A Survey on 3D Gaussian Splatting

Guikun Chen, and Wenguan Wang, *Senior Member, IEEE*

Abstract— 3D高斯飞溅（GS）最近在显式辐射场和计算机图形领域中崭露头角，成为一种具有变革性的技术。这种创新方法以利用数百万可学习的3D高斯函数为特征，与主流的神经辐射场方法有着显著的区别，后者主要使用隐式的基于坐标的模型将空间坐标映射到像素值。3D GS具有明确的场景表示和可微分的渲染算法，不仅具有实时渲染能力，还引入了前所未有的可编辑性水平。这使得3D GS成为下一代3D重建和表示的潜在变革者。

在本文中，我们首次系统地总结了3D GS领域最新发展和重要贡献。我们首先对3D GS的基本原理和出现背后的推动力进行了详细探讨，为理解其重要性奠定了基础。

我们讨论的一个重点是3D GS的实际适用性。通过实现前所未有的渲染速度，3D GS为从虚拟现实到交互式媒体等各种应用提供了广泛的可能性。这得到了对领先的3D GS模型进行比较分析的支持，这些模型在各种基准任务中进行评估，以突出它们的性能和实用性。

调查最后确定了当前的挑战，并提出了未来研究在这一领域的潜在方向。通过这项调查，我们旨在为新手和资深研究人员提供宝贵的资源，促进适用和显式辐射场表示的进一步探索和进步。

Index Terms—3D Gaussian Splatting, Explicit Radiance Field, Real-time Rendering, Scene Understanding

1 Introduction

图像为基础的3D场景重建的目标是将捕获场景的一系列视图或视频转换为计算机可以处理和理解的3D模型。这个困难而长期存在的问题对于机器来理解现实世界环境的复杂性至关重要，有助于广泛的应用，如3D建模和动画、机器人导航、历史保护、增强/虚拟现实和自动驾驶。

3D场景重建的历程始于深度学习激增之前，早期的努力集中在光场和基本的场景重建方法上 [1]–[3]。然而，这些早期尝试受到了对密集采样和结构化捕获的依赖的限制，导致在处理复杂场景和光照条件方面存在重大挑战。结构运动估计的出现 [4]以及后续的多视图立体匹配 [5]算法的进步为3D场景重建提供了更加稳健的框架。尽管取得了这些进步，这些方法在新视角合成方面仍然存在困难，并且缺乏与深度场景理解模型的兼容性。NeRF代表了这一进步的巨大飞跃。通过利用深度神经网络，NeRF实现了将空间坐标直接映射到颜色和密度。NeRF的成功在于它能够创建连续的体积场景函数，产生具有前所未有的细节和真实感的结果。然而，像任何新兴技术一样，这种实现也是有代价的：**i)** 计算密集度。基于NeRF的方法计算密集度很高 [6]–[11]，通常需要大量的训练时间和渲染资源，特别是对于高分辨率的输出。**ii)** 可编辑性。对隐式表示的场景进行操作可能具有挑战性，因为直接修改神经网络的权重与场景的几何或外观属性的变化并不直观相关。

在这种背景下，3D高斯飞溅（GS） [12]不仅仅是一个渐进的改进，而是一种重新定义了场景表示和渲染边界的范式转变的方法。虽然NeRF在创建逼真图像方面表现出色，但对于越来越明显的对延迟敏感的应用，对更快速、更高效的渲染方法的需求也日益迫切。3D GS通过引入一种先进的显式场景表示来满足这一需求，在空间中使用数百万个可学习的3D高斯函数对场景进行建模。与隐式、基于坐标的模型 [13]–[15]不同，3D GS采用显式表示和高度并行化的工作流程，促进了更高效的计算和渲染。3D GS的创新在于它独特地结合了可微分管道和基于点的渲染技术的优势 [16]–[21]。通过用可学习的3D高斯函数表示场景，它保留了连续体积辐射场的强大拟合能力，这对于高质量图像合成至关重要，同时避免了与基于NeRF的方法相关的计算开销（例如，计算密集型的射线行进和在空白空间中不必要的计算）。

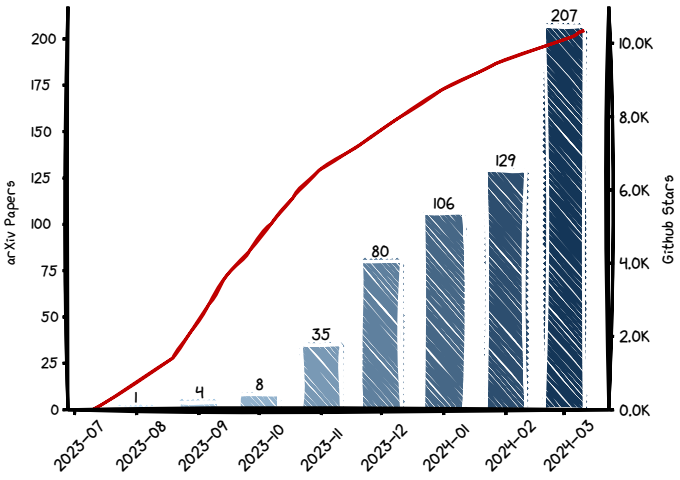

Fig. 1. The number of follow-up arXiv papers and official github stars on 3D GS is increasing every month.

3D GS的引入不仅是技术上的进步；它代表了在计算机视觉和图形领域中我们如何处理场景表示和渲染的根本转变。通过在不降低视觉质量的情况下实现实时渲染能力，3D GS为从虚拟现实和增强现实到实时电影渲染等各种应用打开了大量的可能性 [22]–[25]。这项技术有望不仅增强现有应用，而且使以前由于计算约束而不可行的新应用成为可能。此外，3D GS的显式场景表示提供了前所未有的灵活性，可以控制物体和场景动态，这在涉及复杂几何形状和不同光照条件的复杂场景中是一个关键因素 [26], [27]。这种可编辑性水平，再加上训练和渲染过程的高效性，使3D GS成为塑造相关领域未来发展的变革性力量。

为了帮助读者跟上3D高斯飞溅（GS）迅速发展的步伐，我们提供了第一份关于3D GS的调查报告，这份报告系统地收集了最近关于这一主题的最重要的文献。鉴于3D GS是一种非常新的创新（见图 1），我们的重点是自其引入以来涌现的各种发展和贡献。选定的3D GS后续工作主要来源于顶

- *G. Chen and W. Wang are with College of Computer Science and Technology, Zhejiang University (Email: guikunchen@gmail.com, wenguanwang.ai@gmail.com)*
- *Corresponding Author: Wenguan Wang*

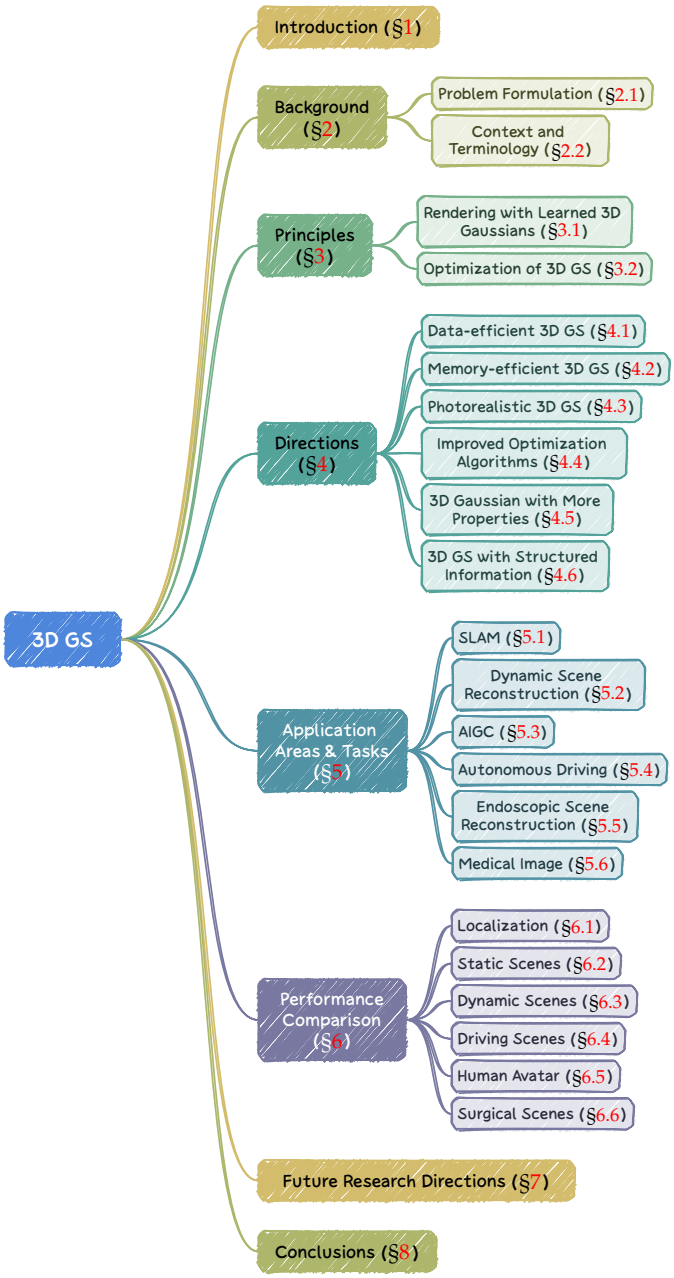

Fig. 2. Structure of the overall review.

级会议。我们的主要目标是全面且及时地分析3D GS的理论基础、显著发展和新兴应用，突出其改变领域的潜力。鉴于3D GS尚处于初创但快速发展的阶段，本调查不可避免地是一种有偏见的观点，但我们努力指出并讨论了有影响力的工作的基本特征，以概括主要的研究趋势。

本文的结构概要如图 2 所示，具体如下：第 2节介绍了问题的背景、术语和相关研究领域。第 3节介绍了3D GS的基本见解，包括用3D高斯函数进行新视角合成和3D GS的优化细微之处。第 4节介绍了几个旨在改进原始3D GS能力的有益方向。第 5节揭示了3D GS产生重大影响的各种应用领域和任务，展示了其多功能性。第 6节进行了性能比较和分析。最后，第 7节和 8节强调了进一步研究的开放问题，并总结了调查。我们旨在提供一个平衡的观点，反映当前的范围和未来的潜力，从而为渴望了解和为这一快速发展领域做出贡献的研究人员和实践者提供宝贵的资源。

2 Background

在本节中，我们首先简要介绍辐射场（Sec. 2.1），这是场景渲染中的一个关键概念。它概述了辐射场表示的两种主要类型：隐式表示，如NeRF [15]，它使用神经网络进行直观但计算量较大的渲染；以及显式表示，如grid [28]，它使用离散结构以更快的速度访问数据，但以更高的内存使用为代价。Sec. 2.2进一步建立了与相关领域（如场景重建和渲染）的联系。关于辐射场、场景重建和表示以及渲染的全面概述，请参阅优秀的调查。 [29]–[34].

2.1 Problem Formulation

2.1.1 辐射场

辐射场是对三维空间中光分布的表示，捕捉了光与环境中的表面和材料相互作用的方式 [30]。在数学上，辐射场可以描述为一个函数$L : \mathbb{R}^5 \mapsto \mathbb{R}^+$，其中$L(x, y, z, \theta, \phi)$ 将空间中的一个点(x, y, z) 和由球坐标(θ, ϕ) 指定的方向映射到一个非负辐射值。辐射场可以通过隐式或显式表示进行封装，每种表示对于场景表示和渲染具有特定的优势。

2.1.2 隐式辐射场

隐式辐射场表示场景中的光分布，而无需明确定义场景的几何形状。在深度学习时代，它通常使用神经网络来学习连续的体积场景表示 [35], [36]。其中最突出的例子是NeRF [15]。在NeRF中，通常使用多层感知器（MLP）的神经网络将一组空间坐标(x, y, z) 和观察方向(θ, ϕ) 映射到颜色和密度值。任意点的辐射并没有显式存储，而是通过查询MLP进行即时计算。因此，该函数可以写成：

$$L_{\text{implicit}}(x, y, z, \theta, \phi) = \text{MLP}(x, y, z, \theta, \phi). \quad (1)$$

这种格式允许对复杂场景进行可微分和紧凑的表示，尽管通常在渲染过程中会由于体积光线行进而导致高计算负载 [12]。

2.1.3 显式辐射场

相比之下，显式辐射场直接表示光在离散空间结构中的分布，例如体素网格或一组点 [28], [37]。该结构中的每个元素都存储了其在空间中相应位置的辐射信息。这种方法允许更直接、通常更快速地访问辐射数据，但以更高的内存使用和潜在的较低分辨率为代价。显式辐射场表示的通用形式可以写成：

$$L_{\text{explicit}}(x, y, z, \theta, \phi) = \text{DataStructure}[(x, y, z)] \cdot f(\theta, \phi), \quad (2)$$

其中DataStructure 可以是体积、点云等格式，$f(\theta, \phi)$ 是一个根据观察方向修改辐射的函数。

2.1.4 3D高斯光斑：两者兼得

3D GS [12]是具有隐式辐射场优势的显式辐射场。具体而言，它通过利用可学习的3D高斯函数作为灵活高效的表示来结合两种范例的优势。这些高斯函数在多视图图像的监督下进行优化，以准确地表示场景。这种基于3D高斯函数的可微分流水线结合了基于神经网络的优化和显式、结构化数据存储的优势。这种混合方法旨在实现实时、高质量的渲染，并且需要更少的训练时间，特别是对于复杂场景和高分辨率输出。3D高斯表示形式如下：

$$L_{\text{3DGS}}(x, y, z, \theta, \phi) = \sum_i G(x, y, z, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \cdot c_i(\theta, \phi), \quad (3)$$

其中G 是具有均值$\boldsymbol{\mu}_i$ 和协方差$\boldsymbol{\Sigma}_i$ 的高斯函数，c 表示与视角相关的颜色。

2.2 背景和术语

几种技术和研究领域与3D高斯光斑密切相关，以下将简要介绍这些领域。

2.2.1 场景重建和渲染

大致而言，场景重建涉及从一系列图像或其他数据中创建场景的3D模型。渲染是一个更具体的术语，专注于将计算机可读信息（例如场景中的3D对象）转换为基于像素的图像。早期技术基于光场生成了逼真的图像 [1]–[3]。结构来自运动 [4]和多视角立体 [5]算法通过从图像序列估计3D结构进一步推动了这一领域的发展。这些历史方法为更复杂的场景重建和渲染技术提供了坚实的基础 [38]–[41]。

2.2.2 神经渲染和辐射场

神经渲染将深度学习与传统图形技术相结合，创建逼真的图像 [42], [43]。早期尝试利用卷积神经网络来估计混合权重 [40]或纹理空间解决方案 [44], [45]。如第 2.1.1节所述，辐射场表示描述光穿过空间中每一点的每个方向的光量的函数。NeRF [10], [11], [15]使用神经网络，通常是MLP，来建模辐射场，实现了详细而逼真的场景渲染。

2.2.3 体积表示和射线步进

体积表示模型不仅将对象和场景表示为表面，还将其表示为充满材料或空白空间的体积 [46]。这种方法允许更准确地渲染雾、烟雾或半透明材料等现象。射线步进是与体积表示一起使用的一种技术，通过逐步追踪光线通过体积的路径来渲染图像 [13], [14]。NeRF [15]与体积射线步进具有相同的精神，并引入了重要性采样和位置编码来提高合成图像的质量。尽管提供了高质量的结果，但体积射线步进的计算成本很高，因此需要更高效的方法，如3D高斯光斑。

