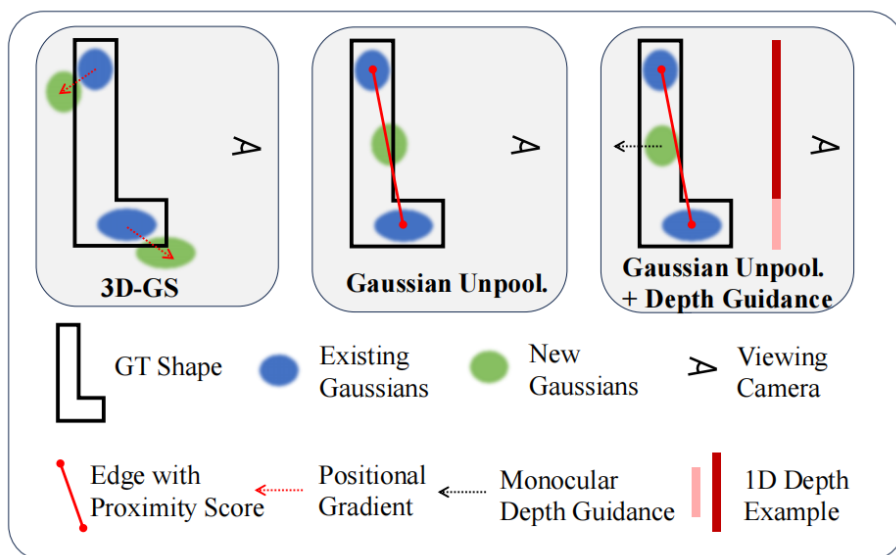


图 4: Zhu 等人的思路 [13]，是在重建过程中加速高斯函数的分裂，并赋予新的高斯点更强的合理性。

Zhu 等人也在稀疏点云的重建工作上有所建树 [13]，他们通过将新的高斯函数放置在代表性的位置并使用现有 3D 高斯的信息进行初始化，有效地增加了高斯个数，以全面覆盖场景。然后，他们使用经过预训练的单ocular深度估计器的丰富深度先验，它指导高斯展开收敛到一个合理的解，并确保场景的几何平滑性 [图 4]。由此，在初始点较少（来自 3 张图片）的情况

下，3D 高斯泼溅技术的效果也达到了高质量的图像和超过 200FPS 的渲染速度，为实时渲染和更具成本效益的捕捉方法提供了新的途径。

5.1 局限性

在目前的研究中，3D 高斯泼溅的场景重建仍然十分依赖于 COLMAP 提供的初始点云，如果点云的分布不准确或缺乏细节，则难以获得重建的结果，特别是在远离画面中心的位置。

6 在自动驾驶领域的应用

6.1 目前进展

3D 高斯泼溅技术在自动驾驶领域的论文较少，现有的 2 篇论文 [11][12] 对于图像重建使用了相似的架构，即分解了静态的街景和动态车辆的 3D 高斯表示。这样的表示有利于优化动态车辆的姿态，从而改善重建结果，实现实时编辑。

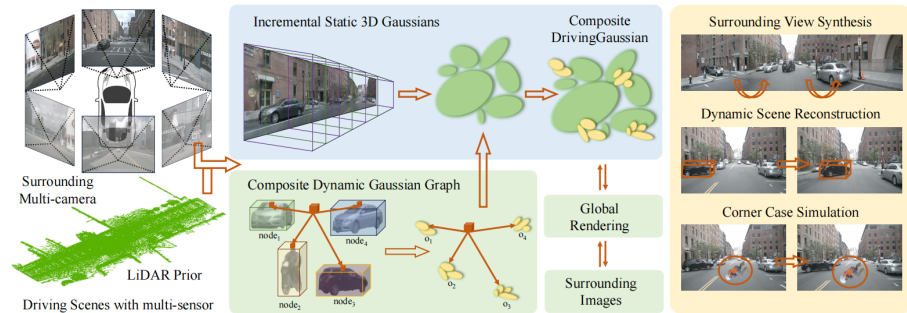


图 5: Zhou 等人的框架 [12]，将高斯图像分为静态和动态，采用不同的渲染方法

作为第一个自动驾驶的 3D 高斯泼溅模型 [12]，Zhou 等人成功完成了静态街景和动态车辆的分别构建。根据他们的实验结果，在初始化阶段使用 LiDAR 点云数据，比 sfM 的结果更好。以雷达为传感器的 LiDAR 点云提供了关于场景中物体位置和几何形状的精确信息，这有助于更准确地初始化高斯场，从而改善重建结果的几何结构。

