

论文引用格式:

# 基于物理的可微渲染综述

邢健开<sup>1,2,3</sup>, 徐昆<sup>1,2,3</sup>

- 普适计算教育部重点实验室
- 北京信息科学与技术国家研究中心
- 清华大学计算机系, 北京 100084

**摘要:** 可微渲染技术是当前计算机图形学与计算机视觉方向的研究热点, 其目标是将计算机图形学中的渲染流水线进行可微化改造以支持计算渲染的输出图像关于输入参数如几何、材质等参数的梯度。结合渲染图像与目标图像之间的损失函数, 可微渲染允许我们在生成式分析的框架中, 通过梯度下降的方式从图像中推理出场景参数, 是解决三维重建、逆向渲染等领域问题的有效方法, 并在虚拟现实、自动驾驶等领域有着广泛的应用前景。基于物理的可微渲染旨在对基于物理的渲染管线进行可微化改造, 主要涉及对场景几何和材质的表达, 以及光路传输模拟过程的梯度计算方法。本文对近年来基于物理的可微渲染领域的发展情况进行了调研, 总结了基于物理的可微渲染研究进展。本文分为五个部分: 第一部分总体介绍正向渲染和可微渲染的计算方法; 第二部分介绍如何针对具体的几何、材质、相机的表达方式进行梯度计算; 第三部分讨论如何提高可微渲染的效率和鲁棒性; 第四部分展示可微渲染如何应用于实际任务中; 在第五部分中, 本文将展望可微渲染的发展趋势, 期望推动该领域的进一步发展。

**关键词:** 渲染; 可微渲染; 逆向渲染; 光线跟踪; 三维重建

## Physically based Differentiable Rendering: A Survey

Xing Jiankai<sup>1,2,3</sup>, Xu Kun<sup>1,2,3</sup>

- Key Laboratory of Pervasive Computing, Ministry of Education
- Beijing National Research Center for Information Science and Technology
- Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China

**Abstract:** Rendering has been a prominent subject in the field of computer graphics for an extended period. It can be regarded as a function that accepts an abstract scene description as input and typically generates a 2D image as output. The theory and practice of rendering have significantly advanced through years of research. In recent years, inverse rendering has emerged as a new research focus in the field of computer graphics due to the development of digital technology. The objective of inverse rendering is to reverse the rendering process and deduce scene parameters from the output image, which is equivalent to solving the inverse function of the rendering function. This plays a crucial role in addressing perception problems in diverse advanced technological domains including virtual reality, autonomous driving, and robotics. Numerous methods exist for implementing inverse rendering, with the current mainstream framework being optimization through “analysis by synthesis” process. Firstly, it estimates a set of initial scene parameters, then performs forward rendering on the scene, compares the rendered result with

收稿日期:

; 修回日期:

基金项目: 国家自然科学基金项目 (项目编号: 61932003)

Supported by: National Natural Science Foundation of China (Project Number: 61932003)

the target image, and then minimizes the difference (loss function) by optimizing the scene parameters using gradient descent based method. This pipeline necessitates the ability to compute the derivatives of the output image in forward rendering with respect to the input parameters. Consequently, differentiable rendering has emerged to fulfill this requirement. Specifically, the research topic of differentiable rendering is to convert the forward rendering pipeline in computer graphics into a differentiable form, enabling the differentiation of the output image with respect to input parameters like geometry, material, light source, and camera. Currently, forward rendering can be broadly categorized into three types: rasterization-based rendering, physically based rendering, and the emerging neural rendering. Rasterization-based rendering is a fundamental technique in computer graphics that converts geometric shapes into pixels for display. It involves projecting 3D objects onto a 2D screen, performing hidden surface removal, shading, and texturing to create realistic images efficiently. While rasterization is fast and suitable for real-time applications, it may lack physical accuracy in simulating light interactions. Physically based rendering, on the other hand, aims to accurately simulate real-world light behavior by considering physical properties of materials, light sources, and the environment. It calculates how light rays interact with surfaces, accounting for reflections, refractions, and scattering to produce photorealistic visual results. This method prioritizes realism and is widely used in industries such as animation, gaming, and visual effects. Neural rendering is an emerging rendering technique in recent years, mainly used for image-based rendering tasks. Unlike traditional graphics rendering, image-based rendering does not require any explicit 3D scene information (geometry, materials, lighting, etc.), but instead implicitly encodes scenes through a sequence of 2D images sampled from different viewpoints, enabling the generation of images of the scene from any viewpoint. Accordingly, differentiable rendering can also be categorized into three types: differentiable rasterization, physically based differentiable rendering, and differentiable neural rendering. In differentiable rasterization, many works employ approximate methods to compute approximate derivatives of the rasterization process for backpropagation of gradients, or modify steps in the traditional rendering pipeline (usually rasterization and testing/blending steps) to make pixels differentiable with respect to vertices. Neural rendering is naturally differentiable because its rendering process is conducted through neural networks. For physically based differentiable rendering, because of the intricate nature of geometry, material, and light transmission processes, accurately calculating the gradient of the image concerning scene parameters is more challenging. Therefore, this article concentrates on recent research in the field of physically based differentiable rendering. The article is organized into the following sections: Chapter 1 introduces the computational methods of forward rendering and differentiable rendering from an abstract standpoint and two types of method for correctly computing boundary integral: edge sampling and reparameterization. Chapter 2 explores the attainment of differentiable rendering for distinct representations of geometry like volumetric representation, signed distance field, height field and vectorized geometry, material like volumetric material, parameterized BRDF (bidirectional reflectance distribution function), BSSRDF (bidirectional surface scattering reflectance distribution function) and continuously varying refractive index fields, and camera related parameters like pixel reconstruction filter and TOF (time of flight) camera. Chapter 3 focuses on enhancing the efficiency and robustness of differentiable rendering, including efficiently sampling, high efficiency system & framework & language for differentiable rendering and several techniques to enhance the robustness of differentiable rendering. Chapter 4 showcases the application of differentiable rendering in practical tasks, which can be generally divided in three types: single object reconstruction, object and environment light reconstruction and scene reconstruction. Lastly in Chapter 5, the future development trends of differentiable rendering are discussed, including improving efficiency, robustness of differentiable rendering and combining differentiable rendering with other methods.

**Key words:** rendering; differentiable rendering; inverse rendering; ray tracing; 3d reconstruction.

# 0 引言

渲染是图形学领域多年来的研究热点，其可以被视作一个函数，输入为对场景的抽象描述，输出一般为二维图像，渲染的理论和实践在多年的研究中已日趋成熟。近年来，随着数字化技术的发展，图形学领域的一个新的研究热点是逆向渲染，逆向渲染的目的就是要将渲染过程反过来，从输出的图像中推断出场景参数，或者说，求解渲染函数的反函数，这对于解决多个先进科技领域如虚拟现实、自动驾驶、机器人中涉及到的感知问题有着重要的作用。

逆向渲染的实现方法有很多种，而当前主流的方法是生成式分析（analysis by synthesis）：首先估计场景参数，然后进行渲染，比较渲染结果和目标图片的差异（损失函数），使用基于梯度下降的方法通过优化场景参数最小化该损失函数。这种方式需要求解出正向渲染中输出的图片关于输入参数的梯度，于是可微渲染便在这种需求下诞生了。

具体的来说，可微渲染的研究内容是将计算机图形学中的正向渲染流水线进行可微化改造以支持计算渲染的输出图像关于输入参数如几何、材质、光源、视点等信息的梯度。根据不同的正向渲染方法，实现可微渲染的方式也不同。正向渲染如今可大致分为三种：基于光栅化的渲染、基于物理的渲染和近年来新兴的神经渲染。基于光栅化的渲染通过将物体在像素平面进行消隐、遮蔽、着色等处理，最后生成一个二维的光栅图像。基于物理的渲染则通过模拟现实世界中的几何、材质和光线传输过程来得到一幅具有真实感的图片。神经渲染则是一种新兴的渲染方法，通过给定视角的图像训练神经网络模型，以生成新视角下的图像。依据所采用的正向渲染方法，可微渲染也可相应的分为三类。光栅化可微渲染通过对光栅化过程进行可微化改造或近似梯度的方式实现可微渲染；神经渲染由于神经网络的天然可微性，直接通过自动微分技术便可实现可微渲染，其主要求解关于神经网络参数的梯度；而基于物理的可微渲染要准确地求解图像关于场景参数的梯度，由于几何、材质、光传输过程的高度复杂性，该梯度的求解也较为复杂。本文也将着重介绍近年来基于物理的可微渲染方向的研究内容。对于其他类型的可微渲染，读者可参考（Ye 等，2021）和（Xu 等，2021）两篇综述文章。为了简便，除非特殊表明，后文中将以（正向）渲染和可微渲染指代