2.2.4 基于点的渲染

基于点的渲染是一种使用点而不是传统多边形来可视化3D场景的技术。这种方法对于渲染复杂、非结构化或稀疏几何数据特别有效。点可以与额外的属性结合使用，如可学习的神经描述符 [47], [48]，并且可以高效地渲染 [49], [50]，但是这种方法存在渲染中的空洞或锯齿等问题。3D高斯光斑 [12]通过使用各向异性高斯函数来更连续、连贯地表示场景，扩展了这个概念。有关更多详细信息，将在第 3节进一步讨论。

3 3D Gaussian Splatting: Principles

3D GS在实时、高分辨率图像渲染方面取得了突破，而无需依赖神经组件。本节旨在提供3D GS的重要见解。首先，我们将详细介绍3D GS在给定良好构建的3D高斯函数的情况下如何合成图像，即第 3.1节中的3D GS的前向过程。然后，我们介绍了如何为给定场景获得良好构建的3D高斯函数，即第 3.2节中的3D GS的优化过程。

3.1 Novel-view Synthesis with Learned 3D Gaussians

考虑一个由（数百万个）优化的3D高斯函数表示的场景。目标是根据指定的摄像机姿态生成图像。回想一下，NeRF通过计算密集的体积光线跟踪来处理这个任务，对每个像素采样3D空间点。这种范式在高分辨率图像合成方面存在困难，无法实现实时渲染，尤其是对于计算资源有限的平台 [12]。相比之下，3D GS首先通过投影这些3D高斯函数到基于像素的图像平面来开始，这个过程被称为“喷洒”（见图 3a）。然后，3D GS对这些高斯函数进行排序，并计算每个像素的值。如图 4所示，NeRF和3D GS的渲染可以视为彼此的逆过程。接下来，我们从3D高斯函数的定义开始，它是3D GS中场景表示的最小元素。接下来，我们描述了这些3D高斯函数如何用于可微渲染。最后，我们介绍了3D GS中使用的加速技术，这是实现快速渲染的关键。

• **Properties of 3D Gaussian.** 一个3D高斯函数由其中心（位置）$\boldsymbol{\mu}$，不透明度α，3D协方差矩阵$\boldsymbol{\Sigma}$和颜色c所特征化。对于视角相关的外观，c由球谐表示。所有这些属性都是可学习的，并通过反向传播进行优化。

• **Frustum Culling.** 在给定的相机姿态下，此步骤确定哪些3D高斯函数在相机的视锥体之外。通过这样做，位于给定视图之外的3D高斯函数将不参与后续的计算，从而节省计算资源。

• **Splatting.** 在这一步中，3D高斯函数（椭球体）被投影到二维图像空间（椭圆）进行渲染。给定视图变换$\boldsymbol{W}$ 和3D协方差矩阵$\boldsymbol{\Sigma}$，通过以下公式计算投影后的二维协方差矩阵$\boldsymbol{\Sigma}'$：

$$\boldsymbol{\Sigma}' = \boldsymbol{J}\boldsymbol{W}\boldsymbol{\Sigma}\boldsymbol{W}^\top\boldsymbol{J}^\top, \tag{4}$$

其中，$\boldsymbol{J}$ 是投影变换的仿射逼近的雅可比矩阵。 [12], [51].

• **Rendering by Pixels.** 在深入讨论最终版本的3D GS之前，该版本利用了几种技术来提高并行计算，我们首先详细阐述其较简单的形式，以了解其基本工作机制。给定像素$\boldsymbol{x}$ 的位置，通过视角变换$\boldsymbol{W}$ 可以计算它与所有重叠高斯函数的距离，即这些高斯函数的深度，形成一个排序后的高斯函数列表$\mathcal{N}$。然后，采用alpha 合成来计算该像素的最终颜色：

$$C = \sum_{i\in\mathcal{N}} c_i\alpha_i' \prod_{j=1}^{i-1}\left(1-\alpha_j'\right), \tag{5}$$

这里的c_i 是学习到的颜色。最终的不透明度α_i' 是学习到的不透明度α_i 与高斯函数的乘积，定义如下：

$$\alpha_i' = \alpha_i \times \exp\Big(-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i')^\top\boldsymbol{\Sigma}_i'^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i')\Big), \tag{6}$$

其中$\boldsymbol{x}'$ 和$\boldsymbol{\mu}_i'$ 是投影空间中的坐标。有一个合理的担忧是，所描述的渲染过程可能比NeRFs 慢，因为生成所需的排序列表很难并行化。确实，这种担忧是合理的；当采用这种简单的逐像素方法时，渲染速度可能会受到显著影响。为了实现实时渲染，3D GS 进行了一些让步以适应并行计算。

• **Tiles (Patches).** 为了避免为每个像素计算高斯函数的成本，3D GS 将精度从像素级别转移到块级别细节。具体地，3D GS 最初将图像划分为多个不重叠的块，原始论文中称为“瓦片” [12]。图3b提供了瓦片的示意图。每个瓦片由16×16像素组成，如 [12]中所建议的。3D GS 进一步确定哪些瓦片与这些投影高斯相交。由于一个投影高斯可能覆盖几个瓦片，一个逻辑的方法涉及复制高斯，为每个副本分配一个标识符（即瓦片ID）以指示相关的瓦片。

• **Parallel Rendering.** 在复制之后，3D GS 将每个高斯的瓦片ID 与从视图转换中获得的深度值相结合。这样，就得到了一个未排序的字节列表，其中高位表示瓦片ID，低位表示深度。通过这样做，可以直接利用排序列表进行渲染（即alpha 合成）。图3c和图3d展示了这些概念的视觉演示。值得强调的是，渲染每个瓦片和像素都是独立进行的，这使得这个过程非常适合并行计算。另一个好处是，每个瓦片的像素可以访问一个公共的共享内存并保持统一的读取顺序，从而实现了alpha 合成的并行执行，提高了效率。在原始论文的官方实现中 [12]，该框架将瓦片和像素的处理视为CUDA 编程架构中的块和线程的类比。

简而言之，3D GS 在前向处理阶段引入了几个近似方法，以提高计算效率，同时保持高水平的图像合成质量。

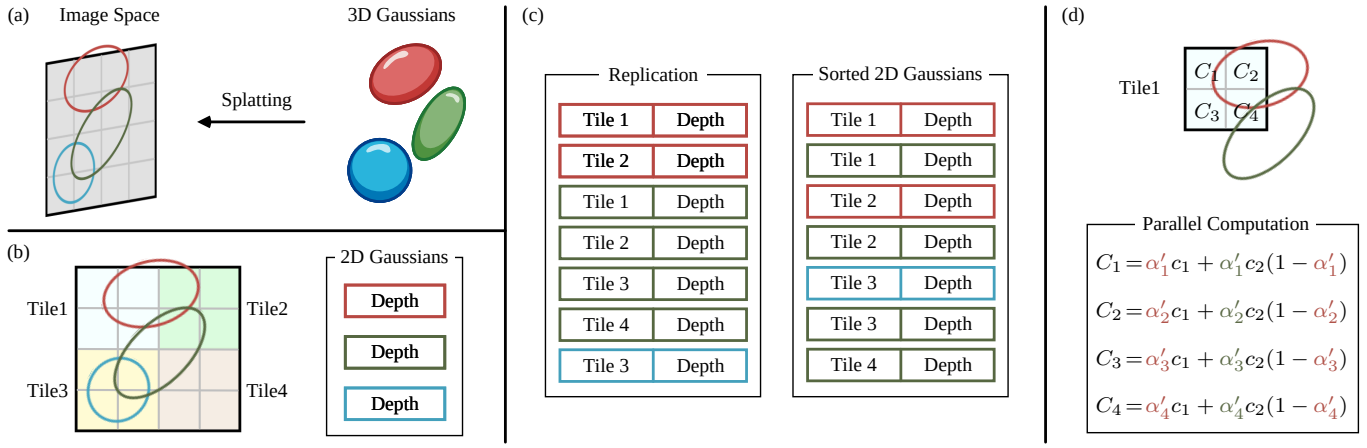

Fig. 3. An illustration of the forward process of 3D GS (Sec. 3.1). (a) The splatting step projects 3D Gaussians into image space. (b) 3D GS divides the image into multiple non-overlapping patches, *i.e.*, tiles. (c) 3D GS replicates the Gaussians which cover several tiles, assigning each copy an identifier, *i.e.*, a tile ID. (d) By rendering the sorted Gaussians, we can obtain all pixels within the tile. Note that the computational workflows for pixels and tiles are independent and can be done in parallel. Best viewed in color.

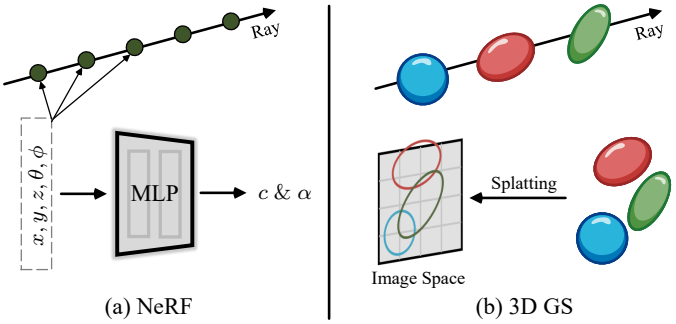

Fig. 4. NeRFs *vs.* 3D GS. (a) NeRF samples along the ray and then queries the MLP to obtain corresponding colors and opacities, which can be seen as a *backward* mapping (ray tracing). (b) In contrast, 3D GS projects all 3D Gaussians into the image space (*i.e.*, splatting) and then performs parallel rendering, which can be viewed as a *forward* mapping (splatting and rasterization). Best viewed in color.

3.2 Optimization of 3D Gaussian Splatting

在3D GS的核心是一种优化过程，旨在构建大量准确捕获场景本质的3D高斯，从而促进自由视点渲染。一方面，通过可微渲染优化3D高斯的属性以适应给定场景的纹理。另一方面，能够有效表示给定场景的3D高斯的数量是未知的。一种有希望的途径是让神经网络自动学习3D高斯的密度。我们将介绍如何优化每个高斯的属性在第 3.2.1节，并且如何在第 3.2.2节中控制高斯的密度。这两个过程在优化工作流中交织在一起。由于在优化过程中有许多手动设置的超参数，为了清晰起见，我们省略了大多数超参数的符号。

3.2.1 Parameter Optimization

• **Loss Function.** 一旦图像的合成完成，就可以测量渲染图像与真实值之间的差异。通过使用L1和D-SSIM损失函数，所有可学习的参数都通过随机梯度下降进行优化：

$$\mathcal{L} = (1 - \lambda)\mathcal{L}_1 + \lambda \mathcal{L}_{\text{D-SSIM}}, \tag{7}$$

其中λ是一个加权因子。3D GS的损失函数与NeRFs的略有不同。由于射线行进成本高昂，NeRFs通常在像素级别而不是图像级别计算损失。

• **Parameter Update.** 大多数3D高斯的属性都可以通过反向传播直接进行优化。需要注意的是，直接优化协方差矩阵$\boldsymbol{\Sigma}$可能会导致非正半定矩阵，这不符合通常与协方差矩阵相关的物理解释。为了避免这个问题，3D GS选择优化四元数$\boldsymbol{q}$和3D向量$\boldsymbol{s}$。$\boldsymbol{q}$和$\boldsymbol{s}$分别表示旋转和缩放。这种方法允许重构协方差矩阵$\boldsymbol{\Sigma}$如下：

$$\boldsymbol{\Sigma} = \boldsymbol{R}\boldsymbol{S}\boldsymbol{S}^\top \boldsymbol{R}^\top, \tag{8}$$

其中$\boldsymbol{R}$和$\boldsymbol{S}$分别表示从$\boldsymbol{q}$和$\boldsymbol{s}$派生的旋转和缩放矩阵。为了避免自动微分的成本，3D GS推导出了$\boldsymbol{q}$和$\boldsymbol{s}$的梯度，以便在优化过程中直接计算它们。

3.2.2 Density Control

• **Initialization.** 3D GS 从SfM 或随机初始化的稀疏点集开始。然后，采用点密集化和修剪来控制3D 高斯的密度。

• **Point Densification.** 在点密集化阶段，3D GS 自适应地增加高斯的密度，以更好地捕获场景的细节。这个过程特别关注缺失几何特征的区域或高斯过于分散的区域。密集化过程将在固定间隔（即，经过一定数量的训练迭代后）执行，重点关注那些具有较大视图空间位置梯度（即，超过特定阈值）的高斯。它包括在重建不足的区域克隆小的高斯或在过度重建的区域分割大的高斯。对于克隆，创建高斯的副本并将其移向位置梯度。对于分割，用两个较小的高斯替换一个大高斯，减小它们的尺度因子。此步骤旨在寻求高斯在3D 空间中的最佳分布和表示，增强重建的整体质量。

• **Point Pruning.** 点修剪阶段涉及消除多余或影响较小的高斯，从某种程度上可以视为正则化过程。该步骤通过消除几乎透明（α 小于指定阈值）以及在世界空间或视图空间中过大的高斯来执行。此外，为了防止在输入相机附近不合理增加高斯密度，高斯的alpha 值在一定数量的迭代后设为接近零。这样做可以控制必要高斯的密度增加，并消除多余的高斯。该过程不仅有助于节省计算资源，还确保模型中的高斯保持精确和有效地表示场景。

4 3D Gaussian Splatting: Directions

过去几个月，3D GS 领域取得了显著进展。我们非常高兴见证了我们在前几版中指出的研究方向的成果。在接下来的几节中，我们将详细阐述3D GS 的一些扩展版本。它们包括：**i)** 数据高效的3D GS [52]–[61] (第4.1 节)，**ii)** 内存高效的3D GS [62]–[68] (第4.2 节)，**iii)** 照片级别逼真的3D GS [69]–[85] (第4.3 节)，**iv)** 改进的优化算法 [82], [86]–[90]

(第4.4 节)，**v)** 具有更多属性的3D 高斯模型 [91]–[97] (第4.5 节)，以及**vi)** 具有结构信息的3D GS [98]–[100] (第4.6 节)。

4.1 Data-efficient 3D GS

3D GS 的一个显著问题是在观测数据不足的区域出现伪影。这一挑战是辐射场渲染中的普遍限制，其中稀疏数据通常导致重建不准确。从实际角度来看，从有限视角重建场景具有重要意义，特别是对于通过最少的输入来增强逼真度和用户体验的潜力。

数据高效的3D GS 主要采用两种策略。**i)** 第一种策略是引入额外的约束，例如深度信息，以增强细节和全局一致性。例如，DNGaussian [56] 提出了一种深度正则化方法，通过硬约束、软约束深度正则化和全局-局部深度归一化来解决稀疏输入视图中几何形状退化的挑战。它利用从预训练的单目深度估计器中提取的深度信息来校正几何形状，并优化高斯原始体的位置而不改变其形状，确保了详细的颜色外观和连贯的场景几何形状之间的平衡。**ii)** 第二种策略是通过使用深度神经网络直接生成可用于渲染的3D 高斯模型，而无需优化。这种范式通常需要多个视图进行训练，但可以仅使用一个输入图像重建3D 场景。例如，PixelSplat [54] 提出从密集概率分布中对高斯模型进行采样。它结合了多视图极线变换和重新参数化技巧，以避免局部最小值并保持梯度流。Splatter Image [55] 通过基于学习的方法在单目环境中应用GS，利用一个2D 图像到图像的网络将输入图像映射到每个像素的3D 高斯模型。Gamba [61] 引入了一个端到端的、基于Mamba 的连续网络 [101] 来生成具有线性可伸缩性的3D 高斯模型。需要注意的是，这种范式主要专注于对象的重建，其通用性还有很大的提升空间。

