Differentiable Time-Gated Rendering

LIFAN WU*, NVIDIA, USA
GUANGYAN CAI*, University of California, Irvine, USA
RAVI RAMAMOORTHI, University of California, San Diego, USA
SHUANG ZHAO, University of California, Irvine, USA

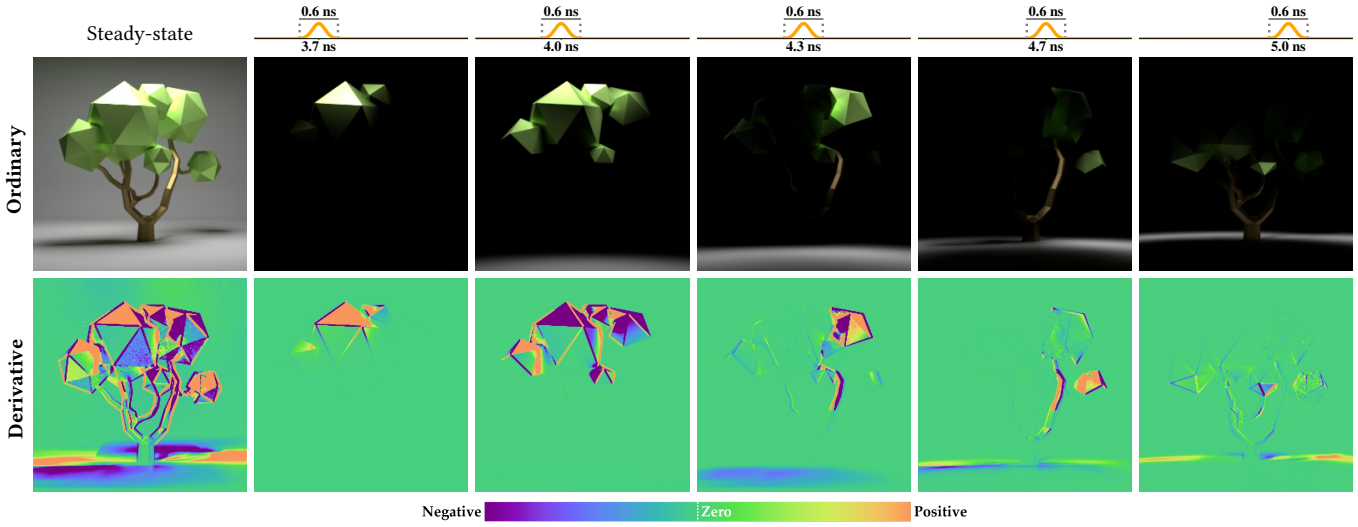

Fig. 1. We introduce a new theory for computing derivatives of time-gated renderings. Our theory captures high-order light transport effects such as soft shadows and interreflection, and offers the generality of differentiating with respect to arbitrary scene parameters such as material optical properties and object geometries. This example consists of a glossy tree model [Telezhkin 2021] that is lit by an area light and casts soft shadows on the ground. We show steady-state and time-gated renderings with varying *path-length importance functions* (that encode camera time gates). The corresponding derivative images with respect to the tree's rotation about the vertical axis are visualized at the bottom. The color encoding for derivatives is also used in the following figures.

The continued advancements of time-of-flight imaging devices have enabled new imaging pipelines with numerous applications. Consequently, several forward rendering techniques capable of accurately and efficiently simulating these devices have been introduced. However, general-purpose differentiable rendering techniques that estimate derivatives of time-of-flight images are still lacking. In this paper, we introduce a new theory of differentiable time-gated rendering that enjoys the generality of differentiating with respect to arbitrary scene parameters. Our theory also allows the design of advanced Monte Carlo estimators capable of handling cameras with near-delta or discontinuous time gates.

We validate our theory by comparing derivatives generated with our technique and finite differences. Further, we demonstrate the usefulness of our technique using a few proof-of-concept inverse-rendering examples that simulate several time-of-flight imaging scenarios.

CCS Concepts: • **Computing methodologies** → **Rendering**.

ACM Reference Format:
Lifan Wu, Guangyan Cai, Ravi Ramamoorthi, and Shuang Zhao. 2021. Differentiable Time-Gated Rendering. *ACM Trans. Graph.* 40, 6, Article 287 (December 2021), 16 pages. https://doi.org/10.1145/3478513.3480489

1 INTRODUCTION

Recently, time-of-flight (ToF) imaging has revolutionized a wide range of applications in robotics, autonomous navigation, atmospheric sciences, medicine, and entertainment. Unlike conventional intensity sensors that only record how many photons are received, the ToF sensors also retain the information about when a photon arrives (by, for example, weighting each photon differently based on its time of arrival). With such information, ToF imaging techniques are capable of recovering geometric and material properties of a scene more effectively than conventional methods, even when parts of the scene are not directly visible to the sensor.

Because of the usefulness of time-of-flight sensors, *time-gated rendering* techniques have been developed to simulate them in computer graphics. These techniques utilize a *path-length importance* function to model how ToF sensors respond to photons with varying times of arrival. Early time-gated rendering methods have

*Both authors contributed equally to the paper.

Authors' addresses: Lifan Wu, lifanw@nvidia.com, NVIDIA, USA; Guangyan Cai, gcai3@uci.edu, University of California, Irvine, USA; Ravi Ramamoorthi, ravir@cs.ucsd.edu, University of California, San Diego, USA; Shuang Zhao, shz@ics.uci.edu, University of California, Irvine, USA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2021 Association for Computing Machinery.
0730-0301/2021/12-ART287 $15.00
https://doi.org/10.1145/3478513.3480489

adopted steady-state rendering approaches that neglect lengths of light transport paths during the path sampling process. Although these methods work adequately when the path-length importance is smooth, they become inefficient for path-length importance functions with narrow supports. To address this problem, more advanced techniques—which are tailored to time-gated rendering—have been introduced [Jarabo et al. 2014; Pediredla et al. 2019b]. These techniques consider lengths of light paths when sampling them, leading to significantly better performance when simulating ToF sensors that accumulate contributions only from small subsets of photons.

Differentiable rendering is concerned with numerically estimating derivatives of images with respect to differential changes of a scene. These techniques have a wide array of applications such as enabling gradient-based optimization for solving inverse-rendering problems. Recently, great progress has been achieved in differentiable rendering theory, algorithms, and systems [Li et al. 2018; Loubet et al. 2019; Nimier-David et al. 2019; Zhang et al. 2019; Nimier-David et al. 2020; Zhang et al. 2020; Bangaru et al. 2020; Zhang et al. 2021b; Zeltner et al. 2021; Vicini et al. 2021]. Unfortunately, all these techniques are developed for the steady-state configuration only.