基于物理的渲染和基于物理的可微渲染。

本文将分成如下几个部分：第一章，本文将基于渲染方程，介绍可微渲染的理论与实际应用中所使用的几种计算方法；第二章，本文将深入渲染的每个部分，介绍如何针对具体的几何、材质、相机的表达方式对渲染进行可微渲染；第三章本文将讨论如何提高可微渲染的效率、鲁棒性；第四章本文将展示可微渲染如何应用于实际任务中；最后本文将展望可微渲染的发展趋势。

## 1 基于物理的正向渲染与可微渲染

本章将首先回顾正向渲染涉及到的计算内容，然后介绍可微渲染要求解的内容的形式化表达，引入可微渲染中特有的边界积分项，最后介绍两类求解边界积分的实际算法。

### 1.1 基于物理的正向渲染

基于物理的正向渲染是计算机图形学领域研究了数十年的问题，其目标为在给定抽象的场景参数如物体几何、材质、光照、视角的情况下，如何通过正确的模拟真实世界中的光线传输过程，将其转化为一幅具有真实感的图像。为实现这一点，正向渲染的核心研究内容是求解渲染方程（Kajiya 等，1986）：

$$L_o(\mathbf{x}, \vec{\omega}) = L_e(\mathbf{x}, \vec{\omega}) + \int_{\Omega} f_r(\mathbf{x}, \vec{\omega}', \vec{\omega}) L_i(\mathbf{x}, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}' \quad (1)$$

式中，等号左边的  $L_o(\mathbf{x}, \vec{\omega})$  为在场景中某一点  $\mathbf{x}$  在方向  $\vec{\omega}$  上的出射光（辐射度），而右边分别为该点在该方向的自发光项  $L_e(\mathbf{x}, \vec{\omega})$  和该点在各个方向接收到的入射光在出射方向的反射或折射项，后者通常为一个定义在点  $\mathbf{x}$  对应的球面或以法线为中心的半球面上对入射方向角  $\vec{\omega}'$  的积分，被积函数为双向散射分布函数  $f_r(\mathbf{x}, \vec{\omega}', \vec{\omega})$ ，入射光  $L_i(\mathbf{x}, \vec{\omega}')$  和入射角带来的衰减项  $(\vec{\omega}' \cdot \vec{n})$  的乘积。其中入射光项需要通过光线追踪等方式进行递归求解。为简化，接下来将需要计算的积分写作如下形式：

$$I(\boldsymbol{\pi}) = \int_{\chi} f(\mathbf{x}, \boldsymbol{\pi}) d\mathbf{x} \quad (2)$$

式中， $f$  是定义在  $\chi$  上的函数，定义域  $\chi$  一般为像素区域或着色点对应的半球面， $\boldsymbol{\pi}$  一般为场景参数集合。

由于被积函数较为复杂，积分通常难以计算解析解，因此，我们一般会用蒙特卡洛方法计算积分：

$$E(I) = \frac{1}{N} \sum_{i=1}^N \frac{f(\mathbf{x}_i, \boldsymbol{\pi})}{p(\mathbf{x}_i)} \quad (3)$$

式中， $N$ 为采样数， $p(\mathbf{x})$ 为蒙特卡洛采样使用的概率密度函数。通过合适的采样方法，随着采样数的增多，对积分 $E(I)$ 的估计值会逐步收敛到真实值 $I$ 。

### 1.2 基于物理的可微渲染

对于基于物理的可微渲染，其目的是求解渲染过程中的积分 $I$ 关于任一场景参数 $\boldsymbol{\pi}$  ( $\boldsymbol{\pi} \in \boldsymbol{\pi}$ )的偏导数（用 $\partial_{\boldsymbol{\pi}} I$ 表示 $\partial I / \partial \boldsymbol{\pi}$ ）：

$$\partial_{\boldsymbol{\pi}} I = \partial_{\boldsymbol{\pi}} \left[ \int_{\mathcal{X}} f(\mathbf{x}, \boldsymbol{\pi}) d\mathbf{x} \right] \quad (4)$$

微积分的运算规则告诉我们如果函数定义域 $\mathcal{X}$ 与 $\boldsymbol{\pi}$ 无关，且被积函数 $f$ 关于 $\boldsymbol{\pi}$ 是连续的，那么可以交换微分和积分符号的位置：

$$\partial_{\boldsymbol{\pi}} I = \partial_{\boldsymbol{\pi}} \left[ \int_{\mathcal{X}} f(\mathbf{x}, \boldsymbol{\pi}) d\mathbf{x} \right] = \int_{\mathcal{X}} \partial_{\boldsymbol{\pi}} f(\mathbf{x}, \boldsymbol{\pi}) d\mathbf{x} \quad (5)$$

然而，函数的定义域可能与参数 $\boldsymbol{\pi}$ 相关，被积函数也可能存在关于参数 $\boldsymbol{\pi}$ 的不连续处，在这种情况下，交换积分和微分的顺序会使得计算结果不正确。为了正确的求解积分，研究者们引入雷诺传输定理(Leal等, 2007)，将积分域上的被积函数分为连续的部分和不连续的部分并分开处理，具体的公式可表示为：

$$\partial_{\boldsymbol{\pi}} I = \int_{\mathcal{X}} \partial_{\boldsymbol{\pi}} f(\mathbf{x}, \boldsymbol{\pi}) d\mathbf{x} + \int_{\partial \mathcal{X}} f(\mathbf{x}, \boldsymbol{\pi}) \langle \partial_{\boldsymbol{\pi}} \mathbf{x}, \hat{\mathbf{n}}(\mathbf{x}) \rangle d\mathbf{x} \quad (6)$$

式中， $\partial \mathcal{X}$ 表示定义域中 $f$ 关于参数 $\boldsymbol{\pi}$ 不连续的边界部分， $\hat{\mathbf{n}}(\mathbf{x})$ 表示 $\mathbf{x}$ 在边界上的法线。一般地，称公式右边的第一项为内部积分，第二项为边界积分。下面分别介绍两项积分的计算方法。

### 1.3 内部积分的计算方法

对于内部积分，可以发现，内部积分和正向过程的积分类似，都是计算某一函数在（半）球面积分，所以可以复用正向渲染中的蒙特卡洛方法进行计算。例如，针对渲染方程（公式(1)），在忽略边界积分项和几何项的情况下，根据导数求解的链式法则，可以将其关于参数 $\boldsymbol{\pi}$ 的导数写为：

$$\begin{aligned} \partial_{\boldsymbol{\pi}} L_o(\mathbf{x}, \bar{\boldsymbol{w}}) &= \partial_{\boldsymbol{\pi}} L_e(\mathbf{x}, \bar{\boldsymbol{w}}) + \\ &\int_{\Omega} [\partial_{\boldsymbol{\pi}} L_i(\mathbf{x}, \bar{\boldsymbol{w}}') f_r(\mathbf{x}, \bar{\boldsymbol{w}}, \bar{\boldsymbol{w}}') \\ &+ L_i(\mathbf{x}, \bar{\boldsymbol{w}}') \partial_{\boldsymbol{\pi}} f_r(\mathbf{x}, \bar{\boldsymbol{w}}, \bar{\boldsymbol{w}}')] d\bar{\boldsymbol{w}}' \end{aligned} \quad (7)$$

当然，由于此处的被积函数与正向渲染时的被积函数形态不同，所以高效计算两者的方法也有所不同，本文将在第三章讨论如何高效地计算内部积

分。

### 1.4 边界积分的计算方法

边界积分是可微渲染中独有的部分，相比于通常定义在像素内或半球面等二维定义域上的积分，边界积分定义在函数的不连续处，通常为一条或多条一维边界构成的集合，该积分和内部积分一样，很难求解解析解，所以也需要进行采样估计。然而，正向渲染和内部积分中使用的蒙特卡洛采样通常为基于面积的采样方法，而用这种方法采样到边界的概率为0，所以无法用来计算边界积分。因此，接下来介绍两类正确计算边界积分的方法，并根据计算边界积分的方式和渲染方法在表1中进行了分类。