4.2 Memory-efficient 3D GS

虽然3D GS 展示了显著的能力，但其可伸缩性带来了重大挑战，特别是与NeRF-based 方法相比。后者受益于仅存储学习的MLP 参数的简单性。在大规模场景管理的背景下，这种可伸缩性问题变得越来越严重，计算和内存需求大幅增加。因此，迫切需要优化内存利用率，无论是在训练阶段还是在模型存储期间。

降低内存使用的两个主要方向如下。**i)** 第一种方法是减少3D 高斯模型的数量，即修剪不重要的3D 高斯模型。例如，LightGaussian [62] 根据它们对场景重建的贡献来修剪不重要的高斯模型。Lee 等人 [64] 引入了一种新的基于体积的遮罩策略，有效减少了场景表示所需的高斯模型数量，而不影响性能。**ii)** 第二类方法专注于压缩3D 高斯模型属性的内存使用。例如，LightGaussian [62] 引入了矢量量化 [102] 以实现较低的位宽表示。Lee 等人 [64] 使用基于网格的神经场来更紧凑地表示视角相关的颜色，并利用码本方法简洁地编码高斯几何结构。Niedermayr 等人 [67] 将颜色和高斯参数压缩成紧凑的码本，使用灵敏度度量进行有效量化和微调，结合熵编码以利用空间一致性。

4.3 Photorealistic 3D GS

目前的3D GS 渲染流程(第3.1 节) 是直接的，但存在一些缺点。例如，简单的可见性算法可能导致高斯模型的深度/混合顺序发生剧烈变化。渲染图像的真实性（例如，锯齿状、反射和伪影）可以进一步优化。

以下是提高真实性的几个关键点。**i)** 不同分辨率。由于离散采样范式（将每个像素视为单个点而不是一个区域），3D GS 在处理不同分辨率时容易出现锯齿状或模糊边缘的混叠现象。Yan 等人 [71] 认为，这主要是因为传统的渲染方法无法有效管理像素的采样频率和场景高频细节之间的差异，导致视觉伪影和性能问题。因此，他们引入了多尺度3D GS，其中场景使用不同大小的高斯模型表示。Analytic-Splatting [83] 采用了高斯积分在像素区域内的解析近似，利用条件逻辑函数作为累积分布函数，以更好地捕获像素的强度响应。它旨在使渲染过程对不同分辨率下像素足迹的变化敏感，从而增强抗锯齿能力并保留高质量的细节。**ii)** 反射。实现具有反射材料的真实渲染是3D 场景重建中一个困难且长期存在的问题。GaussianShader [72] 通过将简化的着色函数与3D 高斯模型集成，增强了具有反射表面的场景的神经渲染。**iii)** 几何。3D GS 的一个局限性是忽略了潜在的场景几何结构，特别是在复杂场景和不同的视角和光照条件下。Scaffold-GS [73] 引入了一个稀疏的锚点网格来组织局部的3D 高斯模型，这些模型根据观察者的视角和距离动态调整属性，如不透明度和颜色。该方法通过一个由场景几何形状指导的分层结构增强了场景表示。

4.4 Improved Optimization Algorithms

各向异性高斯模型虽然有助于表示复杂的几何形状，但可能会产生不良的视觉伪影。例如，那些大型的3D 高斯模型，特别是在具有视角相关外观的区域，可能会引起爆裂伪影，即视觉元素突然出现或消失，打破了沉浸感。此外，引入额外的正则化（如几何正则化 [82], [88] 和频率正则化 [89]）以及改进3D GS 的优化过程(第3.2 节) 可能会加快收敛速度，平滑视觉噪音，并提高渲染图像的质量。

改进3D GS 优化的主要方向有三个。**i)** 引入额外的正则化。3D GS 往往面临过度重建的挑战，即稀疏的大型3D 高斯模型在高变异区域的表示不佳，导致模糊和伪影。为了解决这个问题，FreGS [89] 提出了一种渐进式频率正则化方法，从频率的角度优化高斯模型的密度。在全局几何一致性方面，GeoGaussian [82] 着重于保持非纹理区域（如墙壁和家具）的几何形状，这些区域往往会随着时间而退化。**ii)** 改进优化算法。为了解决3D GS 中规模和结构不均匀性带来的挑战，导致无纹理区域边界模糊和稀疏点云，Feng 等人 [90] 引入了一种先进的分割算法。该算法将N 维高斯模型分解为两个，旨在保持关键的数学特性（如不透明度、位置和协方差），增强表面的一致性和均匀性。**iii)** 在优化过程中放宽约束。依赖外部工具/算法可能会引入错误，并限制系统的性能潜力。例如，在初始化过程中常用的运动结构（SfM）对于复杂场景容易产生误差。Yang 等人 [86] 提出了COLMAP-Free 3D GS，引入视频流连续性和显式点云表示，从而消除了对SfM 预处理的需求。

4.5 3D Gaussian with More Properties

尽管3D 高斯模型的属性（第3.1 节）仅设计用于新视角合成，但通过增加额外的属性，如语言 [91]–[93]、语义/实例 [94]–[96] 和时空 [97] 属性，3D GS 展示了在各个领域中彻底改变的巨大潜力。

以下列举了几个使用具有特殊设计属性的3D 高斯模型的有趣应用。**i)** 语言嵌入式场景表示。由于当前语言嵌入式场景表示的高计算和内存需求，Shi 等人 [91] 提出了一种量化方案，使用简化的语言嵌入代替原始的高维嵌入来增强3D 高斯模型。该方法通过在不同视角之间平滑语义特征，并由不确定性值引导，减轻了语义歧义，增强了开放词汇查询的精度。**ii)** 场景理解和编辑。Feature 3DGS [94] 将3D GS 与来自2D 基础模型的特征场蒸馏相结合。通过学习一个较低维度的特征场，并应用轻量级卷积解码器进行上采样，Feature 3DGS 实现了更快的训练和渲染速度，同时实现了高质量的特征场蒸馏，支持语义分割和语言引导编辑等应用。**iii)** 时空建模。为了捕获3D 场景的复杂空间和时间动态，Yang 等人 [97] 将时空看作一个统一的实体，并使用一组4D 高斯模

型近似动态场景的时空体积。所提出的4D 高斯模型表示和相应的渲染流程能够建模空间和时间中的任意旋转，并允许端到端的训练。

4.6 3D GS with Structured Information

除了使用额外属性增强3D 高斯模型外，另一种有前景的方法是引入结构化信息（例如，空间MLP 和网格），以适应特定应用。

接下来，我们展示了使用特别设计的结构化信息的3D 高斯模型的各种迷人用途。**i)** 面部表情建模。考虑到在稀疏视图条件下创建高保真度的3D 头像角色的挑战，Gaussian Head Avatar [100] 引入了可控的3D 高斯模型和基于MLP 的变形场。具体而言，通过优化中性的3D 高斯模型和变形场，它捕捉了详细的面部表情和动态，从而确保了细节的保真度和表情的准确性。**ii)** 时空建模。Yang 等人 [98] 提出使用可变形的3D 高斯模型重建动态场景。这些可变形的3D 高斯模型是在一个标准空间中学习的，配合一个变形场（即空间MLP），用于建模时空动态。所提出的方法还结合了一个退火平滑训练机制，以增强时序平滑性，而不增加额外的计算成本。**iii)** 风格转移。Saroha 等人 [103] 提出了GS in style，这是一种实时神经场景风格化的先进方法。为了在多视图之间保持一致的风格化外观，同时不影响渲染速度，他们使用预训练的3D 高斯模型，结合多分辨率哈希网格和一个小型MLP 来产生风格化的视图。这样的快速、视角一致的管道特别适用于增强现实/虚拟现实应用。

5 Application Areas and Tasks

3D GS的变革潜力远远超出了其理论和计算方面的进展。本节深入探讨了3D GS正在产生重大影响的各种开创性应用领域，如机器人技术（第 5.1 节）、动态场景重建和表示（第 5.2 节）、AI生成内容（第 5.3 节）、自动驾驶（第 5.4 节）、医疗系统（第 5.5 和第 5.6 节），甚至其他科学领域。3D GS的应用展示了它改革各个领域的多功能性。以下我们概述了一些最值得关注的应用领域，以及3D GS如何在每个领域塑造新的前沿。

5.1 Simultaneous Localization and Mapping (SLAM)

SLAM是机器人和自主系统的核心计算问题。它涉及到机器人或设备在未知环境中理解自身位置的挑战，同时绘制环境的布局 [104]。SLAM在各种应用中至关重要，包括自动驾驶汽车、增强现实和机器人导航 [105], [106]。SLAM的核心是实时创建未知环境的地图，并确定设备在该地图上的位置。因此，SLAM对于计算密集型场景表示技术提出了巨大挑战，但同时也是3D GS的一个良好测试平台。

3D GS以创新的方式进入了SLAM领域作为场景表示的方法。传统的SLAM系统通常使用点云/曲面云或体素网格来表示环境 [107]–[114]。相比之下，3D GS利用各向异性高斯模型更好地表示环境。最近的创新研究已经在SLAM中采用了3D高斯模型，展示了这种范式的潜力和多功能性 [115]–[121]。一个直观的方法是直接将3D高斯模型作为密集地图的基本表示，并优化跟踪过程。例如，GS-SLAM [118]采用自适应策略添加或移除3D高斯模型，以优化场景几何重建，并改善先前观察到的区域的地图绘制。它还通过选择可靠的3D高斯模型实现了高效的相机姿态跟踪的粗到细的方案。SplaTAM [119]集成了一种简单的在线跟踪和绘图方法，利用轮廓遮罩来捕获场景密度的存在，促进快速渲染、密集优化和结构化地图扩展。GSSLAM [116]直接优化3D高斯模型，用于稳健的相机跟踪，并引入了几何验证和正则化来解决重建的歧义性。另一方面，设计先进的场景表示也是值得探索的。例如，Photo-SLAM [115]提出了一个超级原语地图，将显式的几何特征与隐式的光度特征结合起来，用于精确的定位和纹理映射。

5.2 Dynamic Scene Reconstruction

动态场景重建是指捕捉和表示随时间变化的场景的三维结构和外观的过程 [122]–[125]。这涉及到创建一个数字模型，准确反映了场景中物体的几何形状、运动和视觉特征随着时间的演变。动态场景重建在各种应用中至关重要，例如虚拟和增强现实、3D动画和计算机视觉。

为了将3D高斯模型的概念扩展到动态场景中，一个直接的方法是加入时间维度，从而允许对随时间变化的场景进行表示和渲染。基于3D高斯模型的方法 [97]–[99], [126]–[136]用于动态场景重建通常可以分为两个主要类别。第一类利用额外的结构信息，如空间MLP或网格（第 4.6 节）。例如，Yang等人 [98]首次提出了针对动态场景量身定制的可变形3D高斯模型。这些3D高斯模型是在一个标准空间中学习的，并且可以使用隐式变形场（实现为MLP）来模拟时空变形。SC-GS [131]利用稀疏的控制点和变形MLP来捕捉和表示3D场景的动态性。通过将运动分解为应用于这些控制点的紧凑、可学习的变换，SC-GS实现了高效、连贯的运动建模，并允许运动编辑。另一方面，第二类基于这样一个思想，即场景变化可以通过特别设计的渲染过程编码到3D高斯模型表示中（第 4.5 节）。例如，Luiten等人 [130]引入了动态3D高斯模型来模拟动态场景，通过保持3D高斯模型的特性（即颜色、不透明度和大小）随时间保持不变，同时允许它们的位置和方向发生变化。Yang等人 [97]设计了一个4D高斯模型表示，其中使用附加属性来表示4D旋转和球谐函数，以近似场景的时空体积。

5.3 AI-Generated Content (AIGC)

AIGC指的是由人工智能系统自主创建或显著改变的数字内容，特别是在计算机视觉、自然语言处理和机器学习领域。AIGC的特点在于它能够模拟、扩展或增强人类生成的内容，从而实现从逼真图像合成到动态叙事创作等应用。AIGC的重要性在于其在娱乐、教育和技术发展等各个领域的变革潜力 [137]–[140]。在数字内容创建不断发展的背景下，AIGC是一个至关重要的元素，为传统方法提供了可扩展、可定制且通常更高效的替代方案。

3D高斯模型的显式特性使其具有实时渲染能力和前所未有的控制和可编辑性，因此在AIGC应用中具有高度相关性。3D高斯模型的显式场景表示和可微分渲染算法与AIGC对生成高保真度、实时和可编辑内容的需求完美契合，这对虚拟现实、交互媒体等应用至关重要。最近的研究已经有效地将3D高斯模型与生成模型 [141]–[159]、头像 [160]–[185]和场景编辑 [94]–[96], [131]–[133], [186]–[192]等领域相结合。例如，DreamGaussian [142]通过三个步骤加速了从单视图图像生成逼真的3D资产的过程：基于扩散的生成高斯过程，然后根据局部密度查询从3D高斯模型中提取网格，最后通过UV空间细化阶段来提高纹理细节。通过将3D高斯模型与参数化可变形脸部模型结合，GaussianAvatars [169]提供了头像动画中的增强保真度和灵活性，显著改进了当前方法在新视图渲染和表情再现方面的性能。GaussianEditor [186]引入了高斯语义追踪以进行精确编辑，采用了分层高斯模型以在随机生成指导下实现稳定结果，同时还采用了针对性编辑策略以实现高效的对象移除和集成。

TABLE 1
Quantitative localization results (§6.1) on Replica [193], in terms of absolute trajectory error (ATE, cm). * denotes numbers taken from [117]. (The three best scores are marked in **red**, **blue**, and **green**, respectively. These notes also apply to the other tables.)