In this paper, we bridge this gap by introducing the first general theory on differentiable time-gated rendering. Our theory offers the generality to (i) handle complex light transport phenomena (such as soft shadows and interreflections); and (ii) differentiate with respect to arbitrary scene parameters such as object geometry and material optical properties.

Concretely, our contributions include:

- *Establishing* the theoretical formulation of *differential time-gated path integrals* (§4). This formulation introduces a new path-length-*boundary* component unique to differentiable time-gated rendering (§4.3).
- Developing new Monte Carlo methods to estimate these integrals in an efficient and unbiased fashion (§5). Specifically, we show how ellipsoidal change of variable [Pediredla et al. 2019b] and antithetic sampling [Zhang et al. 2021a] can be used to efficiently handle cameras with near-delta and discontinuous time gates.

We validate our derivative estimates by comparing with finite-difference references (Figure 8). Further, to demonstrate the usefulness of our theory and algorithms, we use a few simple proof-of-concept inverse-rendering examples that mimic several time-of-flight imaging scenarios (Figures 10–17).

2 RELATED WORK

Time-gated imaging. Time-gated sensors[1] weight the contributions of individual photons according to their time-of-travel and find applications in many areas. These sensors were first used in imaging applications for looking around corners [Kirmani et al. 2009, 2011]. SPAD sensors are used in biology and medical settings for fluorescence lifetime imaging [Lakowicz et al. 1992], and optical sectioning of layers within scattering media [Das et al. 1993]. Time-gated sensors are also used to enhance visibility by, for instance, imaging through fog and smoke [Satat et al. 2018]. In advanced driver assistance systems (ADAS), time-gated cameras are used to improve safety by avoiding collision and improving night-time navigation [David et al. 2006; Grauer and Sonn 2015]. Coupled with machine learning algorithms, time-gated sensors are also used as alternatives to lidar for dense depth sensing [Gruber et al. 2019]. Lastly, time-gated measurements have also been used for non-line-of-sight (NLOS) imaging [Laurenzis and Velten 2014; Thrampoulidis et al. 2018; Pediredla et al. 2017, 2019a; Saunders et al. 2019; Liu et al. 2019, 2020; Lindell et al. 2019; Raskar et al. 2020; Young et al. 2020].

In computer graphics, time-gated imaging has also attracted increased interest for applications including non-line-of-sight imaging [Heide et al. 2014; Tsai et al. 2017; Jarabo et al. 2017; O'Toole et al. 2018; Heide et al. 2019; Xin et al. 2019; Iseringhausen and Hullin 2020; Chen et al. 2020], separating light transport components [O'Toole et al. 2014], and inverting scattering [Gkioulekas et al. 2016].

Time-gated forward rendering. The theoretical foundation of time-gated rendering is provided by the time-resolved versions of the radiative transfer equation [Chandrasekhar 1960], rendering equation [Kajiya 1986; Smith et al. 2008], and path-integral formulation of light transport [Veach 1997; Pauly et al. 2000; Jarabo et al. 2014]. Most prior research on time-gated rendering has adapted steady-state Monte Carlo rendering algorithms for forward simulations [Jarabo 2012; Jarabo et al. 2014; Jarabo and Arellano 2018]. Most of these works directly repurpose conventional steady-state algorithms such as unidirectional and bidirectional path tracing to sample light transport paths.

Jarabo et al. [2014] introduced a modified BDPT procedure for light paths in scattering media. Additionally, biased methods that trade physical accuracy for computational efficiency, such as photon mapping [Marco et al. 2019] and instant radiosity [Pan et al. 2019] have been adopted for time-gated rendering.

Pediredla et al. [2019b] introduced a new path sampling technique for time-gated rendering of surfaces, which allows precise control of path lengths. We utilize this technique to develop new Monte Carlo estimators for differentiable time-gated rendering.

Recent works on ToF imaging [Tsai et al. 2019; Iseringhausen and Hullin 2020] used a simplified forward-rendering model that considers time-resolved light transport with up to three reflections. This model can be implemented efficiently on the GPU and enables fast NLOS shape reconstruction in an analysis-by-synthesis fashion with derivatives computed using finite differences. In contrast, our method not only enjoys the generality to handle various light transport effects (not limited to three bounces), but also allows estimating scene derivatives in an unbiased and consistent fashion.

Physics-based differentiable rendering. Recently, great progress has been made in physics-based steady-state differentiable rendering. Li et al. [2018] has introduced Monte Carlo edge sampling—a first general-purpose approach for differentiable rendering of surfaces that allows differentiation with respect to arbitrary scene parameters including object geometries. This framework was later generalized by Zhang et al. [2019] to handle volumetric light transport. To improve the efficiency of differentiable rendering, several techniques that reparameterize the rendering equation have been developed [Loubet et al. 2019; Bangaru et al. 2020]. Further,

[1] In this paper, we neglect the practical differences between "transient" and "time-gated" sensors/imaging and use the term "time-gated" to refer to both.

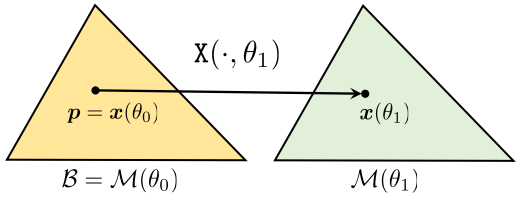

Fig. 2. **Material-form parameterization** of a triangle. In this example, the horizontal translation of this triangle is controlled by a parameter θ. We choose the reference surface $\mathcal{B}$ at some fixed parameter θ_0 (illustrated in yellow). The one-to-one mapping X then specifies the translational motion of the triangle (and hence the name). For each θ, $\mathsf{X}(\cdot,\theta)$ maps each point $\boldsymbol{p}$ on the reference surface $\mathcal{B}$ to one on the actual geometry $\mathcal{M}(\theta)$, which is illustrated in green.

Zhang et al. [2020; 2021b] has introduced the mathematical formulation of differential path integrals, which allows the development of advanced Monte Carlo methods (beyond unidirectional path tracing) for differentiable rendering. All these techniques, however, are developed for steady-state rendering only.

On the other hand, a few specialized time-gated rendering techniques have been developed to solve inverse-rendering problems using time-gated sensors. These techniques, despite being able to compute derivatives, either lack the generality to compute geometric derivatives (that is, those with respect to object geometries) [Naik et al. 2011; Gkioulekas et al. 2016] or the ability to handle complex light transport effects like interreflection [Tsai et al. 2019].

To bridge this gap, we focus on the establishment of a general differential theory of time-gated rendering that offers the generality to differentiate with respect to arbitrary scene parameters (such as material optical properties and object geometries) and the ability to handle a wide range of light transport phenomena including soft shadows and interreflections. While our contribution is primarily theoretical, the results can be used for future development of efficient practical differentiable time-gated renderers.