此外，该文章在 4.5 节也进行了特例模拟（图 5 的右下角），即插入一个摔倒的行人图像作为动态对象，该方法能够成功识别行人。而且，使用不同传感器提供的图像（角度不同），其结果具有一致性。这表明了该方法实现了对自动驾驶场景的可控模拟和编辑，有利于促进安全自动驾驶系统研究。

6.2 局限与挑战

Yan 等人的实验结果 [11] 显示，3D 高斯泼溅技术目前在动态过程的应用上无法良好识别非刚性的物体（如行人），这尚待更加复杂的建模。此外，对于动态车辆的识别完全依赖于车辆追踪器。倘若追踪器的召回率（正确识别车辆的几率）较低，那么在后续的重建中就没有方法纳入被遗漏的车辆。

7 在 SLAM 领域的应用

7.1 目前情况

3D 高斯泼溅在场景表示方面，比其他现有方法具有更快速和更逼真的渲染质量，已经被 SLAM 领域的研究者关注并应用 [1]。目前，3D 高斯泼溅表示主要用于离线系统，专用于已知相机姿态合成新视图，属于视觉 SLAM 和雷达 SLAM 的混合范畴。其中，雷达手段所确认的关键点可以作为 3D 高斯泼溅初始化的稀疏点云；然后，相机获得的不同视角的照片可以让 3D 场景被快速地重建。

另外一个 3D 高斯泼溅的优势在于，可以动态管理添加或删除 3D 高斯函数来高精度地重构观察到的场景几何，同时提高地图精度。GS-SLAM[10] 是目前最新的一个工作，其关键创新在于其自适应扩展策略，即对 3D 高斯函数的动态管理。然而，尽管具有强大之处，GS-SLAM 在大规模场景中对高质量深度信息的依赖和大量内存使用方面仍然存在限制。

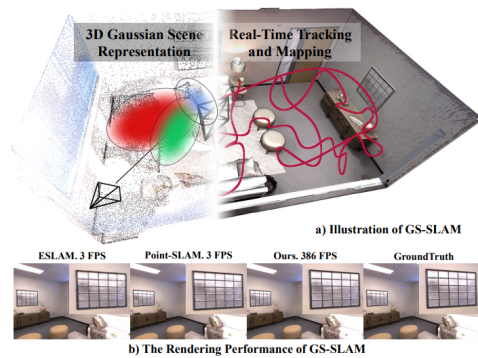


图 6: 以 GS-SLAM 为代表的示意图 [10]

7.2 局限性与挑战

3D 高斯泼溅相比于先前的 NeRF[4] 系列方法，受益于显式的体积显示，具有快速训练、快速渲染、充分优化、更大的地图容量和显式的空间范围控制等优势 [1]。然而，当前 3D 高斯泼溅的 SLAM 应用存在一些已知的限制，包括大场景的可伸缩性问题、缺乏直接网格提取算法和对未观察区域的高斯数量无法控制。

未来，3D 高斯泼溅方法必须关注动态场景的表示，目前的一些时序表示方法来自自动驾驶领域 [11]，使用 4 元的球谐系数替代原先的 3 维球谐系数，或许可以有所启发。对于动态场景重建和跟踪过程中的伪影，研究者还有很大的改进空间。

8 讨论

3D 高斯泼溅是神经辐射场技术的一个巨大革新，在短期的将来会保持较高的热度和发展速度。他的优势在于显式和简单的模型表示、快速的训练和渲染速度、高质量的图像等等。由于不涉及神经网络，3D 高斯泼溅在深度学习火热的当下也能引发更大的兴趣。以下将分别从 3D 高斯泼溅的各个方面，分别来讨论 3D 高斯泼溅的优势、局限性和发展前景。