Method	GS	Room0	Room1	Room2	Office0	Office1	Office2	Office3	Office4	Avarage
F2F* [194] [ICCV17]		1.64	1.92	2.80	2.48	0.80	4.55	2.64	2.27	2.38
iMAP [195] [ICCV21]		3.12	2.54	2.31	1.69	1.03	3.99	4.05	1.93	2.58
Vox-Fusion [196] [ISMAR22]		1.37	4.70	1.47	8.48	2.04	2.58	1.11	2.94	3.09
NICE-SLAM [197] [CVPR22]		0.97	1.31	1.07	0.88	1.00	1.06	1.10	1.13	1.06
ESLAM [198] [CVPR23]		0.71	0.70	0.52	0.57	0.55	0.58	0.72	0.63	0.63
Point-SLAM [199] [ICCV23]		0.61	0.41	0.37	0.38	0.48	0.54	0.69	0.72	0.52
Co-SLAM [200] [CVPR23]		0.70	0.95	1.35	0.59	0.55	2.03	1.56	0.72	1.00
Gaussian-SLAM [117] [arXiv]	✓	3.35	8.74	3.13	1.11	0.81	0.78	1.08	7.21	3.27
GSSLAM [116] [CVPR24]	✓	0.47	0.43	0.31	0.70	0.57	0.31	0.31	3.20	0.79
SplaTAM [119] [CVPR24]	✓	0.31	0.40	0.29	0.47	0.27	0.29	0.32	0.55	0.36
GS-SLAM [118] [CVPR24]	✓	0.48	0.53	0.33	0.52	0.41	0.59	0.46	0.70	0.50

5.4 Autonomous Driving

自动驾驶旨在实现车辆在没有人类干预的情况下导航和操作。这些车辆配备了一套传感器，包括摄像头、光探测与测距（LiDAR）和雷达，结合先进的算法、机器学习模型和大量的计算能力 [201]–[204]。其核心目标是感知环境、做出明智的决策，并安全高效地执行机动操作 [205]–[208]。

自动驾驶车辆需要感知和解释其周围环境以安全导航。这涉及实时重建行驶场景，准确识别静态和动态物体，并理解它们的空间关系和运动状态 [209]–[211]。在动态驾驶场景中，由于其他车辆、行人或动物等移动对象，环境不断变化 [212]。实时准确地重建这些场景对于安全导航至关重要，但由于涉及元素的复杂性和变化性，这是一项具有挑战性的任务。在自动驾驶中，可以利用3D高斯模型通过将数据点（如来自LiDAR等传感器的数据）融合成连贯且连续的表示来重建场景。这在处理数据点的密度变化以及确保对场景中的静态背景和动态物体进行平滑准确的重建方面特别有用。为了从稀疏传感器数据中重建复杂的3D场景，尤其是在高速行驶和有移动对象的情况下，主流框架将城市/街景分为静态和动态元素，其中动态元素使用复合动态高斯图 [213]、点云结合语义logit [214]或受物理约束的模型 [215]进行建模。

5.5 Endoscopic Scene Reconstruction

外科3D重建代表了机器人辅助微创手术领域的一项基本任务，旨在通过对动态外科场景进行精确建模，增强术中导航、术前规划和教育模拟。最近的进展致力于将尖端的动态辐射场技术整合到这一领域，重点解决了单视点视频重建所固有的挑战，例如外科器械的遮挡和内窥镜探查的狭窄空间内稀疏视角多样性导致的问题。尽管取得了进步，对组织可变形性和拓扑变异性的高保真度要求仍然存在，同时对更快的渲染速度的需求也日益迫切，以满足实时应用的需求。从内窥镜视频中重建可变形组织，即即时性和精度的合成在推动机器人手术朝着减少患者创伤和增强现实应用方面至关重要，最终促进更直观的外科环境并培养外科自动化和机器人熟练度的未来。

与典型的动态场景重建相比，内窥镜场景重建面临着独特的挑战，例如，由于受限的摄像机移动而导致的稀疏训练数据、由工具遮挡导致的未观察区域以及组织明显的非刚性变形。现有方法主要使用额外的深度引导来推断组织的几何形状。例如，Endo-4DGS [216]引入了一个时间组件到3D高斯模型中，利用轻量级的MLP来模拟时间变形，从而便于重建动态外科场景。通过利用现成的深度估计模型从单眼视图生成伪深度图，它克服了立体摄像机设置的局限性，改善了深度引导的重建过程。类似地，EndoGaussian [217]引入了两种新策略：整体高斯初始化，利用深度估计模型从双目/单目图像序列预测深度图，通过像素重投影和组合便于通过像素重投影和组合来初始化密集高斯；以及时空高斯跟踪，利用形变场模拟场景动态，包括高效编码体素和轻量级形变解码器，从而实现了有效的高斯跟踪，降低了训练和渲染的工作量。EndoGS [218]还将深度引导监督与时空权重掩模和表面对齐正则化项相结合，以提高3D组织渲染的质量和速度，同时解决工具遮挡的问题。通过利用简化的高斯表示结合可微分栅格化，EndoGSLAM [219]提供了精确的相机跟踪、详细的3D组织重建和实时可视化，以支持外科医生进行微创手术过程中的操作。

5.6 Medical Image

在医学影像领域，X射线是一种基础技术，利用电磁辐射来捕捉人体内部结构的图像。这种方法依赖于X射线被各种组织的不同吸收程度，使骨骼、器官和其他内部元素之间的对比可见，而无需外科手术干预。计算机断层扫描（CT）是X射线成像的一种高级形式，通过从患者周围不同角度获取多个X射线图像来增强这一过程。然后利用这些图像来建模三维表示（例如，体积），提供类似于逐层切割身体的详细横截面视图。这种能力极大地增强了诊断的精度，允许有效地识别、监测和管理医学状况。

3D高斯模型可直接用于X射线新视图合成，只需对高斯的属性进行轻微修改，即从球谐函数改为其他函数来表示X射线辐射的吸收。当前的研究可以根据其对扫描对象的构造进行分类：i）无体积。X-Gaussian [220]提出了两个主要创新：重新设计的辐射高斯点云模型，捕捉X射线辐射的各向同性特性，利用辐射强度响应函数代替球谐函数独立地建模辐射强度，而不受视角方向的影响；以及角度姿态立方体均匀初始化策略，用于高效的点云初始化，利用X射线扫描仪参数显著减少训练时间。ii）有体积。为了减少传统CT扫描的辐射剂量，Li等人 [221]使用3D高斯模型进行了稀疏视图CT重建。他们利用由滤波背投影重建的图像的先验信息初始化高斯属性，并采用自适应密度控制（Sec. 3.2.2）来优化投影空间中的所有3D高斯模型。需要注意的是，[221]将3D高斯模型用作体积表示的基础，并不涉及点云投射过程。

6 Performance Comparison

在本节中，我们通过展示我们先前讨论过的几种3D GS算法的性能，提供更多的经验证据。3D GS在众多任务中的多样化应用，以及针对每个任务的定制算法设计，使得在单个任务或数据集上对所有3D GS算法进行统一比较变得不可行。因此，基于我们在第 5节中的分析，我们选择了3D GS领域内的几个代表性任务进行深入的性能评估。性能评分主要来源于原始论文，除非另有说明。

6.1 Performance Benchmarking: Localization

SLAM中的定位任务涉及确定机器人或设备在环境中的精确位置和方向，通常使用传感器数据进行。

• **Dataset:** Replica [193] dataset is a collection of 18 highly detailed 3D indoor scenes. These scenes are not only visually realistic but also offer comprehensive data including dense meshes, high-quality HDR textures, and detailed semantic information for each element. Following [195], three sequences about rooms and five sequences about offices are used for the evaluation.

• **Benchmarking Algorithms:** For performance comparison, we involve four recent papers which introduce 3D Gaussians into their systems [116]–[119], as well as seven typical SLAM methods [194]–[200]

• **Evaluation Metric:** The root mean square error (RMSE) of the absolute trajectory error (ATE) is a commonly used metric in evaluating SLAM systems [222], which measures the root mean square of the Euclidean distances between the estimated and true positions over the entire trajectory of the robot or device.

• **Result:** As shown in Table 1, the recent 3D Gaussians based localization algorithms have a clear advantage over existing NeRF based dense visual SLAM. For example, SplaTAM [119] achieves a trajectory error improvement of over **30**%, decreasing it from 0.52cm to **0.36cm** compared to the previous state-of-the-art (SOTA) [199]. We attribute this to the dense and accurate 3D Gaussians reconstructed for scenes, which can handle the noise of real sensors. This reveals that effective scene representations can improve the accuracy of localization tasks.

6.2 Performance Benchmarking: Rendering Quality in Static Scenes

Rendering focuses on transforming computer-readable information (*e.g.*, 3D objects in the scene) to pixel-based images. This section focuses on evaluating the quality of rendering results in static scenes.

• **Dataset:** The same dataset as in Sec. 6.1, *i.e.*, Replica [193], is used for performance comparison.

• **Benchmarking Algorithms:** For performance comparison, we involve four recent papers which introduce 3D Gaussians into their systems [116]–[119], as well as three dense SLAM methods [196], [197], [199].

• **Evaluation Metric:** Peak signal-to-noise ratio (PSNR), structural similarity (SSIM) [226], and learned perceptual image patch similarity (LPIPS) [227] are used for measuring RGB rendering performance.

• **Result:** Table 2 shows that 3D Gaussians based systems generally outperform the three dense SLAM competitors. For example, Gaussian-SLAM [117] establishes new SOTA and outperforms previous methods by a large margin. Compared to Point-SLAM [199], GSSLAM [116] is about **578** times faster in achieving very competitive accuracy. In contrast to previous method [199] that relies on depth information, such as depth-guided ray sampling, for synthesizing novel views, 3D GS based system [116] eliminates this need, allowing for efficient rendering for any given views with high fidelity.

6.3 Performance Benchmarking: Rendering Quality in Dynamic Scenes

This section focuses on evaluating the rendering quality in dynamic scenes.

• **Dataset:** D-NeRF [122] dataset includes videos with 50 to 200 frames each, captured from unique viewpoints. It features synthetic, animated objects in complex scenes, with non-Lambertian materials. The dataset provides 50 to 200 training images and 20 test images per scene, designed for evaluating models in monocular settings with varied camera poses.

• **Benchmarking Algorithms:** For performance comparison, we involve four recent papers that model dynamic scenes with 3D GS [97], [99], [133], [134], as well as six NeRF based approaches [37], [122], [125], [223]–[225].

• **Evaluation Metric:** The same metrics as in Sec. 6.2, *i.e.*, PSNR, SSIM [226], and LPIPS [227], are used for evaluation.

• **Result:** From Table 3 we can observe that 3D GS based methods outperform existing SOTAs by a clear margin. The static version of 3D GS [12] fails to reconstruct dynamic scenes, resulting in a sharp drop in performance. By modeling the dynamics, CoGS [133] outperforms the SOTA method, FFDNeRF [125], by **5.22**dB in terms of PSNR. These impressive results indicate the effectiveness of introducing additional properties or structured information to model the deformation of 3D Gaussians so as to model the scene dynamics.

6.4 Performance Benchmarking: Rendering Quality in Driving Scenes

This section focuses on evaluating the rendering quality in driving scenes, which is essential for autonomous driving.

• **Dataset:** nuScenes [204] dataset is a comprehensive collection for autonomous driving research, featuring 1000 driving scenes captured using an array of sensors including six cameras, one LiDAR, five RADARs, GPS, and IMU. It provides detailed annotations for 23 object classes with 3D bounding boxes. Six challenging scenes are used for evaluation [213].

• **Benchmarking Algorithms:** For performance comparison, we involve one recent paper which models driving scene with 3D GS [213], as well as six NeRF based approaches [10], [11], [228]–[231].

• **Evaluation Metric:** PSNR, SSIM [226], and LPIPS [227] are used for evaluation.

• **Result:** The results in Table 4 demonstrate that the 3D GS based methods significantly surpass the NeRF based methods across all evaluated metrics. For instance, DrivingGaussian-L [213] outperforms S-NeRF [230] by **3.31**dB in terms of PSNR. This suggests that 3D Gaussians can benefit from multi-sensor information to capture dynamic objects in driving scenes, especially fast-moving dynamic objects.

6.5 Performance Benchmarking: Human Avatar

Human avatar modeling aims to create the model of human avatars from a given multi-view video.

• **Dataset:** ZJU-MoCap [232] dataset is a prevalent benchmark in human modeling from videos, captured with 23

TABLE 2
Quantitative rendering results (§6.2) on Replica [193], in terms of PSNR, SSIM, and LPIPS. The numbers of FPS are taken from [116].

Method	GS	Metric	Room0	Room1	Room2	Office0	Office1	Office2	Office3	Office4	Avarage	FPS
NICE-SLAM [197] [CVPR22]		PSNR↑	22.12	22.47	24.52	29.07	30.34	19.66	22.23	24.94	24.42	0.54
		SSIM↑	0.69	0.76	0.81	0.87	0.89	0.80	0.80	0.86	0.81	
		LPIPS↓	0.33	0.27	0.21	0.23	0.18	0.23	0.21	0.20	0.23	
Vox-Fusion [196] [ISMAR22]		PSNR↑	22.39	22.36	23.92	27.79	29.83	20.33	23.47	25.21	24.41	2.17
		SSIM↑	0.68	0.75	0.80	0.86	0.88	0.79	0.80	0.85	0.80	
		LPIPS↓	0.30	0.27	0.23	0.24	0.18	0.24	0.21	0.20	0.24	
Point-SLAM [199] [ICCV23]		PSNR↑	32.40	34.08	35.50	38.26	39.16	33.99	33.48	33.49	35.17	1.33
		SSIM↑	0.97	0.98	0.98	0.98	0.99	0.96	0.96	0.98	0.97	
		LPIPS↓	0.11	0.12	0.11	0.10	0.12	0.16	0.13	0.14	0.12	
Gaussian-SLAM [117] [arXiv]	✓	PSNR↑	34.31	37.28	38.18	43.97	43.56	37.39	36.48	40.19	38.90	-
		SSIM↑	0.99	0.99	0.99	1.00	0.99	0.99	0.99	1.00	0.99	
		LPIPS↓	0.08	0.07	0.07	0.04	0.07	0.08	0.08	0.07	0.07	
GSSLAM [116] [CVPR24]	✓	PSNR↑	34.83	36.43	37.49	39.95	42.09	36.24	36.70	36.07	37.50	769
		SSIM↑	0.95	0.96	0.96	0.97	0.98	0.96	0.96	0.96	0.96	
		LPIPS↓	0.07	0.08	0.07	0.07	0.06	0.08	0.07	0.10	0.07	
SplaTAM [119] [CVPR24]	✓	PSNR↑	32.86	33.89	35.25	38.26	39.17	31.97	29.70	31.81	34.11	-
		SSIM↑	0.98	0.97	0.98	0.98	0.98	0.97	0.95	0.95	0.97	
		LPIPS↓	0.07	0.10	0.08	0.09	0.09	0.10	0.12	0.15	0.10	
GS-SLAM [118] [CVPR24]	✓	PSNR↑	31.56	32.86	32.59	38.70	41.17	32.36	32.03	32.92	34.27	-
		SSIM↑	0.97	0.97	0.97	0.99	0.99	0.98	0.97	0.97	0.97	
		LPIPS↓	0.09	0.07	0.09	0.05	0.03	0.09	0.11	0.11	0.08	

TABLE 3
Quantitative rendering results (§6.3) on D-NeRF [122], in terms of PSNR, SSIM, and LPIPS. * denotes numbers taken from [99].

Method	GS	PSNR↑	SSIM↑	LPIPS↓
D-NeRF [122] [CVPR21]		30.50	0.95	0.07
TiNeuVox-B [223] [SGA22]		32.67	0.97	0.04
KPlanes [37] [CVPR23]		31.61	0.97	-
HexPlane-Slim [224] [CVPR23]		32.68	0.97	0.02
FFDNeRF [125] [ICCV23]		32.68	0.97	0.02
MSTH [225] [NeurIPS23]		31.34	0.98	0.02
3D GS* [12] [TOG23]	✓	23.19	0.93	0.08
4DGS [97] [ICLR24]	✓	34.09	0.98	-
CoGS [133] [CVPR24]	✓	37.90	0.98	0.02
4D-GS [99] [CVPR24]	✓	34.05	0.98	0.02
D-3DGS [98] [CVPR24]	✓	39.51	0.99	0.01

TABLE 4
Quantitative rendering results (§6.4) on nuScenes [204], in terms of PSNR, SSIM, and LPIPS. * denotes numbers taken from [213].