3 PRELIMINARIES

In this section, we briefly revisit the mathematical preliminaries that we will use to devise our theory and algorithms in §4 and §5. We summarize all commonly used symbols and their definitions in Table 1.

3.1 Forward Time-Gated Rendering

Given a virtual scene consisting of a set of surfaces $\mathcal{M}$, the radiometric response of a time-gated camera can be expressed as the following **time-gated path integral**:

$$I = \int_\Omega W_\tau(\|\bar{\boldsymbol{x}}\|)\, f(\bar{\boldsymbol{x}})\, \mathrm{d}\mu(\bar{\boldsymbol{x}}), \tag{1}$$

where $\bar{\boldsymbol{x}}$ denotes a **light transport path**, and $\Omega \coloneqq \cup_{N=1}^{\infty}\mathcal{M}^{N+1}$ is the **path space** comprised of all finite-length light paths.[2] Further, given a light path $\bar{\boldsymbol{x}} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)$, the **area-product measure** μ is given by

$$\mathrm{d}\mu(\bar{\boldsymbol{x}}) \coloneqq \textstyle\prod_{n=0}^{N} \mathrm{d}A(\boldsymbol{x}_n), \tag{2}$$

with A being the surface-area measure; and the **measurement contribution function** f is defined as:

$$f(\bar{\boldsymbol{x}}) = L_\mathrm{e}(\boldsymbol{x}_0 \to \boldsymbol{x}_1)\, G(\boldsymbol{x}_0 \leftrightarrow \boldsymbol{x}_1) \tag{3}$$
$$\left[\prod_{n=1}^{N-1} f_\mathrm{s}(\boldsymbol{x}_{n-1} \to \boldsymbol{x}_n \to \boldsymbol{x}_{n+1})\, G(\boldsymbol{x}_n \leftrightarrow \boldsymbol{x}_{n+1})\right] W_\mathrm{e}(\boldsymbol{x}_{N-1} \to \boldsymbol{x}_N),$$

where L_e and W_e capture, respectively, **source emission** and **detector importance**; and G is the **geometric term** given by

$$G(\boldsymbol{x} \leftrightarrow \boldsymbol{y}) \coloneqq \mathbb{V}(\boldsymbol{x} \leftrightarrow \boldsymbol{y})\, \frac{|\boldsymbol{n}(\boldsymbol{x}) \cdot \overrightarrow{\boldsymbol{xy}}|\; |\boldsymbol{n}(\boldsymbol{y}) \cdot \overrightarrow{\boldsymbol{yx}}|}{\|\boldsymbol{x}-\boldsymbol{y}\|^2}, \tag{4}$$

with $\mathbb{V}$ being the **mutual visibility function**, $\boldsymbol{n}(\boldsymbol{x})$ indicating the unit surface normal at $\boldsymbol{x}$, and $\overrightarrow{\boldsymbol{xy}}$ denoting the unit vector pointing from $\boldsymbol{x}$ toward $\boldsymbol{y}$. Lastly, unique to time-gated rendering, $W_\tau : \mathbb{R}_{>0} \mapsto \mathbb{R}_{\geq 0}$ is the **path-length importance** with $\|\bar{\boldsymbol{x}}\| \coloneqq \sum_{n=1}^{N}\|\boldsymbol{x}_n - \boldsymbol{x}_{n-1}\|$ denoting the total (optical) length of a light transport path $\bar{\boldsymbol{x}} = (\boldsymbol{x}_0, \ldots, \boldsymbol{x}_N)$.

3.2 Material-Form Parameterization of Virtual Scenes

For a virtual scene controlled by some abstract parameter $\theta \in \mathbb{R}$, when its geometry—namely the object surfaces $\mathcal{M}$—depends on θ, so does the path space Ω. This dependency makes differentiating the time-gated path integral of Eq. (1) with respect to θ more challenging.

To address this problem, Zhang et al. [2020] has introduced a technique that applies a *material-form reparameterization* to the scene geometry $\mathcal{M}(\theta)$. Specifically, let scene geometry $\mathcal{M}(\theta) \subset \mathbb{R}^3$ be parameterized globally via some differentiable mapping $\mathsf{X}(\cdot,\theta)$ that transforms some fixed **reference surface** $\mathcal{B}$ to the actual scene geometry $\mathcal{M}(\theta)$ for all θ (see Figure 2). We will discuss the choice of reference surfaces later in this section.

Given the reference surface $\mathcal{B}$, any point $\boldsymbol{p}$ on this surface is called a **material point**. Further, the path space $\hat{\Omega} \coloneqq \bigcup_{N=1}^{\infty}\mathcal{B}^{N+1}$ induced by reference surface $\mathcal{B}$ is called the **material path space** with its elements $\bar{\boldsymbol{p}} \in \hat{\Omega}$ termed **material light paths**.

The smooth mapping $\mathsf{X}(\cdot,\theta) : \mathcal{B} \mapsto \mathcal{M}(\theta)$ induces a path-level map $\bar{\mathsf{X}}(\cdot,\theta)$ that transforms any material path $\bar{\boldsymbol{p}} = (\boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) \in \hat{\Omega}$ to its ordinary counterpart:

$$\bar{\mathsf{X}}(\bar{\boldsymbol{p}}, \theta) \coloneqq (\mathsf{X}(\boldsymbol{p}_0, \theta), \ldots, \mathsf{X}(\boldsymbol{p}_N, \theta)) \in \Omega(\theta). \tag{5}$$

3.3 Ellipsoidal Change of Variable

We now revisit a change-of-variable technique [Pediredla et al. 2019b] capable of efficiently handling cameras with near-delta path-length importance in forward time-gated rendering.

Assume the reference surface to be represented with polygonal meshes such that $\mathcal{B} = \cup_i \mathcal{T}_i$, where $\mathcal{T}_i$ indicates a polygonal face for each i. Given two material points $\boldsymbol{p}^\mathrm{S}, \boldsymbol{p}^\mathrm{D} \in \mathcal{B}$, each point $\boldsymbol{p} \in \mathcal{T}_i$ can be parameterized using two variables as follows. The first variable $\tau \in \mathbb{R}$ is used to select a subset $\mathcal{E}_i(\tau) \subset \mathcal{T}_i$ that contains all points $\boldsymbol{p}$ satisfying that the sum of distances from $\boldsymbol{p}$ to $\boldsymbol{p}^\mathrm{S}$ and $\boldsymbol{p}^\mathrm{D}$ equals τ. That is, $\mathcal{E}_i(\tau) \coloneqq \{\boldsymbol{p} \in \mathcal{T}_i \;:\; \|\boldsymbol{p}^\mathrm{S} - \boldsymbol{p}\| + \|\boldsymbol{p}^\mathrm{D} - \boldsymbol{p}\| = \tau\}$. It is easy to

[2] We hyperlink keywords to their definitions.