8.1 Compression (压缩)

3D 高斯泼溅技术的一个关键缺点是比神经网络方法需要更多内存和存储使用，因为除了点的位置，每个高斯函数包含了较多的参数。目前，在建模结果的存储上已经获得了良好的压缩结果 [5]，能够将一个场景的表示压缩在 100MB 以内。对于更大的场景，比如街景，可以研究一种内存上可行的表示方式，看看能否用数千万甚至上亿个 3D 高斯函数来表示更庞大的场景，目前的相关论文还没有将 3D 高斯函数的高效存储压缩应用于各个领域。

8.2 Sparse (有限图像)

在有限视角下的场景重建中，增加 3D 高斯函数的分裂频率以及更加合理的点云初始化能够提高重建图像的质量和速度。目前，3D 高斯泼溅的重建算法不断得到改进 [3][9]，能够用更少的信息、更快的速度重建，仍有一

定的优化空间。在这方面，充分利用 LIDAR 获取的信息来初始化稀疏点云，已经被证明是一个有效的方法 [12]。

8.3 自动驾驶

目前，3D 高斯泼溅技术在自动驾驶的场景重建方面已经产生了优越的成果 [11][12]，静态街景和动态车辆都可以在有竞争力的时间内高质量地呈现出来。然而，3D 高斯泼溅技术在动态对象识别方面还有空缺，他自身无法在重建阶段区分车辆、行人等动态物体，而是依赖于追踪设备的识别。目前，研究者使用已有的不同方法来识别车辆，仍具有发展空间。

8.4 其他领域

除了上文已经讨论的方面，3D 高斯泼溅在众多计算机视觉和计算机图形学的领域也发生了重要革新。譬如人体建模和渲染，已经能够逼真地重建人体和服装；扩散模型也被应用于 3D 高斯泼溅技术的改善；而关于动态的 3D 高斯泼溅技术，也有较多的文章，取得了一些成果。未来，3D 高斯泼溅将有更多应用和发展。

9 结论

综合来看，3D 高斯泼溅技术在图像重建、压缩、稀疏点云初始化、自动驾驶和 SLAM 等领域都取得了显著的进展。其优势在于显式且简单的模型表示、快速的训练和渲染速度、高质量的图像生成以及相对较低的内存需求。该技术仍面临一些挑战和局限性，但是随着研究的深入和技术的不断改进，今年该技术将在图形学、计算机视觉和自动驾驶等领域取得更多的应用和进展。

参考文献

- [1] Guikun Chen and Wenguan Wang. A survey on 3d gaussian splatting. *arXiv preprint arXiv:2401.03890*, 2024.
- [2] Michael Goesele, Noah Snavely, Brian Curless, Hugues Hoppe, and Steven M Seitz. Multi-view stereo for community photo collections. In

- 2007 IEEE 11th International Conference on Computer Vision, pages 1–8. IEEE, 2007.
- [3] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023.
 - [4] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.
 - [5] Simon Niedermayr, Josef Stumpfegger, and Rüdiger Westermann. Compressed 3d gaussian splatting for accelerated novel view synthesis. *arXiv preprint arXiv:2401.02436*, 2023.
 - [6] Aras Pranckevičius. Making gaussian splats smaller, 2023. Accessed on: 2024-02-28.
 - [7] Aras Pranckevičius. Making gaussian splats smaller, 2023. Accessed on: 2024-02-28.
 - [8] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006.
 - [9] Haolin Xiong, Sairisheek Muttukuru, Rishi Upadhyay, Pradyumna Chari, and Achuta Kadambi. SparseSegs: Real-time 360 $\{\backslash\deg\}$ sparse view synthesis using gaussian splatting. *arXiv preprint arXiv:2312.00206*, 2023.
 - [10] Chi Yan, Delin Qu, Dong Wang, Dan Xu, Zhigang Wang, Bin Zhao, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. *arXiv preprint arXiv:2311.11700*, 2023.
 - [11] Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang, Xiaowei Zhou, and Sida Peng. Street gaussians for modeling dynamic urban scenes. *arXiv preprint arXiv:2401.01339*, 2024.

- [12] Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. *arXiv preprint arXiv:2312.07920*, 2023.
- [13] Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. Fsgs: Real-time few-shot view synthesis using gaussian splatting. *arXiv preprint arXiv:2312.00451*, 2023.