Method	GS	PSNR↑	SSIM↑	LPIPS↓
Mip-NeRF [228] [ICCV21]		18.08	0.57	0.55
Mip-NeRF 360 [10] [CVPR22]		22.61	0.69	0.40
Instant-NGP [11] [TOG22]		16.78	0.52	0.57
Urban-NeRF [229] [CVPR22]		20.75	0.63	0.48
S-NeRF [230] [ICLR23]		25.43	0.73	0.30
SUDS [231] [CVPR23]		21.26	0.60	0.47
3D GS* [12] [TOG23]	✓	26.08	0.72	0.30
DrivingGaussian-S [213] [CVPR24]	✓	28.36	0.85	0.26
DrivingGaussian-L [213] [CVPR24]	✓	28.74	0.86	0.24

synchronized cameras at a 1024×1024 resolution. Following [236], six subjects (*i.e.*, 377, 386, 387, 392, 393, and 394) are used for evaluation.

• **Benchmarking Algorithms:** For performance comparison, we involve four recent papers which model human avatar with 3D GS [164], [166], [179], [182], as well as six human rendering approaches [232]–[237].

• **Evaluation Metric:** PSNR, SSIM [226], and LPIPS* [227] are used for measuring RGB rendering performance. Here LPIPS* equals to LPIPS × 1000.

TABLE 5
Quantitative avatar modeling results (§6.5) on ZJU-MoCap [232], in terms of PSNR, SSIM, and LPIPS*. The numbers of non-GS methods and FPS are taken from [164] and [182], respectively. † denotes the average of the values reported in the original paper.

Method	GS	PSNR↑	SSIM↑	LPIPS*↓	FPS
NeuralBody [232] [CVPR21]		29.03	0.96	42.47	3.5
AnimNeRF [233] [ICCV21]		29.77	0.96	46.89	2.1
PixelNeRF [234] [ICCV21]		24.71	0.89	121.86	-
NHP [235] [NeurIPS21]		28.25	0.95	64.77	-
HumanNeRF [236] [CVPR22]		30.66	0.97	33.38	0.4
Instant-NVR [237] [CVPR23]		31.01	0.97	38.45	1.5
Human101 [182] [arXiv]	✓	31.79	0.96	35.75	104.0
HUGS† [166] [CVPR24]	✓	30.98	0.97	26.67	-
3DGS-Avatar† [179] [CVPR24]	✓	30.61	0.97	29.58	-
GART [164] [CVPR24]	✓	32.22	0.98	29.21	-

• **Result:** Table 5 presents the numerical results of top-leading solutions in human avatar modeling. We observe that introducing 3D GS into the framework leads to consistent performance improvements in both rendering quality and speed. For instance, GART [164] outperforms current SOTA, Instant-NVR [237], by **1.21**dB in terms of PSNR. Note that Human101 [182] is about **68** times faster in achieving very competitive accuracy compared to Instant-NVR [237]. Considering the enhanced fidelity, inference speed and editability, 3D GS based avatar modeling may revolutionize the field of 3D animation, interactive gaming, *etc*.

6.6 Performance Benchmarking: Surgical 3D Reconstruction

3D reconstruction from endoscopic video is critical to robotic-assisted minimally invasive surgery, enabling pre-operative planning, training through AR/VR simulations, and intraoperative guidance.

• **Dataset:** EndoNeRF [238] dataset presents a specialized collection of stereo camera captures, comprising two samples of in-vivo prostatectomy. It is tailored to represent real-world surgical complexities, including challenging scenes with tool occlusion and pronounced non-rigid deformation.

TABLE 6
Quantitative surgical 3D reconstruction results (§6.6) on EndoNeRF [238], in terms of PSNR, SSIM, and LPIPS. The numbers of non-GS methods and FPS are taken from [217]. † denotes the average of the values reported in the original paper.

Method	GS	PSNR↑	SSIM↑	LPIPS↓	FPS↑	Mem.↓
EndoNeRF [238] [MICCAI22]		36.06	0.93	0.09	0.04	19GB
EndoSurf [239] [MICCAI23]		36.53	0.95	0.07	0.04	17GB
LerPlane-9k [240] [MICCAI23]		35.00	0.93	0.08	0.91	20GB
LerPlane-32k [240] [MICCAI23]		37.38	0.95	0.05	0.87	20GB
Endo-4DGS† [216] [arXiv]	✓	37.00	0.96	0.05	-	4GB
EndoGaussian [217] [arXiv]	✓	37.85	0.96	0.05	195.09	2GB
EndoGS [218] [arXiv]	✓	37.94	0.97	0.03	-	-

• **Benchmarking Algorithms:** For performance comparison, we involve three recent papers which reconstruct dynamic 3D endoscopic scenes with GS [216]–[218], as well as three NeRF-based surgical reconstruction approaches [238]–[240].
• **Evaluation Metric:** PSNR, SSIM [226], and LPIPS [227] are adopted for evaluation. In addition, the requirement for GPU memory is also reported.
• **Result:** Table 6 shows that introducing the explicit representation of 3D Gaussians leads to several significant improvements. For instance, EndoGaussian [217] outperforms a strong baseline, LerPlane-32k [240], among all metrics. In particular, EndoGaussian demonstrates an approximate 224-fold increase in speed while consumes just 10% of the GPU resources. These impressive results attest to the efficiency of GS-based methods, which not only expedite processing but also minimize GPU load, thus easing the demands on hardware. Such attributes are vitally significant for real-world surgical application deployment, where optimized resource usage can be a crucial determinant of practical utility.

7 Future Research Directions

尽管那些对3D GS的后续工作令人印象深刻，并且那些领域可能已经或可能会被3D GS所改革，但普遍认为，3D GS仍然有相当大的改进空间。

• **Physics- and Semantics-aware Scene Representation.** 作为一种新的显式场景表示技术，3D高斯具有超越仅仅增强新视图合成的变革潜力。通过设计物理和语义感知的3D GS系统，它有望为场景重建和理解的同时进步铺平道路。这将彻底改革一系列领域和下游应用。例如，将先验知识（如对象的一般形状）纳入考虑可以减少对大量训练视角的需求 [54], [55]，同时提高几何/表面重建的质量 [82], [241]。评估场景表示的一个关键指标是其生成场景的真实性，这涵盖了几何、纹理和光照保真度方面的挑战 [70], [131], [186]。通过在3D GS框架内融合物理原理和语义信息，可以期待真实性得到增强，从而促进动态建模 [25], [242]、编辑 [94], [96]、生成 [142], [143]等方面的发展。简而言之，追求这种先进而多才多艺的场景表示为计算创造力和跨领域实际应用带来了新的创新可能性。

• **3D GS for Robotics.** 在机器人领域，特别是涉及类似人类处理对象的任务时，越来越需要这些机器能够以更直观、更动态的方式导航和操纵环境。这种需求源于在现实世界环境中部署智能机器人的愿望，这些机器人通常需要面对新的、陌生的任务。传统的机器人操纵方法主要依赖于通过语义表示理解环境，即识别对象及其属性。然而，这些方法经常忽视了随时间物体如何移动和相互作用的重要性，而这对于以人类意图完成任务至关重要。想象一下，一个机器人试图根据口头命令堆叠木块。仅使用语义表示，机器人可能会识别出木块，但如果它不理解木块如何随时间相对位置和移动，就无法将它们堆叠起来。这种理解上的差距可能导致执行需要与多个对象交互或导航动态环境的任务失败。由于其显式的特性，3D GS不仅可用于环境的语义和结构分析，还涵盖了动态方面，提供了对场景如何随时间演变以及物体如何相互作用的全面理解。这对于负责导航和操纵周围环境的机器人至关重要。尽管使用3D GS构建世界模型的初步努力 [243], [244]令人鼓舞，但这只是可能性的开始。预计在这一领域的进一步研究将增强机器人在执行既需要理解物理空间又需要理解其中时间变化的任务时的能力。

• **Modeling Internal Structures of Objects with 3D GS.** 尽管3D高斯具有生成高度逼真渲染的能力，但在当前的高斯场景表示框架中对物体的内部结构建模（例如，对于计算机断层扫描中的扫描物体）仍然存在显着挑战。由于分散和密度控制过程，当前3D高斯的表示是无组织的，不能很好地与物体的实际内部结构对齐。此外，各种应用程序对将对象描述为体积（例如，计算机断层扫描）有很强的偏好。然而，3D高斯的无序特性使得体积建模特别困难。Li等人 [221]使用带有密度控制的3D高斯作为体积表示的基础，并没有涉及分散过程。X-Gaussian [220]使用分散过程进行快速训练和推断，但无法生成体积表示。使用3D高斯来对物体的内部结构建模仍然没有答案，值得进一步探索。

• **3D Large-scale Scene Reconstruction.** 3D场景重建和渲染，特别是针对大型环境，因其在诸如增强现实（AR）、虚拟现实（VR）和自动驾驶等各个领域的应用而引起了相当大的关注。直接使用3D高斯进行大规模场景重建是不可行的，原因如下：**i)** 在当前框架内重建大规模场景对硬件来说是不可行的；和**ii)** 渲染如此多的高斯会引入大量的计算负载。作为早期尝试，CityGS [245]提出了一种分治训练策略以及一种逐层细节渲染方法，通过将大场景分割成较小的可管理的块，每个块由较少数量的高斯表示。总的来说，使用3D高斯重建大规模场景仍然具有显著的待探索潜力。

• **3D GS for Simulation in Autonomous Driving.** 为自动驾驶收集真实世界数据集既昂贵又具有后勤挑战，但对于训练有效的基于图像的感知系统却至关重要。为了缓解这些问题，仿真成为了一种经济有效的替代方案，可以在各种环境中生成合成数据集。然而，开发能够产生逼真和多样化合成数据的模拟器充满了挑战。其中包括实现高水平的真实感、适应各种控制方法，以及准确模拟各种光照条件。虽然早期尝试 [213]–[215]在使用3D高斯重建城市/街道场景方面是令人鼓舞的，但这只是其全部能力的冰山一角。仍然有许多关键方面需要探索，例如用户定义的对象模型的集成、模拟具有物理感知的场景变化（例如，车轮的旋转），以及增强可控性和真实性（例如，在不同的光照条件下）。

• **Empowering 3D GS with More Possibilities.** 尽管3D高斯具有显著的潜力，但其应用的完整范围仍然大部分未被开发。一个有前途的探索方向涉及将3D高斯与额外属性（如在第4.5节中提到的语言和时空属性）相结合，并引入结构化信息（例如，如第4.6节中提到的空间MLP和网格），以针对特定应用进行定制。此外，最近的研究已经开始揭示3D高斯在几个领域的能力，例如，相机姿态估计 [246]、手对象交互的捕捉 [247]、不确定性的量化 [248]、手术场景重建 [216]–[218]和流体合成 [249]。这些初步发现突显了跨学科学者进一步探索3D高斯的重要机会。

8 Conclusions

To the best of our knowledge, this survey presents the first comprehensive overview of 3D GS, a groundbreaking technique revolutionizing explicit radiance fields, computer

graphics, and computer vision. It delineates the paradigm shift from traditional NeRF based methods, spotlighting the advantages of 3D GS in real-time rendering and enhanced editability. Our detailed analysis demonstrates the superiority of 3D GS in practical applications, particularly those highly sensitive to latency. We offer insights into principles, prospective research directions, and the unresolved challenges within this domain. Overall, 3D GS stands as a transformative technology, poised to significantly influence future advancements in 3D reconstruction and representation. This survey is intended to serve as a foundational resource, propelling further exploration and progress in this rapidly evolving field.

REFERENCES

[1] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, 2023, pp. 453–464.

[2] M. Levoy and P. Hanrahan, "Light field rendering," in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, 2023, pp. 441–452.

[3] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen, "Unstructured lumigraph rendering," in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, 2023, pp. 497–504.

[4] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: exploring photo collections in 3d," in *ACM TOG*, 2006, pp. 835–846.

[5] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz, "Multi-view stereo for community photo collections," in *ICCV*, 2007, pp. 1–8.

[6] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "Tensorf: Tensorial radiance fields," in *European Conference on Computer Vision*, 2022, pp. 333–350.

[7] S. J. Garbin, M. Kowalski, M. Johnson, J. Shotton, and J. Valentin, "Fastnerf: High-fidelity neural rendering at 200fps," in *ICCV*, 2021, pp. 14 346–14 355.

[8] C. Reiser, S. Peng, Y. Liao, and A. Geiger, "Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps," in *ICCV*, 2021, pp. 14 335–14 345.

[9] T. Takikawa, J. Litalien, K. Yin, K. Kreis, C. Loop, D. Nowrouzezahrai, A. Jacobson, M. McGuire, and S. Fidler, "Neural geometric level of detail: Real-time rendering with implicit 3d shapes," in *CVPR*, 2021, pp. 11 358–11 367.

[10] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-nerf 360: Unbounded anti-aliased neural radiance fields," in *CVPR*, 2022, pp. 5470–5479.

[11] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM TOG*, vol. 41, no. 4, pp. 1–15, 2022.

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

[13] P. Henzler, N. J. Mitra, and T. Ritschel, "Escaping plato's cave: 3d shape from adversarial rendering," in *ICCV*, 2019, pp. 9984–9993.

[14] V. Sitzmann, J. Thies, F. Heide, M. Nießner, G. Wetzstein, and M. Zollhofer, "Deepvoxels: Learning persistent 3d feature embeddings," in *CVPR*, 2019, pp. 2437–2446.

[15] 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," in *ECCV*, 2020, pp. 405–421.

[16] H. Pfister, M. Zwicker, J. Van Baar, and M. Gross, "Surfels: Surface elements as rendering primitives," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000, pp. 335–342.

[17] M. Zwicker, H. Pfister, J. Van Baar, and M. Gross, "Surface splatting," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001, pp. 371–378.

[18] L. Ren, H. Pfister, and M. Zwicker, "Object space ewa surface splatting: A hardware accelerated approach to high quality point rendering," in *Computer Graphics Forum*, no. 3, 2002, pp. 461–470.

[19] M. Botsch, A. Hornung, M. Zwicker, and L. Kobbelt, "High-quality surface splatting on today's gpus," in *Proceedings Eurographics/IEEE VGTC Symposium Point-Based Graphics, 2005.*, 2005, pp. 17–141.

[20] W. Yifan, F. Serena, S. Wu, C. Öztireli, and O. Sorkine-Hornung, "Differentiable surface splatting for point-based geometry processing," *ACM TOG*, vol. 38, no. 6, pp. 1–14, 2019.

[21] O. Wiles, G. Gkioxari, R. Szeliski, and J. Johnson, "Synsin: End-to-end view synthesis from a single image," in *CVPR*, 2020, pp. 7467–7477.

[22] D. Kalkofen, E. Mendez, and D. Schmalstieg, "Comprehensible visualization for augmented reality," *IEEE TVCG*, vol. 15, no. 2, pp. 193–204, 2008.

[23] A. Patney, M. Salvi, J. Kim, A. Kaplanyan, C. Wyman, N. Benty, D. Luebke, and A. Lefohn, "Towards foveated rendering for gaze-tracked virtual reality," *ACM TOG*, vol. 35, no. 6, pp. 1–12, 2016.

[24] R. Albert, A. Patney, D. Luebke, and J. Kim, "Latency requirements for foveated rendering in virtual reality," *ACM Transactions on Applied Perception (TAP)*, vol. 14, no. 4, pp. 1–13, 2017.