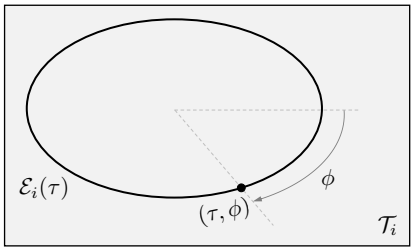

Fig. 3. **Parameterizing** a polygonal face $\mathcal{T}_i \subset \mathcal{B}$ using (i) the sum τ of distances from each material point $\boldsymbol{p} \in \mathcal{T}_i$ to two given points $\boldsymbol{p}^{\mathrm{S}}$ and $\boldsymbol{p}^{\mathrm{D}}$ (which may not belong to $\mathcal{T}_i$); and (ii) another variable ϕ parameterizing $\mathcal{E}_i(\tau)$ that contains all points on the polygon $\mathcal{T}_i$ with the same sum of distances. The set $\mathcal{E}_i(\tau)$ is (part of) an ellipse given by the intersection of an ellipsoid with foci $\boldsymbol{p}^{\mathrm{S}}$ and $\boldsymbol{p}^{\mathrm{D}}$ and the plane in which $\mathcal{T}_i$ resides.

verify that $\mathcal{E}_i(\tau)$ is (part of) an ellipse. The second variable is then used to parameterize the ellipse $\mathcal{E}_i(\tau)$, as illustrated in Figure 3.

This parameterization allows us to rewrite any surface integral over $\mathcal{T}_i$ as:

$$\int_{\mathcal{T}_i} h(\boldsymbol{p})\,\mathrm{d}A(\boldsymbol{p}) = \int_0^\infty \left[\int_{\mathcal{E}_i(\tau)} h(\boldsymbol{p})\, J_i^{\mathrm{e}}(\boldsymbol{p}; \boldsymbol{p}^{\mathrm{S}}, \boldsymbol{p}^{\mathrm{D}})\,\mathrm{d}\ell(\boldsymbol{p})\right]\mathrm{d}\tau, \quad (6)$$

where $h : \mathcal{T}_i \mapsto \mathbb{R}$ is an arbitrary function, ℓ is the curve-length measure, and

$$J_i^{\mathrm{e}}(\boldsymbol{p}; \boldsymbol{p}^{\mathrm{S}}, \boldsymbol{p}^{\mathrm{D}}) \coloneqq \left\|\frac{\mathrm{d}A(\boldsymbol{p})}{\mathrm{d}\ell(\boldsymbol{p})\,\mathrm{d}\tau}\right\| \quad (7)$$

is the corresponding Jacobian determinant. Please refer to the work by Pediredla et al. [2019b] for the exact form of this term.

Based on Eq. (6), we can rewrite integrals over the full reference surface $\mathcal{B}$ as

$$\begin{aligned}\int_{\mathcal{B}} h(\boldsymbol{p})\,\mathrm{d}A(\boldsymbol{p}) &= \int_0^\infty \left[\sum_i \int_{\mathcal{E}_i(\tau)} h(\boldsymbol{p})\, J_i^{\mathrm{e}}(\boldsymbol{p}; \boldsymbol{p}^{\mathrm{S}}, \boldsymbol{p}^{\mathrm{D}})\,\mathrm{d}\ell(\boldsymbol{p})\right]\mathrm{d}\tau \\ &= \int_0^\infty \left[\int_{\mathcal{E}(\tau)} h(\boldsymbol{p})\, J^{\mathrm{e}}(\boldsymbol{p}; \boldsymbol{p}^{\mathrm{S}}, \boldsymbol{p}^{\mathrm{D}})\,\mathrm{d}\ell(\boldsymbol{p})\right]\mathrm{d}\tau, \quad (8)\end{aligned}$$

where the sum is over each polygonal face $\mathcal{T}_i$, and $\mathcal{E}(\tau) \coloneqq \cup_i \mathcal{E}_i(\tau)$. Additionally, $J^{\mathrm{e}}(\boldsymbol{p}; \boldsymbol{p}^{\mathrm{S}}, \boldsymbol{p}^{\mathrm{D}}) \coloneqq J_i^{\mathrm{e}}(\boldsymbol{p}; \boldsymbol{p}^{\mathrm{S}}, \boldsymbol{p}^{\mathrm{D}})$ for all $\boldsymbol{p} \in \mathcal{T}_i$.

In practice, Eq. (8) can be computed efficiently using acceleration structures like BVH to quickly prune polygonal faces that do not intersect with the ellipsoid (determined by foci $\boldsymbol{p}^{\mathrm{S}}$, $\boldsymbol{p}^{\mathrm{D}}$ and sum of distances τ) [Pediredla et al. 2019a].

4 DIFFERENTIABLE TIME-GATED RENDERING THEORY

We now formally establish the mathematical formulation of *differential time-gated path integrals*—a generalization of the steady-state version introduced by Zhang et al. [2020]. Although this formulation has recently been extended to handle volumetric light transport [Zhang et al. 2021b], we focus on the original surface-only variant.

In what follows, we first reparameterize the time-gated path integral of Eq. (1) in its material-form in §4.1. Then, we introduce *differential time-gated path integrals*—our main result of this section—in §4.2 and §4.3.

Table 1. List of symbols commonly used in this paper.

Symbol	Definition
$\bar{\boldsymbol{x}}$	light transport path
$\mathcal{M}(\theta)$	evolving surface
Ω	path space
W_τ	path-length importance
f	measurement contribution
$\bar{\boldsymbol{p}}$	material light path
$\mathcal{B}$	reference configuration
$\hat{\Omega}$	material path space
$\partial\hat{\Omega}$	material boundary path space
$\hat{f}$	material measurement contribution
J_i^{e}	Ellipsoidal change of variable Jacobian

4.1 Material-Form Time-Gated Path Integral

Leveraging the material-form parameterization described in §3.2, applying a change of variable from $\bar{\boldsymbol{x}}$ to $\bar{\boldsymbol{p}}$ to Eq. (1) yields the **material-form time-gated path integral**:

$$\boxed{I = \int_{\hat{\Omega}} W_\tau(\|\bar{\boldsymbol{x}}\|)\,\hat{f}(\bar{\boldsymbol{p}})\,\mathrm{d}\mu(\bar{\boldsymbol{p}}),} \quad (9)$$

where $\bar{\boldsymbol{x}} = \bar{\mathsf{X}}(\bar{\boldsymbol{p}}, \theta)$, and

$$\hat{f}(\bar{\boldsymbol{p}}) \coloneqq f(\bar{\boldsymbol{x}})\, J(\bar{\boldsymbol{p}}), \quad (10)$$

is the **material measurement contribution function** with

$$J(\bar{\boldsymbol{p}}) \coloneqq \left\|\frac{\mathrm{d}\mu(\bar{\boldsymbol{x}})}{\mathrm{d}\mu(\bar{\boldsymbol{p}})}\right\| = \prod_n \left\|\frac{\mathrm{d}A(\boldsymbol{x}_n)}{\mathrm{d}A(\boldsymbol{p}_n)}\right\|, \quad (11)$$

being the Jacobian determinant capturing this change of variable.