表1. 边界积分计算方法分类

Table 1. Taxonomy of computing boundary integral

	光线空间	路径空间
边界采样	Li 等, 2018	Zhang 等, 2020
重参数化	Loubet 等, 2019 Bangaru 等, 2020	Xu 等, 2023

#### 1.4.1 基于边界采样的计算方法

Li等人(2018)首次提出可微渲染中边界积分的概念并针对边界积分提出了一种最直接的处理方式，即直接找到每个边界积分定义域对应的边界，并在这些边界上进行采样，图1中展示了边界采样和面积采样的区别。找到这样的边界非常复杂，因为这些边界可能出现于任意物体的几何边缘、视点相关的由物体之间的遮挡现象造成的边界，甚至还有反射折射等现象造成的边界。Li等人(2018)建立了一个数据结构存储所有的边界，以加速查找不连续处的过程，但算法仍然效率不高，且无法找到由镜面反射或折射造成的不连续处。

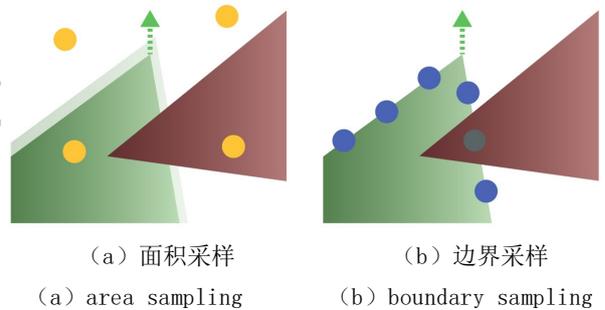


图1 面积采样与边界采样(Li等, 2018)

Fig.1 Area sampling and boundary sampling (Li et. 2018)

为了处理更复杂的场景并提高采样效率，Zhang等人(2020)将可微渲染扩展到了路径空间渲染中，路

径空间渲染同样需要计算积分，而且其积分的定义域为路径空间，相比像素平面或半球面的维度更高。基于路径空间的渲染允许使用更复杂的采样算法如 MLT (metropolis light transport, 梅特罗波利斯光线传输)，更多样的路径构建方式如 BDPT (bidirectional path tracing, 双向路径追踪) 来提高渲染效率。针对路径空间积分的可微渲染同样需要处理边界积分的问题，且由于定义域维度更高，边界的定义也更为复杂。Zhang 等人 (2020) 提出了路径空间下的可微渲染理论，定义了路径空间中的边界和正确的梯度计算方法。其解决边界积分的方案分为两部分，其一是通过对场景进行参数化变换，将场景本身的边界对应的边界积分项转为内部积分，其二是通过从任意边界点出发构造位于路径空间边界上的光路以处理视点相关的边界，相比 (Li 等, 2018) 提出的方法，该方法不需要进行低效的边界查询，效率更高，图 2 展示了两种方法求解出的梯度，(Zhang 等, 2020) 能在更短的时间内能获得噪声更小的结果。但该方法的局限性在于需要对场景的参数化变换，且同样无法处理纯镜面反射产生的边界。

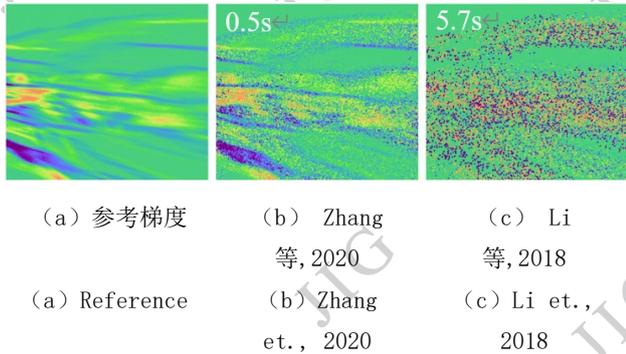


图 2 基于边界采样的两种方法结果对比 (Zhang 等, 2020)  
Fig.2 Gradient comparison between two edge sampling based method (Zhang et., 2020)

#### 1.4.2 基于重参数化的计算方法

基于边界采样的方法有两个缺点，一是效率不高，二是和传统的基于面积采样的方案不兼容。所以研究者们想到能否通过某种方式将定义在边界上的积分变换为基于面积采样的方式求解，并称这种方式为重参数化。Loubet 等人 (2019) 首先提出了一种基于变量替换的方法，即通过一个与求导参数相关的球面参数化变换函数，使得经过变换之后原积分的值不变，而变换之后的不连续处与待求导参数无关，这使得边界积分为 0，不再需要特殊处理，但该方法提出的变换会使得求出的梯度有偏。后来，

Bangaru 等人 (2020) 利用散度定理，找到一个变换场 (warping field) 将边界积分转变为另一个面积积分：

$$\oint_{\partial\chi} f(\mathbf{x}, \boldsymbol{\pi}) \langle \partial_{\boldsymbol{\pi}} \mathbf{x}, \mathbf{n}(\mathbf{x}) \rangle d\mathbf{x} = \iint_{\chi'} \nabla_{\boldsymbol{\omega}} \cdot (f(\boldsymbol{\omega}, \boldsymbol{\pi}) \bar{\mathbf{V}}_{\boldsymbol{\pi}}(\boldsymbol{\omega})) d\boldsymbol{\omega} \quad (8)$$

式中， $\bar{\mathbf{V}}_{\boldsymbol{\pi}}(\boldsymbol{\omega})$  为一个变换场，将只在边界有定义的边界速度项  $\langle \partial_{\boldsymbol{\pi}} \mathbf{x}, \mathbf{n}(\mathbf{x}) \rangle$  扩展到整个定义域，作者通过约束该变换场，保证了转换之后的积分的无偏性。图 3 展示了两种方法将边界积分转化为面积积分后的被积函数在一些采样点的值 (向量)。

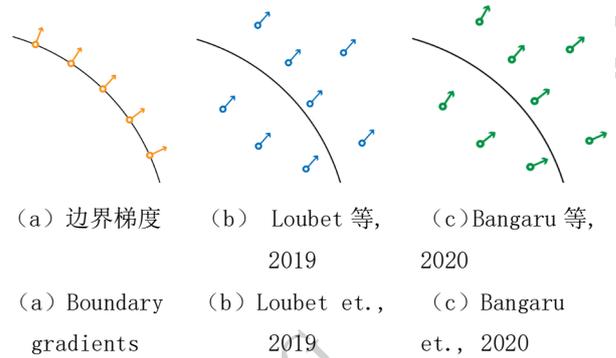


图 3 重参数化方法示意 (Bangaru 等, 2020)

Fig.3 Illustration of reparameterization method (Bangaru et., 2020)

Xu 等人 (2023) 进一步将这种基于变换场的变换扩展到了路径空间中，结合了路径空间可微渲染 (Zhang 等, 2020) 中高效的采样方式与重参数化计算边界积分的优点，获得了较好的效果，图 4 为在同等采样数量下，Xu 等人 (2023) 与 Bangaru 等人 (2020) 得到的梯度的对比。

这种基于重参数化的方案虽然可以避免边界采样，但还是需要通过发射若干辅助光线来估测边界可能出现的位置，据此来计算变换场，这一方面增加了时间上的开销，另一方面也引入了由估计造成的噪声。

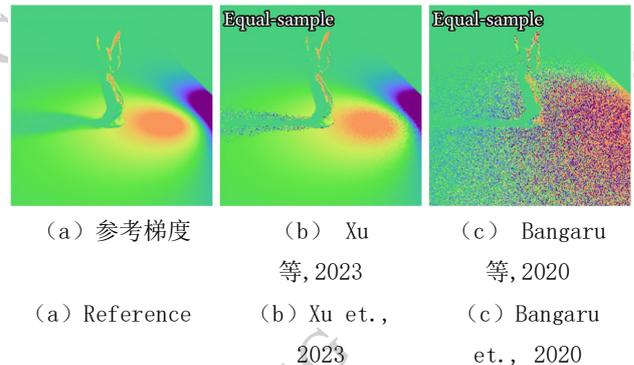


图 4 基于重参数化的两种方法结果对比 (Xu 等, 2023)

Fig.4 Gradient comparison between two reparameterization based method (Xu et., 2023)

需要说明的是,上述理论和方法都对实际渲染中涉及到的被积函数做了简化,只从整体上给出了正确计算渲染方程梯度的几种方案。下一章中,本文将展开渲染方程,分别讨论如何针对被积函数中各种类型的参数进行可微渲染。