[25] Y. Jiang, C. Yu, T. Xie, X. Li, Y. Feng, H. Wang, M. Li, H. Lau, F. Gao, Y. Yang *et al.*, "Vr-gs: A physical dynamics-aware interactive gaussian splatting system in virtual reality," *arXiv preprint arXiv:2401.16663*, 2024.

[26] R. Chabra, J. E. Lenssen, E. Ilg, T. Schmidt, J. Straub, S. Lovegrove, and R. Newcombe, "Deep local shapes: Learning local sdf priors for detailed 3d reconstruction," in *ECCV*, 2020, pp. 608–625.

[27] Z. Wang, J. Philion, S. Fidler, and J. Kautz, "Learning indoor inverse rendering with 3d spatially-varying lighting," in *ICCV*, 2021, pp. 12 538–12 547.

[28] C. Sun, M. Sun, and H.-T. Chen, "Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction," in *CVPR*, 2022, pp. 5459–5469.

[29] L. Kobbelt and M. Botsch, "A survey of point-based techniques in computer graphics," *Computers & Graphics*, vol. 28, no. 6, pp. 801–814, 2004.

[30] Y. Xie, T. Takikawa, S. Saito, O. Litany, S. Yan, N. Khan, F. Tombari, J. Tompkin, V. Sitzmann, and S. Sridhar, "Neural fields in visual computing and beyond," in *Computer Graphics Forum*, no. 2, 2022, pp. 641–676.

[31] A. Tewari, J. Thies, B. Mildenhall, P. Srinivasan, E. Tretschk, W. Yifan, C. Lassner, V. Sitzmann, R. Martin-Brualla, S. Lombardi *et al.*, "Advances in neural rendering," in *Computer Graphics Forum*, no. 2, 2022, pp. 703–735.

[32] X.-F. Han, H. Laga, and M. Bennamoun, "Image-based 3d object reconstruction: State-of-the-art and trends in the deep learning era," *IEEE TPAMI*, vol. 43, no. 5, pp. 1578–1604, 2019.

[33] J. Yuan, T. Chen, B. Li, and X. Xue, "Compositional scene representation learning via reconstruction: A survey," *IEEE TPAMI*, 2023.

[34] W. Wang, Y. Yang, and Y. Pan, "Visual knowledge in the big model era: Retrospect and prospect," *arXiv preprint arXiv:2404.04308*, 2024.

[35] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3d reconstruction in function space," in *CVPR*, 2019, pp. 4460–4470.

[36] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *CVPR*, 2019, pp. 165–174.

[37] S. Fridovich-Keil, G. Meanti, F. R. Warburg, B. Recht, and A. Kanazawa, "K-planes: Explicit radiance fields in space, time, and appearance," in *CVPR*, 2023, pp. 12 479–12 488.

[38] M. Eisemann, B. De Decker, M. Magnor, P. Bekaert, E. De Aguiar, N. Ahmed, C. Theobalt, and A. Sellent, "Floating textures," in *Computer graphics forum*, no. 2, 2008, pp. 409–418.

[39] G. Chaurasia, S. Duchene, O. Sorkine-Hornung, and G. Drettakis, "Depth synthesis and local warps for plausible image-based navigation," *ACM TOG*, vol. 32, no. 3, pp. 1–12, 2013.

[40] P. Hedman, J. Philip, T. Price, J.-M. Frahm, G. Drettakis, and G. Brostow, "Deep blending for free-viewpoint image-based rendering," *ACM TOG*, vol. 37, no. 6, pp. 1–15, 2018.

[41] G. Kopanas, J. Philip, T. Leimkühler, and G. Drettakis, "Point-based neural rendering with per-view optimization," in *Computer Graphics Forum*, no. 4, 2021, pp. 29–43.

[42] J. Flynn, I. Neulander, J. Philbin, and N. Snavely, "Deepstereo: Learning to predict new views from the world's imagery," in *CVPR*, 2016, pp. 5515–5524.

[43] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros, "View synthesis by appearance flow," in *ECCV*, 2016, pp. 286–301.

[44] J. Thies, M. Zollhöfer, and M. Nießner, "Deferred neural rendering: Image synthesis using neural textures," *ACM TOG*, vol. 38, no. 4, pp. 1–12, 2019.

[45] G. Riegler and V. Koltun, "Free view synthesis," in *ECCV*, 2020, pp. 623–640.

[46] E. Penner and L. Zhang, "Soft 3d reconstruction for view synthesis," *ACM TOG*, vol. 36, no. 6, pp. 1–11, 2017.

[47] K.-A. Aliev, A. Sevastopolsky, M. Kolos, D. Ulyanov, and V. Lempitsky, "Neural point-based graphics," in *ECCV*, 2020, pp. 696–712.

[48] D. Rückert, L. Franke, and M. Stamminger, "Adop: Approximate differentiable one-pixel point rendering," *ACM TOG*, vol. 41, no. 4, pp. 1–14, 2022.

[49] S. Laine and T. Karras, "High-performance software rasterization on gpus," in *Proceedings of the ACM SIGGRAPH Symposium on High Performance Graphics*, 2011, pp. 79–88.

[50] M. Schütz, B. Kerbl, and M. Wimmer, "Software rasterization of 2 billion points in real time," *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, vol. 5, no. 3, pp. 1–17, 2022.

[51] M. Zwicker, H. Pfister, J. Van Baar, and M. Gross, "Ewa volume splatting," in *Proceedings Visualization, 2001. VIS'01.*, 2001, pp. 29–538.

[52] H. Xiong, S. Muttukuru, R. Upadhyay, P. Chari, and A. Kadambi, "Sparsegs: Real-time 360 {\deg} sparse view synthesis using gaussian splatting," *arXiv preprint arXiv:2312.00206*, 2023.

[53] Z. Zhu, Z. Fan, Y. Jiang, and Z. Wang, "Fsgs: Real-time few-shot view synthesis using gaussian splatting," *arXiv preprint arXiv:2312.00451*, 2023.

[54] D. Charatan, S. Li, A. Tagliasacchi, and V. Sitzmann, "pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction," *arXiv preprint arXiv:2312.12337*, 2023.

[55] S. Szymanowicz, C. Rupprecht, and A. Vedaldi, "Splatter image: Ultra-fast single-view 3d reconstruction," *arXiv preprint arXiv:2312.13150*, 2023.

[56] J. Li, J. Zhang, X. Bai, J. Zheng, X. Ning, J. Zhou, and L. Gu, "Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization," *arXiv preprint arXiv:2403.06912*, 2024.

[57] A. Swann, M. Strong, W. K. Do, G. S. Camps, M. Schwager, and M. Kennedy III, "Touch-gs: Visual-tactile supervised 3d gaussian splatting," *arXiv preprint arXiv:2403.09875*, 2024.

[58] Y. Chen, H. Xu, C. Zheng, B. Zhuang, M. Pollefeys, A. Geiger, T.-J. Cham, and J. Cai, "Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images," *arXiv preprint arXiv:2403.14627*, 2024.

[59] C. Wewer, K. Raj, E. Ilg, B. Schiele, and J. E. Lenssen, "latentsplat: Autoencoding variational gaussians for fast generalizable 3d reconstruction," *arXiv preprint arXiv:2403.16292*, 2024.

[60] Y. Xu, Z. Shi, W. Yifan, H. Chen, C. Yang, S. Peng, Y. Shen, and G. Wetzstein, "Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation," *arXiv preprint arXiv:2403.14621*, 2024.

[61] Q. Shen, X. Yi, Z. Wu, P. Zhou, H. Zhang, S. Yan, and X. Wang, "Gamba: Marry gaussian splatting with mamba for single view 3d reconstruction," *arXiv preprint arXiv:2403.18795*, 2024.

[62] 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.

[63] K. Navaneet, K. P. Meibodi, S. A. Koohpayegani, and H. Pirsiavash, "Compact3d: Compressing gaussian splat radiance field models with vector quantization," *arXiv preprint arXiv:2311.18159*, 2023.

[64] 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.

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

[66] X. Zhang, X. Ge, T. Xu, D. He, Y. Wang, H. Qin, G. Lu, J. Geng, and J. Zhang, "Gaussianimage: 1000 fps image representation and compression by 2d gaussian splatting," *arXiv preprint arXiv:2403.08551*, 2024.

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

[68] Y. Chen, Q. Wu, J. Cai, M. Harandi, and W. Lin, "Hac: Hash-grid assisted context for 3d gaussian splatting compression," *arXiv preprint arXiv:2403.14530*, 2024.

[69] Z. Yu, A. Chen, B. Huang, T. Sattler, and A. Geiger, "Mip-splatting: Alias-free 3d gaussian splatting," *arXiv preprint arXiv:2311.16493*, 2023.

[70] J. Gao, C. Gu, Y. Lin, H. Zhu, X. Cao, L. Zhang, and Y. Yao, "Relightable 3d gaussian: Real-time point cloud relighting with brdf decomposition and ray tracing," *arXiv preprint arXiv:2311.16043*, 2023.

[71] Z. Yan, W. F. Low, Y. Chen, and G. H. Lee, "Multi-scale 3d gaussian splatting for anti-aliased rendering," *arXiv preprint arXiv:2311.17089*, 2023.

[72] Y. Jiang, J. Tu, Y. Liu, X. Gao, X. Long, W. Wang, and Y. Ma, "Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces," *arXiv preprint arXiv:2311.17977*, 2023.

[73] T. Lu, M. Yu, L. Xu, Y. Xiangli, L. Wang, D. Lin, and B. Dai, "Scaffold-gs: Structured 3d gaussians for view-adaptive rendering," *arXiv preprint arXiv:2312.00109*, 2023.

[74] B. Lee, H. Lee, X. Sun, U. Ali, and E. Park, "Deblurring 3d gaussian splatting," *arXiv preprint arXiv:2401.00834*, 2024.

[75] D. Malarz, W. Smolak, J. Tabor, S. Tadeja, and P. Spurek, "Gaussian splitting algorithm with color and opacity depended on viewing direction," *arXiv preprint arXiv:2312.13729*, 2023.

[76] L. Bolanos, S.-Y. Su, and H. Rhodin, "Gaussian shadow casting for neural characters," *arXiv preprint arXiv:2401.06116*, 2024.

[77] L. Radl, M. Steiner, M. Parger, A. Weinrauch, B. Kerbl, and M. Steinberger, "Stopthepop: Sorted gaussian splatting for view-consistent real-time rendering," *arXiv preprint arXiv:2402.00525*, 2024.

[78] Z. Yang, X. Gao, Y. Sun, Y. Huang, X. Lyu, W. Zhou, S. Jiao, X. Qi, and X. Jin, "Spec-gaussian: Anisotropic view-dependent appearance for 3d gaussian splatting," *arXiv preprint arXiv:2402.15870*, 2024.

[79] C. Peng, Y. Tang, Y. Zhou, N. Wang, X. Liu, D. Li, and R. Chellappa, "Bags: Blur agnostic gaussian splatting through multi-scale kernel modeling," *arXiv preprint arXiv:2403.04926*, 2024.

[80] L. Zhao, P. Wang, and P. Liu, "Bad-gaussians: Bundle adjusted deblur gaussian splatting," *arXiv preprint arXiv:2403.11831*, 2024.

[81] H. Dahmani, M. Bennehar, N. Piasco, L. Roldao, and D. Tsishkou, "Swag: Splatting in the wild images with appearance-conditioned gaussians," *arXiv preprint arXiv:2403.10427*, 2024.

[82] Y. Li, C. Lyu, Y. Di, G. Zhai, G. H. Lee, and F. Tombari, "Geogaussian: Geometry-aware gaussian splatting for scene rendering," *arXiv preprint arXiv:2403.11324*, 2024.

[83] Z. Liang, Q. Zhang, W. Hu, Y. Feng, L. Zhu, and K. Jia, "Analytic-splatting: Anti-aliased 3d gaussian splatting via analytic integration," *arXiv preprint arXiv:2403.11056*, 2024.

[84] O. Seiskari, J. Ylilammi, V. Kaatrasalo, P. Rantalankila, M. Turkulainen, J. Kannala, E. Rahtu, and A. Solin, "Gaussian splatting on the move: Blur and rolling shutter compensation for natural camera motion," *arXiv preprint arXiv:2403.13327*, 2024.

[85] X. Song, J. Zheng, S. Yuan, H.-a. Gao, J. Zhao, X. He, W. Gu, and H. Zhao, "Sa-gs: Scale-adaptive gaussian splatting for training-free anti-aliasing," *arXiv preprint arXiv:2403.19615*, 2024.

[86] Y. Fu, S. Liu, A. Kulkarni, J. Kautz, A. A. Efros, and X. Wang, "Colmap-free 3d gaussian splatting," *arXiv preprint arXiv:2312.07504*, 2023.

[87] J. Jung, J. Han, H. An, J. Kang, S. Park, and S. Kim, "Relaxing accurate initialization constraint for 3d gaussian splatting," *arXiv preprint arXiv:2403.09413*, 2024.

[88] M. Yu, T. Lu, L. Xu, L. Jiang, Y. Xiangli, and B. Dai, "Gsdf: 3dgs meets sdf for improved rendering and reconstruction," *arXiv preprint arXiv:2403.16964*, 2024.

[89] J. Zhang, F. Zhan, M. Xu, S. Lu, and E. Xing, "Fregs: 3d gaussian splatting with progressive frequency regularization," *arXiv preprint arXiv:2403.06908*, 2024.

[90] Q. Feng, G. Cao, H. Chen, T.-J. Mu, R. R. Martin, and S.-M. Hu, "A new split algorithm for 3d gaussian splatting," *arXiv preprint arXiv:2403.09143*, 2024.

[91] J.-C. Shi, M. Wang, H.-B. Duan, and S.-H. Guan, "Language embedded 3d gaussians for open-vocabulary scene understanding," *arXiv preprint arXiv:2311.18482*, 2023.

[92] M. Qin, W. Li, J. Zhou, H. Wang, and H. Pfister, "Langsplat: 3d language gaussian splatting," *arXiv preprint arXiv:2312.16084*, 2023.

[93] X. Zuo, P. Samangouei, Y. Zhou, Y. Di, and M. Li, "Fmgs: Foundation model embedded 3d gaussian splatting for holistic 3d scene understanding," *arXiv preprint arXiv:2401.01970*, 2024.

[94] S. Zhou, H. Chang, S. Jiang, Z. Fan, Z. Zhu, D. Xu, P. Chari, S. You, Z. Wang, and A. Kadambi, "Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields," *arXiv preprint arXiv:2312.03203*, 2023.

[95] M. Ye, M. Danelljan, F. Yu, and L. Ke, "Gaussian grouping: Segment and edit anything in 3d scenes," *arXiv preprint arXiv:2312.00732*, 2023.

[96] J. Cen, J. Fang, C. Yang, L. Xie, X. Zhang, W. Shen, and Q. Tian, "Segment any 3d gaussians," *arXiv preprint arXiv:2312.00860*, 2023.

[97] Z. Yang, H. Yang, Z. Pan, X. Zhu, and L. Zhang, "Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting," in *ICLR*, 2024.

[98] Z. Yang, X. Gao, W. Zhou, S. Jiao, Y. Zhang, and X. Jin, "Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction," *arXiv preprint arXiv:2309.13101*, 2023.

[99] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, and X. Wang, "4d gaussian splatting for real-time dynamic scene rendering," *arXiv preprint arXiv:2310.08528*, 2023.