Compared with the original time-gated path integral of Eq. (1), the material-form variant of Eq. (9) has an integral domain independent of the scene parameter θ.

Choice of reference. In practice, when rendering the scene at some given $\theta = \theta_0$, it is desired to set the reference surface as the scene geometry at θ_0. That is, $\mathcal{B} = \mathcal{M}(\theta_0)$. In this way, the smooth mapping $\mathsf{X}(\cdot, \theta_0)$ reduces to the identity map, and the material path space $\hat{\Omega}$ coincides with the ordinary path space $\Omega(\theta_0)$. Also, the Jacobian determinant J defined in Eq. (11) has unit value for all $\bar{\boldsymbol{p}}$ at $\theta = \theta_0$. We note that J can still have nonzero derivatives with respect to θ.

Ellipsoidal reparameterization. Let $\hat{\Omega}_{N\geq 2} \coloneqq \cup_{N=2}^\infty \mathcal{B}^{N+1}$ denote the space containing all material light paths with at least two segments (and three vertices). The ellipsoidal change of variable of Eq. (8) allows reparameterizing any material-form path integral via:

$$\begin{aligned}\int_{\hat{\Omega}_{N\geq 2}} F(\bar{\boldsymbol{p}}_+)\,\mathrm{d}\mu(\bar{\boldsymbol{p}}_+) &= \int_{\hat{\Omega}} \left[\int_{\mathcal{B}} F(\bar{\boldsymbol{p}}_+)\,\mathrm{d}A(\boldsymbol{q})\right]\mathrm{d}\mu(\bar{\boldsymbol{p}}) \\ = \int_{\hat{\Omega}} \int_0^\infty \left(\int_{\mathcal{E}(\tau)} F(\bar{\boldsymbol{p}}_+)\, J^{\mathrm{e}}(\boldsymbol{q}; \boldsymbol{p}_0, \boldsymbol{p}_1)\,\mathrm{d}\ell(\boldsymbol{q})\right)\mathrm{d}\tau\,\mathrm{d}\mu(\bar{\boldsymbol{p}}).\end{aligned} \quad (12)$$

where $F : \hat{\Omega} \mapsto \mathbb{R}$ is an arbitrary integrable function over the material path space. Further, given a material light path $\bar{\boldsymbol{p}} = (\boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$, $\bar{\boldsymbol{p}}_+$ is defined as $\bar{\boldsymbol{p}}$ with an additional vertex $\boldsymbol{q} \in \mathcal{B}$ inserted between $\boldsymbol{p}_0$ and $\boldsymbol{p}_1$. That is, $\bar{\boldsymbol{p}}_+ \coloneqq (\boldsymbol{p}_0, \boldsymbol{q}, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$.

4.2 Differential Time-Gated Path Integral

Utilizing the mathematical formulations discussed in §3, we now differentiate the time-gated path integral of Eq. (9) with respect to arbitrary scene parameter θ.

To this end, we make the following four continuity assumptions:

A.1 There are no ideal specular surfaces (e.g., perfect mirrors);

A.2 The source emission L_e, sensor importance W_e, and BSDFs f_s are all differentiable.

A.3 Discontinuities of the Jacobian determinants J of Eq. (11), if they exist, are independent of the parameter θ.

A.4 The path-length importance $W_\tau : \mathbb{R}_{>0} \mapsto \mathbb{R}$ is smooth almost everywhere except at a finite set of jump-discontinuity points which we denote as $\Delta\mathbb{R}[W_\tau]$.

We note that, among all the assumptions above, only **A.4** is new: Assumptions **A.1** and **A.2** are shared by most previous works [Li et al. 2018; Zhang et al. 2019, 2020, 2021b]; Assumption **A.3** follows the steady-state path-space techniques [Zhang et al. 2020, 2021b].

In the supplemental document, we deviate from Assumption **A.4** and discuss the case where the path-length importance W_τ is a Dirac delta function.

Based on the assumptions above, the derivative $\mathrm{d}I/\mathrm{d}\theta$ of Eq. (9) can be expressed as the following **differential time-gated path integral** comprised of an *interior*, a visibility-*boundary*, and a path-length-*boundary* component:

$$\frac{\mathrm{d}I}{\mathrm{d}\theta} = \underbrace{\int_{\hat{\Omega}} \frac{\mathrm{d}}{\mathrm{d}\theta}\left[W_\tau(\|\bar{\boldsymbol{x}}\|)\,\hat{f}(\bar{\boldsymbol{p}})\right]\mathrm{d}\mu(\bar{\boldsymbol{p}})}_{\text{interior}} + \underbrace{\int_{\partial\hat{\Omega}} W_\tau(\|\bar{\boldsymbol{x}}\|)\,\Delta\hat{f}(\bar{\boldsymbol{p}})\,v(\boldsymbol{p}_K)\,\mathrm{d}\dot{\mu}(\bar{\boldsymbol{p}})}_{\text{visibility boundary}} + \underbrace{\sum_{s\in\Delta\mathbb{R}[W_\tau]}\int_{\partial\hat{\Omega}_\tau(\|\bar{\boldsymbol{x}}_+\|)}\Delta W_\tau(s)\,\hat{f}(\bar{\boldsymbol{p}}_+)\,v_\tau(\boldsymbol{q})\,\mathrm{d}\dot{\mu}_\tau(\bar{\boldsymbol{p}}_+)}_{\text{path-length boundary}}. \tag{13}$$

This result is essentially a generalization of the steady-state variant derived by Zhang et al. [2020]. In what follows, we explain each component in more detail.

Interior component. In Eq. (13), the *interior* component is over the same material path space $\hat{\Omega}$ as the ordinary time-gated path integral. Further, the integrand of this component equals:

$$\frac{\mathrm{d}}{\mathrm{d}\theta}\left[W_\tau(\|\bar{\boldsymbol{x}}\|)\,\hat{f}(\bar{\boldsymbol{p}})\right] = \left[\frac{\mathrm{d}}{\mathrm{d}\theta}W_\tau(\|\bar{\boldsymbol{x}}\|)\right]\hat{f}(\bar{\boldsymbol{p}}) + W_\tau(\|\bar{\boldsymbol{x}}\|)\left[\frac{\mathrm{d}}{\mathrm{d}\theta}\hat{f}(\bar{\boldsymbol{p}})\right], \tag{14}$$

where $\hat{f}$ is the material measurement contribution defined in Eq. (10), and μ is the area-product measure. We note that, in Eqs. (13) and (14), all derivatives with respect to θ are **scene derivatives**—a type of material (or total) derivative that takes into consideration dependencies of function parameters on θ. Additionally, when calculating $\mathrm{d}W_\tau(\|\bar{\boldsymbol{x}}\|)/\mathrm{d}\theta$ in Eq. (14), we neglect the (zero-measure) jump discontinuities $\Delta\mathbb{R}[W_\tau]$ of W_τ since they will be later handled by the path-length-*boundary* component.