## 2 各类型参数的可微渲染方法

基于物理的正向渲染中包含多种类型的参数,其中最重要的是几何和材质参数,可微渲染的研究重点也是针对这两种参数的梯度计算方法。由于几何参数的梯度受不连续部分的影响较大,故大多数关于几何的基于物理的可微渲染方法在(Li等,2018)引入边界积分的概念后才被提出,而材质相关的导数由于一般与边界积分无关,所以早些年便有研究。本章中将分别介绍针对不同几何表达方式和材质类型的可微渲染方法,另外也对不同的相机模型的可微渲染方法进行简要介绍。

### 2.1 几何参数微分

渲染中,有很多种表示物体几何的方法,包括网格(Mesh)、体表达、有向距离场(SDF)等。对于部分表达方式,在求出光线与物体的交点后,直接通过自动微分即可计算梯度,如网格表达,但对于某些表达方式,需要用特殊的方式来计算相关的导数,以保证正确性或提高效率。

体表达是图形学中常用于表达烟雾或某些半透明物体的几何描述方式,该方式将物体视作一群粒子,光线穿过物体时可能和粒子发生碰撞、吸收和散射等现象。Zhang等人(2019)首先提出辐射度传输方程的微分方法,将Li等人(2018)的基于边界采样的可微渲染方法扩展到了体表达中,使其支持对体渲染中吸收、散射现象的梯度计算。Zhang等人(2021a)将(Zhang等,2019)和(Zhang等,2020)的工作结合在一起,提出了路径空间中体表达的可微渲染方法。Yu等人(2022)弱化了体表达下路径空间可微渲染方法中要求提供场景物体变化率的约束,提出了一种重参数化方法,使得只需要知道物体边界的变化率即可正确估计梯度。Nimier-David等人(2022)提出了一种无偏的针对体表达可微渲染的采样技术,其提出了微分比率追踪(differential Ratio Tracking)技术,以透过率(transmittance)进行采样,结合蓄水池采样技术,避免了以往方法在低密度区域出现的方差较大的问

题。

Vicini等人(2021)提出了一种针对有向距离场的可微渲染方法。有向距离场(SDF)是一种隐式的描述物体几何的方式,由于其连续、可变拓扑的特性,近年来在重建、生成领域应用较为广泛。该方法分别对着色梯度和不连续处的梯度进行了处理。对于着色梯度,其主要难点在于处理光线与物体的相交时间对于场景参数的梯度,这需要通过隐函数的理论处理。对于不连续处的梯度,该方法采用了重参数化的方式,用类似(Bangaru等,2020)的思路,构建了一个变换场(warping field)将边界积分转化为了面积分,同时,此方法将构建变换场的过程融入了光线与SDF求交的ray marching过程中,使得其相比需要额外发射光线进行变换场构建的网格表达,具有更高的效率。同时期的另一个工作(Bangaru等,2021)也用类似的方法完成了SDF表达的可微渲染。

Tong等人(2023)提出了一种对于高度场表达的可微渲染方法。该方法进行了对于高度场可微渲染的理论推导,并利用了高度场的特殊性质设计了基于Mip-map的不连续处检测算法,在能用高度场表示的场景中取得了优于基线(Loubet等,2019)的效率。

Zhou等人(2021)提出了一种新的渲染方程求解方法,其将场景几何向量化,将渲染方程积分的定义域转化为若干三角形的集合,并将蒙特卡洛采样替换为基于光束追踪(beam tracing)的无偏积分,且该积分具有解析解,故通过自动微分技术即可获得准确的梯度。

Worchel和Alexa(2023)提出了一种参数化几何的可微渲染方法,该方法将用一些控制点生成的曲面可微地转为网格,并给出了网格顶点位置和法线关于控制点位置的梯度,结合一系列预条件和可微网格细分技术,实现了高质量参数化几何重建。

### 2.2 材质参数微分

渲染中物体的材质决定物体的表现和光线与物体相交后的传输方式(光源可视作一种发光材质)。由于现实世界的复杂性,材质的构成可能相当复杂,如玉石、皮肤、牛奶等。在图形学中,研究者提出了多种模型来描述这些材质。对于这些材质的可微渲染,由于渲染中的积分一般不含有与材质参数相关的不连续处,所以不涉及到边界积分的问题,因此,关于材质的可微渲染研究开始较早。

Gkioulekas等人(2013)提出首个通过求解渲染

染过程的梯度获取体表达材质中的散射参数的方法,该方法可以在给定准确光照、几何和一组关于物体的照片时,准确的求解像素颜色关于散射参数的导数,并通过梯度下降的方式优化散射参数的权重。论文中还给出了大量针对具体物体的测量结果,包括固体和液体。在此之后,研究人员提出了若干针对体表达材质的可微渲染工作:Khungurn 等人(2016)利用梯度优化布料的微外观模型参数,过程中用到了重新确定尺度的方法自适应决定各参数的优化范围和幅度;Gkioulekas 等人(2016)将(Gkioulekas 等, 2013)的方法扩展到了更广泛的异质媒介和更多的成像技术中;Zhao 等人(2016)利用可微渲染优化多个体素融合后的 albedo 和 phase function 组合权重,以减小下采样后的体表达和原始表达的渲染结果的差异。

Shem-Tov 等人(2020)提出利用可微渲染优化参数化 BRDF(bidirectional reflectance distribution function, 双向反射分布函数)模型的方法,并提出通过多次反射后的光路进行优化能获得更好的效果。

Deng 等人(2022)提出了一种针对 BSSRDF (bidirectional scattering-surface reflectance distribution function, 双向次表面散射分布函数)描述的材质的可微渲染方法,该方法将标准的 BSSRDF 做了一定简化,用入射角度项,出射角度项和入射点和出射点之间的距离项的乘积来定义表面任意两点的光线传输概率,然后采用路径空间的可微渲染框架进行材质梯度计算,并同时通过计算边界积分项来同时优化几何。

Teh 等人(2022)提出一种方法用于恢复连续可变折射率介质参数,该方法将(Nimier-David 等, 2019)的辐射度反向传播算法用于连续可变折射率介质中的光线传播,在梯度指数光学设计和气体流重建等任务上获得不错的结果。

### 2.3 相机参数微分

关于渲染的研究中对于相机参数,常用较为简单的针孔相机模型,并认为像素重建函数较为简单且可微。但对于存在不连续的像素重建函数和某些特殊类型的相机,我们需要特殊处理。

Yu 等人(2022)观察到常用的 Box 类型的像素重建函数中的不连续处对场景梯度的估计有着很大的贡献,于是他们研究了路径空间下对像素重建函数的可微化方法,给出了针对像素重建函数关于场景参数的梯度的计算公式,并分别针对内部和边界

积分给出了计算方法。

Wu 等人(2021)和 Yi 等人(2021)同时发表了关于针对 ToF(Time of Flight, 飞行时间)相机的可微渲染问题的研究,ToF 相机会将光路飞行时间引入渲染,使得渲染函数中会增加一个时间维度,且时间维度也会引入不连续性,故这两个方法都对路径空间可微渲染进行了在更高维度的扩展,以处理光飞行时间关于场景参数的导数,这种相机模型在恢复一些被遮挡的、不可目视的物体上表现较好。

## 3 可微渲染的效率与鲁棒性提升方法

### 3.1 高效可微渲染采样方法

渲染中采用的蒙特卡洛采样方法在采样数较少的情况下,往往存在较为严重的噪声,为了在相同时间内获得更好的渲染效果,研究者们对如何使用更高效的采样方式以降低噪声做了很多研究。可微渲染中同样需要考虑如何提高采样效率的问题。可微渲染与正向渲染寻找高效采样函数的目标不同,以渲染方程为例,传统渲染期望找到与入射光辐射度分布相近的采样函数,而可微渲染期望找到的是与入射光梯度分布相近的采样函数,这需要对微分分布的性质做出分析与估计。(Zeltner 等, 2021)中详细讨论了一系列包括是否对概率密度函数计算梯度、重要性采样与微分的先后顺序等问题在内的用于可微渲染的采样方法。Zhang 等人(2021b)观察到可微渲染中,对某些 BRDF 函数的梯度分布会在二维平面上呈现出奇对称的特点,故提出对称采样的方式,降低用蒙特卡洛方法估计梯度时出现的方差,Yu 等人(2022)将其扩展并应用到了像素重建函数上。