[100] Y. Xu, B. Chen, Z. Li, H. Zhang, L. Wang, Z. Zheng, and Y. Liu, "Gaussian head avatar: Ultra high-fidelity head avatar via dynamic gaussians," *arXiv preprint arXiv:2312.03029*, 2023.

[101] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv preprint arXiv:2312.00752*, 2023.

[102] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[103] A. Saroha, M. Gladkova, C. Curreli, T. Yenamandra, and D. Cremers, "Gaussian splatting in style," *arXiv preprint arXiv:2403.08498*, 2024.

[104] J. A. Placed, J. Strader, H. Carrillo, N. Atanasov, V. Indelman, L. Carlone, and J. A. Castellanos, "A survey on active simultaneous localization and mapping: State of the art and new frontiers," *IEEE Transactions on Robotics*, 2023.

[105] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part i," *IEEE robotics & automation magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[106] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (slam): Part ii," *IEEE robotics & automation magazine*, vol. 13, no. 3, pp. 108–117, 2006.

[107] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE international symposium on mixed and augmented reality*, 2011, pp. 127–136.

[108] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3d reconstruction at scale using voxel hashing," *ACM TOG*, vol. 32, no. 6, pp. 1–11, 2013.

[109] R. Maier, R. Schaller, and D. Cremers, "Efficient online surface correction for real-time large-scale 3d reconstruction," *arXiv preprint arXiv:1709.03763*, 2017.

[110] O. Kähler, V. Prisacariu, J. Valentin, and D. Murray, "Hierarchical voxel block hashing for efficient integration of depth images," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 192–197, 2015.

[111] F. Ruetz, E. Hernández, M. Pfeiffer, H. Oleynikova, M. Cox, T. Lowe, and P. Borges, "Ovpc mesh: 3d free-space representation for local ground vehicle navigation," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 8648–8654.

[112] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger, "Elasticfusion: Real-time dense slam and light source estimation," *The International Journal of Robotics Research*, vol. 35, no. 14, pp. 1697–1716, 2016.

[113] K. Wang, F. Gao, and S. Shen, "Real-time scalable dense surfel mapping," in *2019 International conference on robotics and automation (ICRA)*, 2019, pp. 6919–6925.

[114] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[115] H. Huang, L. Li, H. Cheng, and S.-K. Yeung, "Photo-slam: Real-time simultaneous localization and photorealistic mapping for monocular, stereo, and rgb-d cameras," *arXiv preprint arXiv:2311.16728*, 2023.

[116] H. Matsuki, R. Murai, P. H. Kelly, and A. J. Davison, "Gaussian splatting slam," *arXiv preprint arXiv:2312.06741*, 2023.

[117] V. Yugay, Y. Li, T. Gevers, and M. R. Oswald, "Gaussian-slam: Photo-realistic dense slam with gaussian splatting," *arXiv preprint arXiv:2312.10070*, 2023.

[118] C. Yan, D. Qu, D. Wang, D. Xu, Z. Wang, B. Zhao, and X. Li, "Gs-slam: Dense visual slam with 3d gaussian splatting," *arXiv preprint arXiv:2311.11700*, 2023.

[119] N. Keetha, J. Karhade, K. M. Jatavallabhula, G. Yang, S. Scherer, D. Ramanan, and J. Luiten, "Splatam: Splat, track & map 3d gaussians for dense rgb-d slam," *arXiv preprint arXiv:2312.02126*, 2023.

[120] S. Hong, J. He, X. Zheng, H. Wang, H. Fang, K. Liu, C. Zheng, and S. Shen, "Liv-gaussmap: Lidar-inertial-visual fusion for real-time 3d radiance field map rendering," *arXiv preprint arXiv:2401.14857*, 2024.

[121] Y. Ji, Y. Liu, G. Xie, B. Ma, and Z. Xie, "Neds-slam: A novel neural explicit dense semantic slam framework using 3d gaussian splatting," *arXiv preprint arXiv:2403.11679*, 2024.

[122] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-nerf: Neural radiance fields for dynamic scenes," in *CVPR*, 2021, pp. 10 318–10 327.

[123] K. Park, U. Sinha, P. Hedman, J. T. Barron, S. Bouaziz, D. B. Goldman, R. Martin-Brualla, and S. M. Seitz, "Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields," *arXiv preprint arXiv:2106.13228*, 2021.

[124] K. Park, U. Sinha, J. T. Barron, S. Bouaziz, D. B. Goldman, S. M. Seitz, and R. Martin-Brualla, "Nerfies: Deformable neural radiance fields," in *ICCV*, 2021, pp. 5865–5874.

[125] X. Guo, J. Sun, Y. Dai, G. Chen, X. Ye, X. Tan, E. Ding, Y. Zhang, and J. Wang, "Forward flow for novel view synthesis of dynamic scenes," in *ICCV*, 2023, pp. 16 022–16 033.

[126] Y. Lin, Z. Dai, S. Zhu, and Y. Yao, "Gaussian-flow: 4d reconstruction with dynamic 3d gaussian particle," *arXiv preprint arXiv:2312.03431*, 2023.

[127] A. Kratimenos, J. Lei, and K. Daniilidis, "Dynmf: Neural motion factorization for real-time dynamic view synthesis with 3d gaussian splatting," *arXiv preprint arXiv:2312.00112*, 2023.

[128] D. Das, C. Wewer, R. Yunus, E. Ilg, and J. E. Lenssen, "Neural parametric gaussians for monocular non-rigid object reconstruction," *arXiv preprint arXiv:2312.01196*, 2023.

[129] R. Shaw, J. Song, A. Moreau, M. Nazarczuk, S. Catley-Chandar, H. Dhamo, and E. Perez-Pellitero, "Swags: Sampling windows adaptively for dynamic 3d gaussian splatting," *arXiv preprint arXiv:2312.13308*, 2023.

[130] J. Luiten, G. Kopanas, B. Leibe, and D. Ramanan, "Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis," *arXiv preprint arXiv:2308.09713*, 2023.

[131] Y.-H. Huang, Y.-T. Sun, Z. Yang, X. Lyu, Y.-P. Cao, and X. Qi, "Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes," *arXiv preprint arXiv:2312.14937*, 2023.

[132] R. Shao, J. Sun, C. Peng, Z. Zheng, B. Zhou, H. Zhang, and Y. Liu, "Control4d: Dynamic portrait editing by learning 4d gan from 2d diffusion-based editor," *arXiv preprint arXiv:2305.20082*, 2023.

[133] H. Yu, J. Julin, Z. Á. Milacski, K. Niinuma, and L. A. Jeni, "Cogs: Controllable gaussian splatting," *arXiv preprint arXiv:2312.05664*, 2023.

[134] Y. Liang, N. Khan, Z. Li, T. Nguyen-Phuoc, D. Lanman, J. Tompkin, and L. Xiao, "Gaufre: Gaussian deformation fields for real-time dynamic novel view synthesis," *arXiv preprint arXiv:2312.11458*, 2023.

[135] K. Katsumata, D. M. Vo, and H. Nakayama, "An efficient 3d gaussian representation for monocular/multi-view dynamic scenes," *arXiv preprint arXiv:2311.12897*, 2023.

[136] Z. Li, Z. Chen, Z. Li, and Y. Xu, "Spacetime gaussian feature splatting for real-time dynamic view synthesis," *arXiv preprint arXiv:2312.16812*, 2023.

[137] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[138] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *NeurIPS*, 2020, pp. 6840–6851.

[139] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *ICCV*, 2023, pp. 3836–3847.

[140] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *CVPR*, 2022, pp. 10 684–10 695.

[141] Z. Chen, F. Wang, and H. Liu, "Text-to-3d using gaussian splatting," *arXiv preprint arXiv:2309.16585*, 2023.

[142] J. Tang, J. Ren, H. Zhou, Z. Liu, and G. Zeng, "Dreamgaussian: Generative gaussian splatting for efficient 3d content creation," in *ICLR*, 2024.

[143] T. Yi, J. Fang, G. Wu, L. Xie, X. Zhang, W. Liu, Q. Tian, and X. Wang, "Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors," *arXiv preprint arXiv:2310.08529*, 2023.

[144] X. Li, H. Wang, and K.-K. Tseng, "Gaussiandiffusion: 3d gaussian splatting for denoising diffusion probabilistic models with structured noise," *arXiv preprint arXiv:2311.11221*, 2023.

[145] Y. Liang, X. Yang, J. Lin, H. Li, X. Xu, and Y. Chen, "Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching," *arXiv preprint arXiv:2311.11284*, 2023.

[146] J. Chung, S. Lee, H. Nam, J. Lee, and K. M. Lee, "Luciddreamer: Domain-free generation of 3d gaussian splatting scenes," *arXiv preprint arXiv:2311.13384*, 2023.

[147] X. Liu, X. Zhan, J. Tang, Y. Shan, G. Zeng, D. Lin, X. Liu, and Z. Liu, "Humangaussian: Text-driven 3d human generation with gaussian splatting," *arXiv preprint arXiv:2311.17061*, 2023.

[148] A. Vilesov, P. Chari, and A. Kadambi, "Cg3d: Compositional generation for text-to-3d via gaussian splatting," *arXiv preprint arXiv:2311.17907*, 2023.

[149] X. Yang, Y. Chen, C. Chen, C. Zhang, Y. Xu, X. Yang, F. Liu, and G. Lin, "Learn to optimize denoising scores for 3d generation: A unified and improved diffusion prior on nerf and 3d gaussian splatting," *arXiv preprint arXiv:2312.04820*, 2023.

[150] Z.-X. Zou, Z. Yu, Y.-C. Guo, Y. Li, D. Liang, Y.-P. Cao, and S.-H. Zhang, "Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers," *arXiv preprint arXiv:2312.09147*, 2023.

[151] H. Ling, S. W. Kim, A. Torralba, S. Fidler, and K. Kreis, "Align your gaussians: Text-to-4d with dynamic 3d gaussians and composed diffusion models," *arXiv preprint arXiv:2312.13763*, 2023.

[152] J. Ren, L. Pan, J. Tang, C. Zhang, A. Cao, G. Zeng, and Z. Liu, "Dreamgaussian4d: Generative 4d gaussian splatting," *arXiv preprint arXiv:2312.17142*, 2023.

[153] Y. Yin, D. Xu, Z. Wang, Y. Zhao, and Y. Wei, "4dgen: Grounded 4d content generation with spatial-temporal consistency," *arXiv preprint arXiv:2312.17225*, 2023.

[154] H. Ouyang, K. Heal, S. Lombardi, and T. Sun, "Text2immersion: Generative immersive scene with 3d gaussians," *arXiv preprint arXiv:2312.09242*, 2023.

[155] J. Zhang, Z. Tang, Y. Pang, X. Cheng, P. Jin, Y. Wei, W. Yu, M. Ning, and L. Yuan, "Repaint123: Fast and high-quality one image to 3d generation with progressive controllable 2d repainting," *arXiv preprint arXiv:2312.13271*, 2023.

[156] Z. Pan, Z. Yang, X. Zhu, and L. Zhang, "Fast dynamic 3d object generation from a single-view video," *arXiv preprint arXiv:2401.08742*, 2024.

[157] D. Xu, Y. Yuan, M. Mardani, S. Liu, J. Song, Z. Wang, and A. Vahdat, "Agg: Amortized generative 3d gaussians for single image to 3d," *arXiv preprint arXiv:2401.04099*, 2024.

[158] J. Tang, Z. Chen, X. Chen, T. Wang, G. Zeng, and Z. Liu, "Lgm: Large multi-view gaussian model for high-resolution 3d content creation," *arXiv preprint arXiv:2402.05054*, 2024.

[159] C. Yang, S. Li, J. Fang, R. Liang, L. Xie, X. Zhang, W. Shen, and Q. Tian, "Gaussianobject: Just taking four images to get a high-quality 3d object with gaussian splatting," *arXiv preprint arXiv:2402.10259*, 2024.

[160] W. Zielonka, T. Bagautdinov, S. Saito, M. Zollhöfer, J. Thies, and J. Romero, "Drivable 3d gaussian avatars," *arXiv preprint arXiv:2311.08581*, 2023.

[161] R. Jena, G. S. Iyer, S. Choudhary, B. Smith, P. Chaudhari, and J. Gee, "Splatarmor: Articulated gaussian splatting for animatable humans from monocular rgb videos," *arXiv preprint arXiv:2311.10812*, 2023.

[162] K. Ye, T. Shao, and K. Zhou, "Animatable 3d gaussians for high-fidelity synthesis of human motions," *arXiv preprint arXiv:2311.13404*, 2023.

[163] Z. Li, Z. Zheng, L. Wang, and Y. Liu, "Animatable gaussians: Learning pose-dependent gaussian maps for high-fidelity human avatar modeling," *arXiv preprint arXiv:2311.16096*, 2023.

[164] J. Lei, Y. Wang, G. Pavlakos, L. Liu, and K. Daniilidis, "Gart: Gaussian articulated template models," *arXiv preprint arXiv:2311.16099*, 2023.

[165] A. Moreau, J. Song, H. Dhamo, R. Shaw, Y. Zhou, and E. Pérez-Pellitero, "Human gaussian splatting: Real-time rendering of animatable avatars," *arXiv preprint arXiv:2311.17113*, 2023.

[166] M. Kocabas, J.-H. R. Chang, J. Gabriel, O. Tuzel, and A. Ranjan, "Hugs: Human gaussian splats," *arXiv preprint arXiv:2311.17910*, 2023.

[167] R. Abdal, W. Yifan, Z. Shi, Y. Xu, R. Po, Z. Kuang, Q. Chen, D.-Y. Yeung, and G. Wetzstein, "Gaussian shell maps for efficient 3d human generation," *arXiv preprint arXiv:2311.17857*, 2023.

[168] J. Wang, X. Li, J. Xie, F. Xu, and H. Gao, "Gaussianhead: Impressive 3d gaussian-based head avatars with dynamic hybrid neural field," *arXiv preprint arXiv:2312.01632*, 2023.

[169] S. Qian, T. Kirschstein, L. Schoneveld, D. Davoli, S. Giebenhain, and M. Nießner, "Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians," *arXiv preprint arXiv:2312.02069*, 2023.

[170] S. Zheng, B. Zhou, R. Shao, B. Liu, S. Zhang, L. Nie, and Y. Liu, "Gps-gaussian: Generalizable pixel-wise 3d gaussian splatting for real-time human novel view synthesis," *arXiv preprint arXiv:2312.02155*, 2023.

[171] S. Hu and Z. Liu, "Gauhuman: Articulated gaussian splatting from monocular human videos," *arXiv preprint arXiv:2312.02973*, 2023.

[172] H. Dhamo, Y. Nie, A. Moreau, J. Song, R. Shaw, Y. Zhou, and E. Pérez-Pellitero, "Headgas: Real-time animatable head avatars via 3d gaussian splatting," *arXiv preprint arXiv:2312.02902*, 2023.

[173] Y. Jiang, Z. Shen, P. Wang, Z. Su, Y. Hong, Y. Zhang, J. Yu, and L. Xu, "Hifi4g: High-fidelity human performance rendering via compact gaussian splatting," *arXiv preprint arXiv:2312.03461*, 2023.

[174] L. Hu, H. Zhang, Y. Zhang, B. Zhou, B. Liu, S. Zhang, and L. Nie, "Gaussianavatar: Towards realistic human avatar modeling from a single video via animatable 3d gaussians," *arXiv preprint arXiv:2312.02134*, 2023.