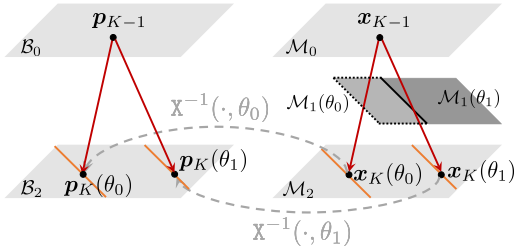

Fig. 4. **Evolution of visibility boundaries:** This example shows a simple scene where a surface $\mathcal{M}_1$ is controlled by a parameter θ (while $\mathcal{M}_0$ and $\mathcal{M}_2$ stay constant). (We omitted the reference $\mathcal{B}_1$ of $\mathcal{M}_1$ for a cleaner illustration.) Under this configuration, for any fixed $\boldsymbol{x}_{K-1} \in \mathcal{M}_0$, the discontinuity points (illustrated as orange lines) of the mutual visibility function $\mathbb{V}(\boldsymbol{x}_{K-1} \leftrightarrow \boldsymbol{x}_K)$ (with respect to $\boldsymbol{x}_K$) vary with θ. This makes $\boldsymbol{p}_K = \mathsf{X}^{-1}(\boldsymbol{x}_K, \theta)$ to also depend on the parameter θ, and $v(\boldsymbol{p}_K)$ in Eq. (13) is given by the scalar change rate of $\boldsymbol{p}_K$ with respect to θ along the normal direction of the discontinuity curve on $\mathcal{B}_2$.

Visibility-boundary component. In Eq. (13), the visibility-*boundary* integral is identical to the *boundary* term of the steady-state result except for the inclusion of the path-length importance W_τ. In the following, we describe the definitions of the key terms in this term for completeness.

The domain of integration $\partial\hat{\Omega}$ is the **material boundary path space** consisting of **material boundary paths** $\bar{\boldsymbol{p}} = (\boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$. These paths are identical to the ordinary ones from $\hat{\Omega}$ except for containing exactly one **material boundary segment** $\overline{\boldsymbol{p}_{K-1}\,\boldsymbol{p}_K}$ (for some $0 < K \le N$) such that $\boldsymbol{x}_K = \mathsf{X}(\boldsymbol{p}_K, \theta)$ is located on a visibility boundary with respect to $\boldsymbol{x}_{K-1} = \mathsf{X}(\boldsymbol{p}_{K-1}, \theta)$, and vice versa (see Figure 4). The measure $\dot{\mu}$ associated with the material boundary path space $\partial\hat{\Omega}$ is identical to the area-product measure μ over $\hat{\Omega}$ expect that, for a boundary path $\bar{\boldsymbol{p}}$ with boundary segment $\overline{\boldsymbol{p}_{K-1}\,\boldsymbol{p}_K}$, the measure over $\boldsymbol{p}_K$ is replaced with the curve-length measure ℓ:

$$\mathrm{d}\dot{\mu}(\bar{\boldsymbol{p}}) \coloneqq \mathrm{d}\ell(\boldsymbol{p}_K)\prod_{n\neq K}\mathrm{d}A(\boldsymbol{p}_n). \tag{15}$$

At the path level, with vertices $\boldsymbol{p}_n$ fixed for all $n \neq K$, $\boldsymbol{p}_K$ becomes a jump discontinuity point of the material measurement contribution $\hat{f}(\bar{\boldsymbol{p}})$. The term $\Delta\hat{f}(\bar{\boldsymbol{p}})$ then captures the difference in $\hat{f}$ across this discontinuity boundary.

Unlike the material path space $\hat{\Omega}$ that is independent of the scene parameter θ, the material boundary path space $\partial\hat{\Omega}$ does depend on θ in general. This is because, given a fixed $\boldsymbol{p}_{K-1} \in \mathcal{B}$, for $\overline{\boldsymbol{p}_{K-1}\,\boldsymbol{p}_K}$ to be a material boundary segment, the other endpoint $\boldsymbol{p}_K$ typically depends on the scene parameter θ. The term $v(\boldsymbol{p}_K)$ in Eq. (13) is a scalar that captures the change rate of $\boldsymbol{p}_K$ with respect to θ along the normal direction (of the discontinuity boundary). Please refer to §6 of the work by Zhang et al. [2021b] for more details on how this term is calculated.

4.3 Path-Length Boundary Integral

Many path-length importance functions have jump discontinuities. A common example is the (normalized) boxcar function

$$W_\tau(s) = \begin{cases} 1/(\tau_{\max}-\tau_{\min}), & (\tau_{\min} < s < \tau_{\max}) \\ 0, & (\text{otherwise}) \end{cases} \tag{16}$$

with $\Delta\mathbb{R}[W_\tau] = \{\tau_{\min}, \tau_{\max}\}$ for some $\tau_{\min}, \tau_{\max} \in \mathbb{R}_{>0}$.

These discontinuities give rise to a new path integral[3]—the path-length-*boundary* component of Eq. (13). For each discontinuity point $s \in \mathbb{R}_{>0}$ of the path-length importance W_τ, this integral is over a new **material path-length-boundary space** $\partial\hat{\Omega}_\tau(s)$ that consists of material paths $\bar{\boldsymbol{p}}_+$ satisfying that the geometric lengths $\|\bar{\boldsymbol{x}}_+\|$ of the corresponding light paths $\bar{\boldsymbol{x}}_+ := \bar{\mathsf{X}}(\bar{\boldsymbol{p}}_+, \theta)$ equal s. Precisely,

$$\partial\hat{\Omega}_\tau(s) := \left\{ \bar{\boldsymbol{p}}_+ \in \hat{\Omega} \ : \ \|\bar{\boldsymbol{x}}_+\| = s \right\}. \tag{17}$$

The measure $\mathrm{d}\hat{\mu}_\tau$ over $\partial\hat{\Omega}_\tau(s)$ is similar to that over the material boundary path space:

$$\mathrm{d}\hat{\mu}_\tau(\bar{\boldsymbol{p}}_+) := \mathrm{d}\ell(\boldsymbol{q}) \prod_n \mathrm{d}A(\boldsymbol{p}_n), \tag{18}$$

for any $\bar{\boldsymbol{p}}_+ := (\boldsymbol{p}_0, \boldsymbol{q}, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$.