针对边界采样,也有一些加速方案,Yan 等人(2022)提出了一种加速采样方法,用于提高路径空间可微渲染中计算边界积分的效率,该方法提出用 KD 树对三维空间中的边界积分函数进行建模,根据函数值大小分配采样概率,并与多重重要性采样相结合。另外,这篇文章也提出了一种边界排序的方案,使得 KD 树的划分效率尽可能高。Zhang 等人(2023)则提出,将面积采样时获得的样本投影到边界上,并用网格或八叉树结构来估计边界样本的概率密度函数。

另外,由于可微渲染通常需要反复迭代,所以如何利用相邻迭代之间的相似性信息以节省计算开销是一个可行的研究方向。Nicolet 等人(2023)提出了通过递归控制变量和关联采样法,提高损失函

数准确度和梯度估计准确度。Balint 等人(2023)提出联合采样与优化,通过过去的样本估计相邻迭代间梯度的方差,并结合普通样本和梯度方差估计样本实现更加稳定的梯度估计。

实时渲染中的样本池重采样(ReSTIR)技术也可辅助可微渲染,Chang 等人(2023)在迭代过程中为每个优化参数保存一个样本池,并在新一轮迭代时基于样本池进行重采样以实现迭代间信息复用和高效光路采样。Wang 等人(2023)也采用 ReSTIR 方案进行样本复用,不过选择将样本池放置在场景而不是纹理或像素上,并且对边界积分也采用了边界样本池进行采样优化。

### 3.2 高效可微渲染系统、框架与语言

渲染是需要高性能计算的领域,经过多年发展,已经有一套成熟的渲染管线和多种专用框架和语言用于正向渲染,而可微渲染需要对这些管线、框架和语言进行扩展以支持梯度计算,如何保持高效性是一个值得研究的问题。

系统层面,Mitsuba 2(Nimier-David 等,2019)和 Mitsuba 3 是两个开源可微渲染器系统,使用基于阵列(array)的编程模型,并支持了一些可微渲染算法。其中,Mitsuba 2 的底层计算库为支持自动微分功能的开源框架 Enoki; Mitsuba 3 则基于 Enoki 的后继 Dr. JIT (Jakob 等,2022)实现,将对于阵列的正向和反向计算过程分别融合,并即时编译为不同计算后端的 kernel,以降低全局内存读写并提高计算效率。Zheng 等人(2022)开发了一个跨平台高性能渲染系统 LuisaRender,通过结合嵌入于 C++ 等语言的 DSL 和即时编译技术,提供了多阶段 kernel 编程能力,且初步支持对 SSA IR 的自动微分程序变换,以简化可微渲染实现。

框架层面,如今的很多梯度计算框架往往需要保存过程中的计算图,而渲染涉及到的计算图非常复杂,如果要将每一步运算都记录下来,对内存的开销是不可承受的。为了解决这一问题,Loubet 等人(2019)提出了辐射度反向传播算法。此方法将梯度的反向传播类比为正向渲染中的辐射传输过程,辐射从光源出发经过若干次散射折射后进入相机,而梯度也可以理解为从像素出发,在场景中不断在发生散射的同时传播到场景参数上。而且这一方法也将正向渲染和反向传播的过程分开,使得我们可以分别采用不同的采样方法以提高效率。Vicini 等人(2021)进一步在 Mitsuba 3 的基础上,通过复用光路的方式降低了之前方法的时间复杂度并保证

了无偏性,同时还讨论了如何处理反向传播时由于场景参数变化导致的光路几何变化对梯度的影响。

编程语言方面,为避免手动设计复杂的求解梯度的方法,一些研究者在研究如何自动化地求解类似渲染过程中遇到的这种存在不连续的函数的梯度。Bangaru 等人(2021)提出了 Teg 语言,将标准形式的狄拉克函数引入积分中并对其微分进行计算,再将其他类型不连续函数通过变量替换的方式转化为含有标准形式的狄拉克函数以处理更多函数。Yang 等人(2022)则提出  $\Delta\delta$  语言,通过对任意函数中的不连续处进行一个与 1 维 Box 核函数进行卷积来将梯度的狄拉克函数转换为一个常规函数,并证明了该方法的一阶正确性。Bangaru 等人(2023)则提出 Slang.D 语言,使得用户可以容易的将基于 Direct3D 的渲染器转为可微渲染器,并提高了深度学习框架和图形 API 的交互效率。

### 3.3 可微渲染鲁棒性提高方法

由于渲染函数高度非线性,导致基于梯度的优化方法容易陷入局部最小值,这使得可微渲染方法需要一个较为准确的初始解才能收敛到正确结果,限制了可微渲染的应用范围。为了解决这一点,研究者们提出了一些提高可微渲染的鲁棒性和泛用性的方法。

Xing 等人(2022)观察到大多数可微渲染方法的都是要求解光路颜色关于场景参数的微分,以更好的适应基于颜色差异的损失函数如 L1/L2 Loss。在面对初始场景和目标场景差异较大的情况时,基于颜色损失和微分求解出的导数往往是稀疏且限于局部的,对优化不具有指导意义。于是该方法提出了一种基于最优传输的损失函数:将像素的三维颜色和二维位置 XY 组合变为五维,并对渲染图片和目标图片运用最优传输方法为每条光路找到一个目标,包含目标颜色和位置。为了求解光路位置关于场景参数的导数,Xing 等人(2023)通过扩展路径空间流形,使用场景参数对光路几何建立约束,并通过隐函数求导的理论求解出光路几何关于场景参数的梯度。

Fischer 和 Ritschel(2022)将高维度的渲染函数与经过模糊的参数空间进行卷积,一定程度解决了可微渲染过程中容易陷入局部最小值的问题。该方法的优点在于不需要对渲染器进行过多的修改,甚至不要求渲染过程可微,仅依靠卷积过程关于参数空间的导数也可获得较好的效果。

关于场景几何的梯度往往只在几何的边界部分

数值非零，只通过这部分梯度进行优化很容易使得基于网格表达的几何形状变得不规则。于是，Nicolet 等人 (2021) 提出了针对网格模型的一种特殊优化方法，该方法提出将顶点坐标表示为 Cartesian 坐标和微分坐标的加权组合，将针对每个顶点的导数扩散到所有顶点。Palfinger 等人 (2022) 则提出了一种在可微渲染迭代过程中不断对网格进行重网格化 (Remesh) 的方法，两种方法都能使得几何在优化过程中始终维持较好的性质。

## 4 可微渲染的应用

前述章节讨论了可微渲染本身的理论、实践与优化方法。本章将介绍若干基于可微渲染的实际应用。

由于本文主题是基于物理的可微渲染，所以聚焦于主体为可微渲染的工作，对于当前较为热门的基于神经网络的工作，虽然有很多都将基于物理的可微渲染作为其中一个模块，但通常不是其研究重点，所以不在本文中进行讨论。

### 4.1 使用可微渲染进行物体几何与材质重建

Luan 等人 (2021) 提出用可微渲染来优化物体几何和空间可变 BRDF 的方法。该方法首先用 COLMAP 算法对物体进行初始化，然后运用基于边界采样的可微渲染方法优化物体几何。对于材质，该方法采用若干技巧如多尺度优化、加入了一些对于几何和反射的先验知识。Goe1 等人 (2020) 同样采用 COLMAP 进行初始化，通过在优化过程中不断对几何做和几何简化的操作使得几何始终保持一个较好的形态且参数量可控，并证明几何和材质交替优化的策略能起到更好的恢复效果。

Cai 等人 (2022) 中提出了一种结合隐式表达和显式表达的物体几何和材质优化方法，该方法首先将物体用有向距离场 (SDF) 表示，然后通过可微 Marching Cube 算法转为网格表达后用可微渲染器进行渲染和优化，最后优化 SDF 参数。这种方法结合了 SDF 可变拓扑的优点与显式网格表达细节丰富的优点，一定程度上使得可微渲染不再要求一个较为精确的几何作为初始解，从一个简单的球体就能恢复出复杂的几何，效果见图 5。



图 5 物体重建结果 (Cai 等, 2022)

Fig.5 Object reconstruction result of (Cai et., 2022)

Lyu 等人 (2020) 提出了一种用来恢复完全透明的物体的方法。由于物体完全透明，所以呈现出的颜色为背景经过折射后的颜色。所以此方法首先利用环境铺垫 (environment matting) 的方法取得了目标图像中每个像素的光路经过物体折射后，光线最终处于背景图像中的位置，再将初始模型进行渲染得到每个像素对应的光路终点，优化光路终点的目标位置。Gao 等人 (2023) 同样提出了一种优化透明物体几何的方法，该方法不需要预先确定光路在背景中的位置，而是通过用神经网络表示的有向距离场来表示几何。通过投影，可获得物体在每个视角下的轮廓，然后通过基于轮廓和颜色的损失函数来优化几何。