[175] J. Xiang, X. Gao, Y. Guo, and J. Zhang, "Flashavatar: High-fidelity digital avatar rendering at 300fps," *arXiv preprint arXiv:2312.02214*, 2023.

[176] S. Saito, G. Schwartz, T. Simon, J. Li, and G. Nam, "Relightable gaussian codec avatars," *arXiv preprint arXiv:2312.03704*, 2023.

[177] Y. Chen, L. Wang, Q. Li, H. Xiao, S. Zhang, H. Yao, and Y. Liu, "Monogaussianavatar: Monocular gaussian point-based head avatar," *arXiv preprint arXiv:2312.04558*, 2023.

[178] H. Pang, H. Zhu, A. Kortylewski, C. Theobalt, and M. Habermann, "Ash: Animatable gaussian splats for efficient and photoreal human rendering," *arXiv preprint arXiv:2312.05941*, 2023.

[179] Z. Qian, S. Wang, M. Mihajlovic, A. Geiger, and S. Tang, "3dgs-avatar: Animatable avatars via deformable 3d gaussian splatting," *arXiv preprint arXiv:2312.09228*, 2023.

[180] Y. Yuan, X. Li, Y. Huang, S. De Mello, K. Nagano, J. Kautz, and U. Iqbal, "Gavatar: Animatable 3d gaussian avatars with implicit mesh learning," *arXiv preprint arXiv:2312.11461*, 2023.

[181] H. Jung, N. Brasch, J. Song, E. Perez-Pellitero, Y. Zhou, Z. Li, N. Navab, and B. Busam, "Deformable 3d gaussian splatting for animatable human avatars," *arXiv preprint arXiv:2312.15059*, 2023.

[182] M. Li, J. Tao, Z. Yang, and Y. Yang, "Human101: Training 100+ fps human gaussians in 100s from 1 view," *arXiv preprint arXiv:2312.15258*, 2023.

[183] Z. Zhao, Z. Bao, Q. Li, G. Qiu, and K. Liu, "Psavatar: A point-based morphable shape model for real-time head avatar creation with 3d gaussian splatting," *arXiv preprint arXiv:2401.12900*, 2024.

[184] M. Li, S. Yao, Z. Xie, K. Chen, and Y.-G. Jiang, "Gaussianbody: Clothed human reconstruction via 3d gaussian splatting," *arXiv preprint arXiv:2401.09720*, 2024.

[185] H. Luo, M. Ouyang, Z. Zhao, S. Jiang, L. Zhang, Q. Zhang, W. Yang, L. Xu, and J. Yu, "Gaussianhair: Hair modeling and rendering with light-aware gaussians," *arXiv preprint arXiv:2402.10483*, 2024.

[186] Y. Chen, Z. Chen, C. Zhang, F. Wang, X. Yang, Y. Wang, Z. Cai, L. Yang, H. Liu, and G. Lin, "Gaussianeditor: Swift and controllable 3d editing with gaussian splatting," *arXiv preprint arXiv:2311.14521*, 2023.

[187] J. Fang, J. Wang, X. Zhang, L. Xie, and Q. Tian, "Gaussianeditor: Editing 3d gaussians delicately with text instructions," *arXiv preprint arXiv:2311.16037*, 2023.

[188] J. Huang and H. Yu, "Point'n move: Interactive scene object manipulation on gaussian splatting radiance fields," *arXiv preprint arXiv:2311.16737*, 2023.

[189] K. Lan, H. Li, H. Shi, W. Wu, Y. Liao, L. Wang, and P. Zhou, "2d-guided 3d gaussian segmentation," *arXiv preprint arXiv:2312.16047*, 2023.

[190] J. Zhuang, D. Kang, Y.-P. Cao, G. Li, L. Lin, and Y. Shan, "Tip-editor: An accurate 3d editor following both text-prompts and image-prompts," *arXiv preprint arXiv:2401.14828*, 2024.

[191] B. Dou, T. Zhang, Y. Ma, Z. Wang, and Z. Yuan, "Cosseggaussians: Compact and swift scene segmenting 3d gaussians," *arXiv preprint arXiv:2401.05925*, 2024.

[192] X. Hu, Y. Wang, L. Fan, J. Fan, J. Peng, Z. Lei, Q. Li, and Z. Zhang, "Semantic anything in 3d gaussians," *arXiv preprint arXiv:2401.17857*, 2024.

[193] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. J. Engel, R. Mur-Artal, C. Ren, S. Verma *et al.*, "The replica dataset: A digital replica of indoor spaces," *arXiv preprint arXiv:1906.05797*, 2019.

[194] J. Park, Q.-Y. Zhou, and V. Koltun, "Colored point cloud registration revisited," in *ICCV*, 2017, pp. 143–152.

[195] E. Sucar, S. Liu, J. Ortiz, and A. J. Davison, "imap: Implicit mapping and positioning in real-time," in *ICCV*, 2021, pp. 6229–6238.

[196] X. Yang, H. Li, H. Zhai, Y. Ming, Y. Liu, and G. Zhang, "Vox-fusion: Dense tracking and mapping with voxel-based neural implicit representation," in *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2022, pp. 499–507.

[197] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, "Nice-slam: Neural implicit scalable encoding for slam," in *CVPR*, 2022, pp. 12 786–12 796.

[198] M. M. Johari, C. Carta, and F. Fleuret, "Eslam: Efficient dense slam system based on hybrid representation of signed distance fields," in *CVPR*, 2023, pp. 17 408–17 419.

[199] E. Sandström, Y. Li, L. Van Gool, and M. R. Oswald, "Point-slam: Dense neural point cloud-based slam," in *ICCV*, 2023, pp. 18 433–18 444.

[200] H. Wang, J. Wang, and L. Agapito, "Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam," in *CVPR*, 2023, pp. 13 293–13 302.

[201] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang, "The apolloscape dataset for autonomous driving," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 954–960.

[202] J. Wang, J. Liu, and N. Kato, "Networking and communications in autonomous driving: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1243–1274, 2018.

[203] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *CVPR*, 2020, pp. 2446–2454.

[204] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *CVPR*, 2020, pp. 11 621–11 631.

[205] J. Guo, U. Kurup, and M. Shah, "Is it safe to drive? an overview of factors, metrics, and datasets for driveability assessment in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 8, pp. 3135–3151, 2019.

[206] K. Muhammad, A. Ullah, J. Lloret, J. Del Ser, and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4316–4336, 2020.

[207] S. Teng, X. Hu, P. Deng, B. Li, Y. Li, Y. Ai, D. Yang, L. Li, Z. Xuanyuan, F. Zhu *et al.*, "Motion planning for autonomous driving: The state of the art and future perspectives," *IEEE Transactions on Intelligent Vehicles*, 2023.

[208] Y. Hu, J. Yang, L. Chen, K. Li, C. Sima, X. Zhu, S. Chai, S. Du, T. Lin, W. Wang *et al.*, "Planning-oriented autonomous driving," in *CVPR*, 2023, pp. 17 853–17 862.

[209] X. Ma, W. Ouyang, A. Simonelli, and E. Ricci, "3d object detection from images for autonomous driving: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[210] J. Mao, S. Shi, X. Wang, and H. Li, "3d object detection for autonomous driving: A comprehensive survey," *International Journal of Computer Vision*, pp. 1–55, 2023.

[211] D. Fu, X. Li, L. Wen, M. Dou, P. Cai, B. Shi, and Y. Qiao, "Drive like a human: Rethinking autonomous driving with large language models," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 910–919.

[212] J. Wang, Y. Yuan, Z. Luo, K. Xie, D. Lin, U. Iqbal, S. Fidler, and S. Khamis, "Learning human dynamics in autonomous driving scenarios," in *ICCV*, 2023, pp. 20 796–20 806.

[213] X. Zhou, Z. Lin, X. Shan, Y. Wang, D. Sun, and M.-H. Yang, "Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes," *arXiv preprint arXiv:2312.07920*, 2023.

[214] Y. Yan, H. Lin, C. Zhou, W. Wang, H. Sun, K. Zhan, X. Lang, X. Zhou, and S. Peng, "Street gaussians for modeling dynamic urban scenes," *arXiv preprint arXiv:2401.01339*, 2024.

[215] H. Zhou, J. Shao, L. Xu, D. Bai, W. Qiu, B. Liu, Y. Wang, A. Geiger, and Y. Liao, "Hugs: Holistic urban 3d scene understanding via gaussian splatting," *arXiv preprint arXiv:2403.12722*, 2024.

[216] Y. Huang, B. Cui, L. Bai, Z. Guo, M. Xu, and H. Ren, "Endo-4dgs: Distilling depth ranking for endoscopic monocular scene reconstruction with 4d gaussian splatting," *arXiv preprint arXiv:2401.16416*, 2024.

[217] Y. Liu, C. Li, C. Yang, and Y. Yuan, "Endogaussian: Gaussian splatting for deformable surgical scene reconstruction," *arXiv preprint arXiv:2401.12561*, 2024.

[218] L. Zhu, Z. Wang, Z. Jin, G. Lin, and L. Yu, "Deformable endoscopic tissues reconstruction with gaussian splatting," *arXiv preprint arXiv:2401.11535*, 2024.

[219] K. Wang, C. Yang, Y. Wang, S. Li, Y. Wang, Q. Dou, X. Yang, and W. Shen, "Endogslam: Real-time dense reconstruction and tracking in endoscopic surgeries using gaussian splatting," *arXiv preprint arXiv:2403.15124*, 2024.

[220] Y. Cai, Y. Liang, J. Wang, A. Wang, Y. Zhang, X. Yang, Z. Zhou, and A. Yuille, "Radiative gaussian splatting for efficient x-ray novel view synthesis," *arXiv preprint arXiv:2403.04116*, 2024.

[221] Y. Li, X. Fu, S. Zhao, R. Jin, and S. K. Zhou, "Sparse-view ct reconstruction with 3d gaussian volumetric representation," *arXiv preprint arXiv:2312.15676*, 2023.

[222] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *2012 IEEE/RSJ international conference on intelligent robots and systems*, 2012, pp. 573–580.

[223] J. Fang, T. Yi, X. Wang, L. Xie, X. Zhang, W. Liu, M. Nießner, and Q. Tian, "Fast dynamic radiance fields with time-aware neural voxels," in *SIGGRAPH Asia 2022 Conference Papers*, 2022, pp. 1–9.

[224] A. Cao and J. Johnson, "Hexplane: A fast representation for dynamic scenes," in *CVPR*, 2023, pp. 130–141.

[225] F. Wang, Z. Chen, G. Wang, Y. Song, and H. Liu, "Masked space-time hash encoding for efficient dynamic scene reconstruction," in *NeurIPS*, 2023.

[226] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE TIP*, vol. 13, no. 4, pp. 600–612, 2004.

[227] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018, pp. 586–595.

[228] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, "Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields," in *ICCV*, 2021, pp. 5855–5864.

[229] K. Rematas, A. Liu, P. P. Srinivasan, J. T. Barron, A. Tagliasacchi, T. Funkhouser, and V. Ferrari, "Urban radiance fields," in *CVPR*, 2022, pp. 12 932–12 942.

[230] Z. Xie, J. Zhang, W. Li, F. Zhang, and L. Zhang, "S-nerf: Neural radiance fields for street views," in *ICLR*, 2023.

[231] H. Turki, J. Y. Zhang, F. Ferroni, and D. Ramanan, "Suds: Scalable urban dynamic scenes," in *CVPR*, 2023, pp. 12 375–12 385.

[232] S. Peng, Y. Zhang, Y. Xu, Q. Wang, Q. Shuai, H. Bao, and X. Zhou, "Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans," in *CVPR*, 2021, pp. 9054–9063.

[233] S. Peng, J. Dong, Q. Wang, S. Zhang, Q. Shuai, X. Zhou, and H. Bao, "Animatable neural radiance fields for modeling dynamic human bodies," in *ICCV*, 2021, pp. 14 314–14 323.

[234] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "pixelnerf: Neural radiance fields from one or few images," in *CVPR*, 2021, pp. 4578–4587.

[235] Y. Kwon, D. Kim, D. Ceylan, and H. Fuchs, "Neural human performer: Learning generalizable radiance fields for human performance rendering," in *NeurIPS*, 2021, pp. 24 741–24 752.

[236] C.-Y. Weng, B. Curless, P. P. Srinivasan, J. T. Barron, and I. Kemelmacher-Shlizerman, "Humannerf: Free-viewpoint rendering of moving people from monocular video," in *CVPR*, 2022, pp. 16 210–16 220.
[237] C. Geng, S. Peng, Z. Xu, H. Bao, and X. Zhou, "Learning neural volumetric representations of dynamic humans in minutes," in *CVPR*, 2023, pp. 8759–8770.
[238] Y. Wang, Y. Long, S. H. Fan, and Q. Dou, "Neural rendering for stereo 3d reconstruction of deformable tissues in robotic surgery," in *MICCAI*, 2022, pp. 431–441.
[239] R. Zha, X. Cheng, H. Li, M. Harandi, and Z. Ge, "Endosurf: Neural surface reconstruction of deformable tissues with stereo endoscope videos," in *MICCAI*, 2023, pp. 13–23.
[240] C. Yang, K. Wang, Y. Wang, X. Yang, and W. Shen, "Neural lerplane representations for fast 4d reconstruction of deformable tissues," in *MICCAI*, 2023, pp. 46–56.
[241] A. Guédon and V. Lepetit, "Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering," *arXiv preprint arXiv:2311.12775*, 2023.
[242] T. Xie, Z. Zong, Y. Qiu, X. Li, Y. Feng, Y. Yang, and C. Jiang, "Physgaussian: Physics-integrated 3d gaussians for generative dynamics," *arXiv preprint arXiv:2311.12198*, 2023.
[243] J. Abou-Chakra, K. Rana, F. Dayoub, and N. Sünderhauf, "Physically embodied gaussian splatting: Embedding physical priors into a visual 3d world model for robotics," in *Conference on Robot Learning*, no. 7th, 2023.
[244] G. Lu, S. Zhang, Z. Wang, C. Liu, J. Lu, and Y. Tang, "Manigaussian: Dynamic gaussian splatting for multi-task robotic manipulation," *arXiv preprint arXiv:2403.08321*, 2024.
[245] Y. Liu, H. Guan, C. Luo, L. Fan, J. Peng, and Z. Zhang, "Citygaussian: Real-time high-quality large-scale scene rendering with gaussians," *arXiv preprint arXiv:2404.01133*, 2024.
[246] Y. Sun, X. Wang, Y. Zhang, J. Zhang, C. Jiang, Y. Guo, and F. Wang, "icomma: Inverting 3d gaussians splatting for camera pose estimation via comparing and matching," *arXiv preprint arXiv:2312.09031*, 2023.
[247] C. Pokhariya, I. N. Shah, A. Xing, Z. Li, K. Chen, A. Sharma, and S. Sridhar, "Manus: Markerless hand-object grasp capture using articulated 3d gaussians," *arXiv preprint arXiv:2312.02137*, 2023.
[248] W. Jiang, B. Lei, and K. Daniilidis, "Fisherrf: Active view selection and uncertainty quantification for radiance fields using fisher information," *arXiv preprint arXiv:2311.17874*, 2023.
[249] Y. Feng, X. Feng, Y. Shang, Y. Jiang, C. Yu, Z. Zong, T. Shao, H. Wu, K. Zhou, C. Jiang *et al.*, "Gaussian splashing: Dynamic fluid synthesis with gaussian splatting," *arXiv preprint arXiv:2401.15318*, 2024.