Additionally, when $\boldsymbol{p}_0, \ldots, \boldsymbol{p}_N$ are fixed, for a material light path $\bar{\boldsymbol{p}}_+ = (\boldsymbol{p}_0, \boldsymbol{q}, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ to be an element of the material path-length-boundary space $\partial\hat{\Omega}_\tau(s)$, the vertex $\boldsymbol{q}$ must reside on a curve:

$$\Delta\mathcal{B}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) := \{ \boldsymbol{q} \in \mathcal{B} \ : \ \|\bar{\boldsymbol{x}}_+\| = s \}, \tag{19}$$

which typically depends on the scene parameter θ. If we treat $W_\tau(\|\bar{\boldsymbol{x}}_+\|)$ as a function $W_\tau(\cdot; \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) : \mathcal{B} \mapsto \mathbb{R}_{\geq 0}$ of the vertex $\boldsymbol{q}$:

$$W_\tau(\boldsymbol{q}; \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) := W_\tau(\|\bar{\boldsymbol{x}}_+\|), \tag{20}$$

it is easy to verify that the curve $\Delta\mathcal{B}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ is comprised of discontinuity points of this function.

Lastly, in the path-length-*boundary* component of Eq. (13), ΔW_τ denotes the difference in path-length importance W_τ across the discontinuity boundary:

$$\begin{aligned} \Delta W_\tau(\|\bar{\boldsymbol{x}}_+\|) := & \lim_{\epsilon \to 0^-} W_\tau(\boldsymbol{q} + \epsilon \boldsymbol{n}_{\Delta\mathcal{B}}(\boldsymbol{q}); \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) - \\ & \lim_{\epsilon \to 0^+} W_\tau(\boldsymbol{q} + \epsilon \boldsymbol{n}_{\Delta\mathcal{B}}(\boldsymbol{q}); \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N), \end{aligned} \tag{21}$$

where $\boldsymbol{n}_{\Delta\mathcal{B}}(\boldsymbol{q})$ denotes the unit normal of the discontinuity curve $\Delta\mathcal{B}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ at $\boldsymbol{q}$. Further, $v_\tau(\boldsymbol{q})$ is a scalar change rate capturing how this curve evolves with respect to θ (see Figure 5 for an illustration). In what follows, we provide more details on the calculation of this change rate.

Calculating $v_\tau(\boldsymbol{q})$. Let $\Delta\mathcal{M}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) \subset \mathcal{M}(\theta)$ be the image of $\Delta\mathcal{B}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ defined in Eq. (19) via the mapping $\mathsf{X}(\cdot, \theta)$. Precisely,

$$\Delta\mathcal{M}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) := \left\{ \mathsf{X}(\boldsymbol{q}, \theta) \ : \ \boldsymbol{q} \in \Delta\mathcal{B}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) \right\}. \tag{22}$$

Then, for any θ, each $\boldsymbol{y} \in \Delta\mathcal{M}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ must satisfy

$$\|\overline{\boldsymbol{y}\,\boldsymbol{x}_0}\| + \|\overline{\boldsymbol{y}\,\boldsymbol{x}_1}\| = s - \sum_{n=2}^{N} \|\overline{\boldsymbol{x}_{n-1}\,\boldsymbol{x}_n}\| =: \tau_0, \tag{23}$$

where $\|\overline{\boldsymbol{x}\,\boldsymbol{y}}\| := \|\boldsymbol{x} - \boldsymbol{y}\|$ denotes the length of the line segment $\overline{\boldsymbol{x}\,\boldsymbol{y}}$, and $\boldsymbol{x}_n := \mathsf{X}(\boldsymbol{p}_n, \theta)$ for $n = 0, 1, \ldots, N$. It is easy to verify that

[3]Please refer to Appendix A for a derivation of this result.

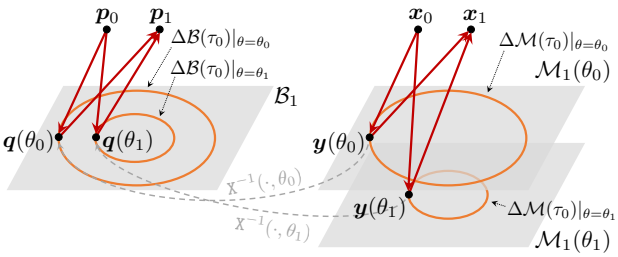

Fig. 5. **Evolution of path-length boundaries:** This example shows a simple scene where the vertical position of a planar surface $\mathcal{M}_1$ is controlled by a parameter θ (with everything else held static), as shown on the right. The corresponding reference configuration is shown on the left. With $\boldsymbol{p}_0, \boldsymbol{p}_1, \ldots$ fixed, all $\boldsymbol{y} \in \mathcal{M}_1$ such that $\|\overline{\boldsymbol{x}_0\,\boldsymbol{y}}\| + \|\overline{\boldsymbol{x}_1\,\boldsymbol{y}}\|$ equals some fixed $\tau_0 \in \mathbb{R}_{>0}$ (given by discontinuities of path-length importance W_τ) form ellipses $\Delta\mathcal{M}(\tau_0)$ illustrated in orange. When mapped back to the reference configuration using $\mathsf{X}^{-1}(\cdot, \theta) : \mathcal{M}_1(\theta) \mapsto \mathcal{B}_1$, these ellipses form the discontinuity curve $\Delta\mathcal{B}(\tau_0)$ with respect to $\boldsymbol{q}$. We omit the dependencies of $\Delta\mathcal{M}$ and $\Delta\mathcal{B}$ on $\boldsymbol{p}_0, \boldsymbol{p}_1, \ldots$ for notational simplicity.

$\Delta\mathcal{M}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ is the intersection between the scene geometry $\mathcal{M}(\theta)$ and an ellipsoid (which has foci $\boldsymbol{x}_0$, $\boldsymbol{x}_1$ and the sum of distances τ_0).

In practice, when the scene geometry is described using polygonal meshes, it is easy to verify that $\Delta\mathcal{M}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ defined in Eq. (22) takes the form of a set of elliptical arcs. Given a point $\boldsymbol{y} \in \Delta\mathcal{M}(\tau_0, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$, we can calculate its derivative $\mathrm{d}\boldsymbol{y}/\mathrm{d}\theta$ by locally parameterizing the elliptical arc (whose foci and sum of distances can be obtained in a differentiable fashion). Upon fully obtaining $\boldsymbol{y}$ and $\mathrm{d}\boldsymbol{y}/\mathrm{d}\theta$, we can calculate the derivative $\mathrm{d}\boldsymbol{q}/\mathrm{d}\theta$ by differentiating the relation $\boldsymbol{q} = \mathsf{X}^{-1}(\boldsymbol{y}, \theta)$. Lastly, we calculate the change rate v_τ using

$$v_\tau(\boldsymbol{q}) = \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}\theta} \cdot \boldsymbol{n}_{\Delta\mathcal{B}}(\boldsymbol{q}), \tag{24}$$

where "$\cdot$" is the dot-product operator.