还有一些针对于特定类型物体的重建工作，(Velinov 等, 2018) 通过模拟口腔环境内复杂的光路传输进行关于牙齿的几何和材质的获取；(Levis 等, 2015) 和 (Czerninski 等, 2023) 提出了针对大气中的散射介质如云和气溶胶的散射参数的恢复，后者还将迭代过程中的路径复用以提高效率。

(Nindel 等, 2021) 使用可微渲染进行适用于 3D 打印物体材质的优化，(Wu 等, 2023) 利用可微渲染简化复杂遮挡物的几何。

### 4.2 使用可微渲染进行物体与环境光重建

Hasselgren 等人 (2022) 除了优化物体的几何和材质，还将环境光纳入优化目标，该方法同样采用显式和隐式结合的几何表达，从隐式表达转为显式表达采用的是对物体细节恢复较好的 Deep Marching Tetrahedra (Shen 等, 2021) 方法。另外，该方法在蒙特卡洛光线追踪渲染后还通过可微的方式进行了降噪，以获得更准确的损失函数。

Lyu 等人 (2023) 则在给定物体几何的情况下，优化物体材质与环境光，该方法考虑到通过物体恢复环境光具有模糊性和不确定性，所以提出将环境光优化的过程与基于扩散模型的图像生成算法结合，利用对于环境光的梯度指导扩散模型，实现高质量环境光的生成。

### 4.3 使用可微渲染进行场景重建

对于场景级别的重建，Azinovic 等人 (2019) 和 Nimier-David 等人 (2021) 都尝试在给定室内物体几何的情况下恢复场景中物体的材质和光照。其中 (Nimier-David 等, 2021) 还进行了多种优化：

采用了材质空间的采样方法降低了由于视角问题造成的场景参数优化程度不均的问题;使用掩膜(mask)来避免 Adam 优化器放大优化过程中的噪声问题;结合场景物体识别以减轻光照与材质混淆的问题。使用该方法的重建效果见图 6, 该方法重建后的场景可进行材质编辑、重光照等操作。



(a) 参考图 (b) 优化结果  
(a) Reference (b) Optimized

图 6 场景重建结果 (Nimier-David 等, 2021)

Fig.6 Scene reconstruction result of (Nimier-David et., 2022)

Yan 等人 (2023) 提出一种从单张图片恢复场景几何、材质和光照的方法, 该方法首先利用场景理解方法获得图片语义信息, 并通过检索模型数据库得到对场景几何的较好的初始化, 然后使用可微渲染进一步优化几何、材质、光照。

Xing 等人 (2022, 2023) 利用其提出的扩展路径空间流形和基于最优传输的损失函数, 可优化含有多物体、且初始参数和目标有大尺度差异的场景, 可以完成室内家具布局、焦散与阴影图案设计等复杂优化任务。

## 5 可微渲染的发展趋势与展望

基于物理的可微渲染的发展使得我们可以将图形学中发展多年的关于虚拟物体的几何表达、材质模型用于对现实世界的重建中, 获得对现实世界的越来越准确的数字化模型。在各行各业中的数字化技术都愈发重要的今天, 拥有数字化模型能在很多方面优化设计、管理能力和效率。所以, 可微渲染有着广阔的应用前景。

当然, 作为一种新型工具, 可微渲染还有很多尚未完全解决的问题。本文将这些问题归为三类, 对每一类问题进行分析并介绍现在的解决方案和今后可能的发展趋势, 期望对未来的研究提供参考。

### 5.1 可微渲染的效率

基于物理的可微渲染目前的效率还比较低, 主要有三个方面的原因, 第一, 和正向渲染相同, 目前可微渲染还是主要依托于模拟光路传输进行梯度估

计, 需要采样大量的光路才能获得较为准确的梯度。第二, 可微渲染涉及到边界积分项, 该项不仅要引入额外的计算, 且这种计算与传统渲染流水线的差异较大, 缺乏硬件支持的高效率实现。第三, 可微渲染需要记录一系列渲染的中间状态以方便梯度的反向传播, 会引入额外的时间和空间开销。

为了解决这些问题, 算法层面, 我们需要设计更高效的可微渲染算法, 目前已经有一些在采样和迭代间信息复用方面的研究, 但还有很多点可以进行探究, 如针对可微渲染设计合适的降噪算法、提高采样到梯度较大的路径的概率等。

框架和系统层面, 我们需要结合可微渲染的特点设计更高效的渲染管线, 一方面要利用当前硬件的各种新特性, 另一方面也要让用户容易使用, 这方面, Dr. JIT (Jakob 等, 2022) 是一个较为全面的系统, 但其效率还有提升空间, LuisaRender (Zheng 等, 2022) 则需完善对可微渲染的支持。

### 5.2 可微渲染的鲁棒性

由于渲染函数的高度复杂性和非线性, 通过梯度下降的方式优化场景参数可能会面临陷入局部最小值的问题。另外, 由于可微渲染中涉及到的优化参数通常都具有实际的物理意义, 不同类型参数的梯度可能具有不同的性质, 用同样的方法优化所有参数有可能使优化过程不够稳定。

对于局部最小值的问题, 一个值得研究的方向是定义更为全面的损失函数, 而不仅限于渲染图和目标图之间逐像素的颜色差异, Xing 等人 (2023) 的方法是在这个方面的一种探索, 但其算法也存在准确度、效率不高的问题, 引入基于图像语义定义的损失函数是值得考虑的改进方向。

对于优化方法问题, 目前已经有一些工作提出了对特定参数的优化方法, 如 (Nicolet 等, 2021) 提出对几何参数采用更为平滑的优化方式, 但还没有一个能自适应各类型参数的方法。另外, 不同参数所对应的梯度值域和分布差异可能较大, 如何自适应地利用这些梯度也值得研究。

### 5.3 可微渲染与其他方法的结合

可微渲染擅长对复杂的光路传输现象进行梯度的计算, 但由于缺乏对图片的先验知识, 导致在初始化不佳、或者从零开始的重建问题中, 单独应用可微渲染, 较难获得较好效果。

将可微渲染与其他方法结合是一个直观且有效的想法, 实际应用中大部分方法也都是这样做的。一些应用先用经典的方法获得一个初始化结果后再

用可微渲染进行优化,一些方法采用神经网络生成部分场景参数,充分利用了网络更强的表达能力或更多的先验知识。

在此基础上,本文认为可微渲染还可以更进一步的与其他方法相结合,比如对于渲染效率问题,可以考虑通过某些方式缓存全局光照,如NRC(Neural Radiance Cache,神经辐射都缓存)(Muller等,2021)或某些基于神经辐射场的方法;对于场景表达问题,可以考虑用最近较为流行的3D Gaussian Splatting(三维高斯泼溅)作为几何表达,兼顾灵活性与效率。

### 参考文献(References)

Azinovic D, Li T-M, Kaplanyan A and Nießner M. 2019. Inverse Path Tracing for Joint Material and Lighting Estimation. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 2442–2451). Presented at the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). [DOI:10.1109/CVPR.2019.00255]

Balint M, Myszkowski K, Seidel H-P and Singh G. 2023. Joint Sampling and Optimisation for Inverse Rendering. In SIGGRAPH Asia 2023 Conference Papers (pp. 1–10). New York, NY, USA: Association for Computing Machinery. [DOI:10.1145/3610548.3618244]

Bangaru S P, Gharbi M, Luan F, Li T-M, Sunkavalli K, Hasan M, ... Durand F. 2022. Differentiable Rendering of Neural SDFs through Reparameterization. In SIGGRAPH Asia 2022 Conference Papers (pp. 1–9). New York, NY, USA: Association for Computing Machinery. [DOI:10.1145/3550469.3555397]

Bangaru S P, Li T-M and Durand F. 2020. Unbiased warped-area sampling for differentiable rendering. *ACM Transactions on Graphics*, 39(6), 1–18. [DOI:10.1145/3414685.3417833]

Bangaru S P, Michel J, Mu K, Bernstein G, Li T-M and Ragan-Kelley J. 2021. Systematically differentiating parametric discontinuities. *ACM Transactions on Graphics*, 40(4), 1–18. [DOI:10.1145/3476576.3476671]

Bangaru S P, Wu L, Li T-M, Munkberg J, Bernstein G, Ragan-Kelley J, ... He Y. 2023. SLANG.D: Fast, Modular and Differentiable Shader Programming. *ACM*

*Transactions on Graphics*, 42(6), 1–28. [DOI:10.1145/3618353]

Cai G, Yan K, Dong Z, Gkioulekas I and Zhao S. 2022. Physics-Based Inverse Rendering using Combined Implicit and Explicit Geometries. *Computer Graphics Forum*, 41(4), 129–138. [DOI:10.1111/cgf.14592]

Chang W, Sivaram V, Nowrouzezahrai D, Hachisuka T, Ramamoorthi R and Li T-M. 2023. Parameter-space ReSTIR for Differentiable and Inverse Rendering. In *ACM SIGGRAPH 2023 Conference Proceedings* (pp. 1–10). New York, NY, USA: Association for Computing Machinery. [DOI:10.1145/3588432.3591512]

Czerninski I and Schechner Y Y. 2023. PARS - Path recycling and sorting for efficient cloud tomography. *Intelligent Computing*, 2, 0007. [DOI:10.34133/icomputing.0007]

Deng X, Luan F, Walter B, Bala K and Marschner S. 2022. Reconstructing Translucent Objects using Differentiable Rendering. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings* (pp. 1–10). Presented at the SIGGRAPH '22: Special Interest Group on Computer Graphics and Interactive Techniques Conference, Vancouver BC Canada: ACM. [DOI:10.1145/3528233.3530714]

Fischer M and Ritschel T. 2023. Plateau-Reduced Differentiable Path Tracing (pp. 4285–4294). Presented at the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society. [DOI:10.1109/CVPR52729.2023.00417]

Gao F, Zhang L, Wang L, Cheng J and Zhang J. 2023. Transparent Object Reconstruction via Implicit Differentiable Refraction Rendering. In *SIGGRAPH Asia 2023 Conference Papers* (pp. 1–11). Presented at the SA '23: SIGGRAPH Asia 2023, Sydney NSW Australia: ACM. [DOI:10.1145/3610548.3618236]

Gkioulekas I, Levin A and Zickler T. 2016. An Evaluation of Computational Imaging Techniques for Heterogeneous Inverse Scattering. In B. Leibe, J. Matas, N. Sebe & M. Welling (Eds.), *Computer Vision – ECCV 2016* (Vol. 9907, pp. 685–701). Cham: Springer International Publishing. [DOI:10.1007/978-3-319-46487-9\_42]

Gkioulekas I, Zhao S, Bala K, Zickler T and Levin A.

2013. Inverse volume rendering with material dictionaries. *ACM Transactions on Graphics*, 32(6), 1–13. [DOI:10.1145/2508363.2508377]
- Goel P, Cohen L, Guesman J, Thamizharasan V, Tompkin J and Ritchie D. 2020. Shape from Tracing: Towards Reconstructing 3D Object Geometry and SVBRDF Material from Images via Differentiable Path Tracing (pp. 1186–1195). Presented at the 2020 International Conference on 3D Vision (3DV), IEEE Computer Society. [DOI:10.1109/3DV50981.2020.00129]
- Hasselgren J, Hofmann N and Munkberg J. 2022. Shape, Light, and Material Decomposition from Images using Monte Carlo Rendering and Denoising. *Advances in Neural Information Processing Systems*, 35, 22856–22869.
- Jakob W, Speierer S, Roussel N and Vicini D. 2022. DR.JIT: a just-in-time compiler for differentiable rendering. *ACM Transactions on Graphics*, 41(4), 1–19. [DOI:10.1145/3528223.3530099]
- Kajiya J T. 1986. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques* (pp. 143–150). New York, NY, USA: Association for Computing Machinery. [DOI:10.1145/15922.15902]
- Khungurn P, Schroeder D, Zhao S, Bala K and Marschner S. 2016. Matching Real Fabrics with Micro-Appearance Models. *ACM Transactions on Graphics*, 35(1), 1:1-1:26. [DOI:10.1145/2818648]
- Levis A, Schechner Y Y, Aides A and Davis A B. 2015. Airborne Three-Dimensional Cloud Tomography. In *2015 IEEE International Conference on Computer Vision (ICCV)* (pp. 3379–3387). Presented at the 2015 IEEE International Conference on Computer Vision (ICCV). [DOI:10.1109/ICCV.2015.386]
- Li G, Meka A, Mueller F, Buehler M C, Hilliges O and Beeler T. 2022. EyeNeRF: a hybrid representation for photorealistic synthesis, animation and relighting of human eyes. *ACM Transactions on Graphics*, 41(4), 1–16. [DOI:10.1145/3528223.3530130]
- Li T-M, Aittala M, Durand F and Lehtinen J. 2019. Differentiable Monte Carlo ray tracing through edge sampling. *ACM Transactions on Graphics*, 37(6), 1–11. [DOI:10.1145/3272127.3275109]
- Loubet G, Holzschuch N and Jakob W. 2019. Reparameterizing discontinuous integrands for differentiable rendering. *ACM Transactions on Graphics*, 38(6), 1–14. [DOI:10.1145/3355089.3356510]
- Luan F, Zhao S, Bala K and Dong Z. 2021. Unified Shape and SVBRDF Recovery using Differentiable Monte Carlo Rendering. *Computer Graphics Forum*, 40(4), 101–113. [DOI:10.1111/cgf.14344]
- Lyu J, Wu B, Lischinski D, Cohen-Or D and Huang H. 2020. Differentiable refraction-tracing for mesh reconstruction of transparent objects. *ACM Transactions on Graphics*, 39(6), 195:1-195:13. [DOI:10.1145/3414685.3417815]
- Lyu L, Tewari A, Habermann M, Saito S, Zollhöfer M, Leimkühler T and Theobalt C. 2023. Diffusion Posterior Illumination for Ambiguity-Aware Inverse Rendering. *ACM Transactions on Graphics*, 42(6), 233:1-233:14. [DOI:10.1145/3618357]
- Müller T, Rousselle F, Novák J and Keller A. 2021. Real-time neural radiance caching for path tracing. *ACM Transactions on Graphics*, 40(4), 1–16. [DOI:10.1145/3450626.3459812]
- Nicolet B, Jacobson A and Jakob W. 2021. Large steps in inverse rendering of geometry. *ACM Transactions on Graphics*, 40(6), 1–13. [DOI:10.1145/3478513.3480501]
- Nicolet B, Rousselle F, Novak J, Keller A, Jakob W and Müller T. 2023. Recursive Control Variates for Inverse Rendering. *ACM Transactions on Graphics*, 42(4), 1–13. [DOI:10.1145/3592139]
- Nimier-David M, Dong Z, Jakob W and Kaplanyan A. 2021. Material and Lighting Reconstruction for Complex Indoor Scenes with Texture-space Differentiable Rendering. *The Eurographics Association*. [DOI:10.2312/sr.20211292]
- Nimier-David M, Müller T, Keller A and Jakob W. 2022. Unbiased inverse volume rendering with differential trackers. *ACM Transactions on Graphics*, 41(4), 44:1-44:20. [DOI:10.1145/3528223.3530073]
- Nimier-David M, Speierer S, Ruiz B and Jakob W. 2020. Radiative backpropagation: an adjoint method for lightning-fast differentiable rendering. *ACM Transactions on Graphics*, 39(4),

[DOI:10.1145/3386569.3392406]

Nimier-David M, Vicini D, Zeltner T and Jakob W. 2019. Mitsuba 2: a retargetable forward and inverse renderer. *ACM Transactions on Graphics*, 38(6), 1–17.

[DOI:10.1145/3355089.3356498]

Nindel T K, Iser T, Rittig T, Wilkie A and Krivánek J. 2021. A gradient-based framework for 3D print appearance optimization. *ACM Transactions on Graphics*, 40(4), 178:1-178:15.

[DOI:10.1145/3450626.3459844]

Palfinger W. 2022. Continuous remeshing for inverse rendering. *Computer Animation and Virtual Worlds*, 33(5), e2101. [DOI:10.1002/cav.2101]

Shem-Tov K, Bangaru S P, Levin A and Gkioulekas I. 2020. Towards Reflectometry from Interreflections. In 2020 IEEE International Conference on Computational Photography (ICCP) (pp. 1–12). Presented at the 2020 IEEE International Conference on Computational Photography (ICCP), Saint Louis, MO, USA: IEEE. [DOI:10.1109/ICCP48838.2020.9105251]

Shen T, Gao J, Yin K, Liu M-Y and Fidler S. 2021. Deep Marching Tetrahedra: a Hybrid Representation for High-Resolution 3D Shape Synthesis. In *Advances in Neural Information Processing Systems* (Vol. 34, pp. 6087–6101).

Teh A, O’Toole M and Gkioulekas I. 2022. Adjoint nonlinear ray tracing. *ACM Transactions on Graphics*, 41(4), 126:1-126:13. [DOI:10.1145/3528223.3530077]

Tong X, Liu H-T D, Gingold Y and Jacobson A. 2023. Differentiable Heightfield Path Tracing with Accelerated Discontinuities. In *ACM SIGGRAPH 2023 Conference Proceedings* (pp. 1–9). New York, NY, USA: Association for Computing Machinery. [DOI:10.1145/3588432.3591530]

Velinov Z, Papas M, Bradley D, Gotardo P, Mirdehghan P, Marschner S, ... Beeler T. 2018. Appearance capture and modeling of human teeth. *ACM Transactions on Graphics*, 37(6), 207:1-207:13. [DOI:10.1145/3272127.3275098]

Vicini D, Speierer S and Jakob W. 2021. Path replay backpropagation: differentiating light paths using constant memory and linear time. *ACM Transactions on Graphics*, 40(4), 1–14. [DOI:10.1145/3450626.3459804]

Vicini D, Speierer S and Jakob W. 2022. Differentiable signed distance function rendering. *ACM Transactions on Graphics*, 41(4), 1–18.

[DOI:10.1145/3528223.3530139]

Wang Y-C, Wyman C, Wu L and Zhao S. 2023. Amortizing Samples in Physics-Based Inverse Rendering Using ReSTIR. *ACM Transactions on Graphics*, 42(6), 1–17. [DOI:10.1145/3618331]

Worchel M and Alexa M. 2023. Differentiable Rendering of Parametric Geometry. *ACM Transactions on Graphics*, 42(6), 232:1-232:18. [DOI:10.1145/3618387]

Wu J, Lin Y and Lu D. 2023. DR-Occluder: Generating Occluders Using Differentiable Rendering. *ACM Transactions on Graphics*, 42(6), 1–14. [DOI:10.1145/3618346]

Wu L, Cai G, Ramamoorthi R and Zhao S. 2021. Differentiable time-gated rendering. *ACM Transactions on Graphics*, 40(6), 1–16. [DOI:10.1145/3478513.3480489]

Xing J, Hu X, Luan F, Yan L-Q and Xu K. 2023. Extended Path Space Manifolds for Physically Based Differentiable Rendering. In *SIGGRAPH Asia 2023 Conference Papers* (pp. 1–11). Presented at the SA ’23: SIGGRAPH Asia 2023, Sydney NSW Australia: ACM. [DOI:10.1145/3610548.3618195]

Xing J, Luan F, Yan L-Q, Hu X, Qian H and Xu K. 2022. Differentiable Rendering Using RGBXY Derivatives and Optimal Transport. *ACM Transactions on Graphics*, 41(6), 189:1-189:13. [DOI:10.1145/3550454.3555479]

Xu P, Bangaru S, Li T-M and Zhao S. 2023. Warped-Area Reparameterization of Differential Path Integrals. *ACM Transactions on Graphics*, 42(6), 1–18. [DOI:10.1145/3618330]

Yan K, Lassner C, Budge B, Dong Z and Zhao S. 2022. Efficient estimation of boundary integrals for path-space differentiable rendering. *ACM Transactions on Graphics*, 41(4), 1–13. [DOI:10.1145/3528223.3530080]

Yan K, Luan F, Hašan M, Groueix T, Deschaintre V and Zhao S. 2023. PSDR-Room: Single Photo to Scene using Differentiable Rendering. In *SIGGRAPH Asia 2023 Conference Papers* (pp. 1–11). Presented at the SA ’23: SIGGRAPH Asia 2023, Sydney NSW Australia:

ACM. [DOI:10.1145/3610548.3618165]

Yang Y, Barnes C, Adams A and Finkelstein A. 2022. A $\delta$ : autodiff for discontinuous programs - applied to shaders. ACM Transactions on Graphics, 41(4), 135:1-135:24. [DOI:10.1145/3528223.3530125]

Yi S, Kim D, Choi K, Jarabo A, Gutierrez D and Kim M H. 2021. Differentiable transient rendering. ACM Transactions on Graphics, 40(6), 286:1-286:11. [DOI:10.1145/3478513.3480498]

Yu Z, Zhang C, Maury O, Hery C, Dong Z and Zhao S. 2023. Efficient Path-Space Differentiable Volume Rendering With Respect To Shapes. [DOI:10.1111/cgf.14884]

Yu Z, Zhang C, Nowrouzezahrai D, Dong Z and Zhao S. 2022. Efficient Differentiation of Pixel Reconstruction Filters for Path-Space Differentiable Rendering. ACM Transactions on Graphics, 41(6), 1-16. [DOI:10.1145/3550454.3555500]

Zeltner T, Speierer S, Georgiev I and Jakob W. 2021. Monte Carlo estimators for differential light transport. ACM Transactions on Graphics, 40(4), 1-16. [DOI:10.1145/3476576.3476633]

Zhang C, Dong Z, Doggett M and Zhao S. 2021. Antithetic sampling for Monte Carlo differentiable rendering. ACM Transactions on Graphics, 40(4), 1-12. [DOI:10.1145/3450626.3459783]

Zhang C, Miller B, Yan K, Gkioulekas I and Zhao S. 2020. Path-space differentiable rendering. ACM Transactions on Graphics, 39(4). [DOI:10.1145/3386569.3392383]

Zhang C, Wu L, Zheng C, Gkioulekas I, Ramamoorthi R and Zhao S. 2019. A differential theory of radiative transfer. ACM Transactions on Graphics, 38(6), 1-16. [DOI:10.1145/3355089.3356522]

Zhang C, Yu Z and Zhao S. 2021. Path-space differentiable rendering of participating media. ACM Transactions on Graphics, 40(4), 1-15. [DOI:10.1145/3476576.3476631]

Zhang Z, Roussel N and Jakob W. 2023. Projective Sampling for Differentiable Rendering of Geometry. ACM Transactions on Graphics, 42(6), 212:1-212:14. [DOI:10.1145/3618385]

Zhao S, Wu L, Durand F and Ramamoorthi R. 2016. Downsampling scattering parameters for rendering

anisotropic media. ACM Transactions on Graphics, 35(6), 166:1-166:11. [DOI:10.1145/2980179.2980228]

Zheng S, Zhou Z, Chen X, Yan D, Zhang C, Geng Y, ... Xu K. 2022. LuisaRender: A High-Performance Rendering Framework with Layered and Unified Interfaces on Stream Architectures. ACM Transactions on Graphics, 41(6), 232:1-232:19. [DOI:10.1145/3550454.3555463]

Zhou Y, Wu L, Ramamoorthi R and Yan L-Q. 2021. Vectorization for Fast, Analytic, and Differentiable Visibility. ACM Transactions on Graphics, 40(3), 1-21. [DOI:10.1145/3452097]

Ye Z P, Xia W Y, Sun Z Y, Yi R, Yu M J and Liu Y J. 2021. From traditional rendering to differentiable rendering: theories, methods and applications (in Chinese). SciSinInform, 2021(51): 1043-1067. (叶子鹏, 夏雯宇, 孙志尧, 易冉, 余旻婧, 刘永进. 2021. 从传统渲染到可微渲染: 基本原理、方法和应用. 中国科学: 信息科学, 2021(51): 1043-1067), [DOI: 10.1360/SSI-2020-0272]

Xu W W, Zhou Y, Wu H Z, Guo J. 2021. Differential rendering: a survey. Journal of Image and Graphics, 26(6): 1521-1535. (许威威, 周漾, 吴鸿智, 过洁. 2021. 可微绘制技术研究进展. 中国图象图形学报, 26(6): 1521-1535.) [DOI: 10.11834/jig.200853]

#### 作者简介 第一作者



邢健开, 1998年生, 男, 博士研究生, 研究方向为计算机图形学  
Email:  
xjk21@mails.tsinghua.edu.cn

#### 通讯作者



徐昆, 1985年生, 男, 副教授, 研究方向为计算机图形学  
Email:  
xukun@tsinghua.edu.cn