Relation to Prior Works

Theoretically, our differential time-gated path integral of Eq. (13) is a significant generalization of the steady-state variant [Zhang et al. 2020]. Specifically, our result has included the path-length importance W_τ into the *interior* and the visibility-*boundary* components. We have also introduced the path-length-*boundary* integral (§4.3), which is unique to time-gated differentiable rendering.

In the special case where $W_\tau(s) \equiv 1$, the path-length-*boundary* component vanishes (as $\Delta\mathbb{R}[W_\tau] = \emptyset$), and Eq. (13) reduces to the steady-state result.

5 MONTE CARLO ESTIMATORS

We now present how the differential time-gated path integral can be estimated in an unbiased fashion using Monte Carlo integration, which in turn amounts to estimating the *interior* and *boundary* path integrals in Eq. (13).

Specifically, we focus on the problem of estimating the derivative $\mathrm{d}I/\mathrm{d}\theta$ at some user-specified $\theta = \theta_0$. As discussed in the end of §3, we set the reference surface to the actual scene geometry at θ_0:

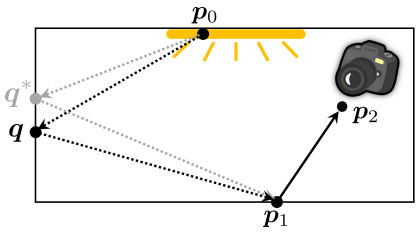

Fig. 6. **Antithetic sampling:** To efficiently handle near-delta path-length importance functions W_τ whose derivatives are odd functions, we utilize antithetic sampling and generate pairs of correlated light paths $\bar{\boldsymbol{p}}_+ = (\boldsymbol{p}_0, \boldsymbol{q}, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ and $\bar{\boldsymbol{p}}^*_+ = (\boldsymbol{p}_0, \boldsymbol{q}^*, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ with $\frac{\mathrm{d}}{\mathrm{d}s} W_\tau(\|\bar{\boldsymbol{p}}_+\|) = -\frac{\mathrm{d}}{\mathrm{d}s} W_\tau(\|\bar{\boldsymbol{p}}^*_+\|)$. To this end, we fix $\boldsymbol{p}_0, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N$ and sample $\boldsymbol{q}$ and $\boldsymbol{q}^*$ from two elliptical curves in a correlated fashion.

$\mathcal{B} = \mathcal{M}(\theta_0)$. This causes the material path space $\hat{\Omega}$ to coincide with the ordinary one $\Omega(\theta_0)$.

In what follows, we discuss the estimation of the *interior* integral in §5.1 and the *boundary* components in §5.2 and §5.3.

5.1 Estimating the Interior Integral

Since the *interior* integral in Eq. (13) is over the same domain—the material path space $\hat{\Omega}$—as Eq. (9), we use path sampling methods previously developed for forward time-gated rendering to estimate this *interior* integral.

Standard path sampling. A simple solution is to sample material light paths $\bar{\boldsymbol{p}}$—which coincide with the ordinary ones at $\theta = \theta_0$—using standard methods like unidirectional path tracing. Although this method works adequately when the path-length importance W_τ has a wide support, it can become inefficient when W_τ becomes concentrated.

Ellipsoidal next-event estimation. To estimate the *interior* integral more efficiently when W_τ is near-delta, we utilize the ellipsoidal reparameterization of Eq. (12). Specifically, by setting $F(\bar{\boldsymbol{p}}) \coloneqq \frac{\mathrm{d}}{\mathrm{d}\theta}\left[W_\tau(\|\bar{\boldsymbol{x}}\|)\, \hat{f}(\bar{\boldsymbol{p}})\right]$, we develop an unidirectional algorithm with an *ellipsoidal next-event estimation (NEE)* that works as follows. Note that the Jacobian determinant J^{e}_i from the ellipsoidal reparameterization is outside of the differentiation operator, so we do not need to compute its derivatives with respect to the scene parameter θ.

Our technique starts with tracing a material light path $\bar{\boldsymbol{p}}^{\mathrm{D}} = (\boldsymbol{p}^{\mathrm{D}}_0, \boldsymbol{p}^{\mathrm{D}}_1, \ldots)$ from the detector using standard unidirectional path tracing. Then, at each vertex $\boldsymbol{p}^{\mathrm{D}}_n$, we randomly sample a point $\boldsymbol{p}^{\mathrm{S}}$ on a light source, and a length $\tau \in \mathbb{R}_{>0}$ with a probability density proportional to the path-length importance W_τ (i.e., $p(\tau) \propto W_\tau(\tau)$).

With $\boldsymbol{p}^{\mathrm{S}}$ and τ obtained, we aim to find another vertex $\boldsymbol{q} \in \mathcal{B}$ such that the path $\bar{\boldsymbol{p}}_+ = (\boldsymbol{p}^{\mathrm{S}}, \boldsymbol{q}, \boldsymbol{p}^{\mathrm{D}}_n \ldots, \boldsymbol{p}^{\mathrm{D}}_0)$ has a total length of τ. We note that, under our choice of reference surface $\mathcal{B} = \mathcal{M}(\theta_0)$, the original path $\bar{\boldsymbol{x}}_+ \coloneqq \bar{\mathrm{X}}(\bar{\boldsymbol{p}}_+, \theta_0)$ coincides with $\bar{\boldsymbol{p}}_+$ and has the same total length τ. In this way, the resulting probability density for sampling $\bar{\boldsymbol{p}}_+$ will involve a factor of $W_\tau(\tau)$, canceling this term (and partially its derivative) in the integrand $\frac{\mathrm{d}}{\mathrm{d}\theta}\left[W_\tau(\|\bar{\boldsymbol{x}}\|)\, \hat{f}(\bar{\boldsymbol{p}})\right]$.

Let $\tau_0 \coloneqq \tau - \sum_{i=1}^{n} \left\|\overline{\boldsymbol{p}^{\mathrm{D}}_{i-1}\, \boldsymbol{p}^{\mathrm{D}}_i}\right\|$ be the desired total length of $\boldsymbol{p}^{\mathrm{S}} \to \boldsymbol{q} \to \boldsymbol{p}^{\mathrm{D}}_n$. To find a point $\boldsymbol{q}$ satisfying this constraint, we first draw a polygonal face $\mathcal{T}_i$ (that intersects the ellipsoid determined with foci $\boldsymbol{p}^{\mathrm{S}}$ and $\boldsymbol{p}^{\mathrm{D}}_n$, as well as the sum of distances τ_0) and then sample $\phi \in \mathbb{R}$ that parameterizes $\mathcal{E}_i(\tau_0)$ (see Figure 3).

Previously, Pediredla et al. [2019b] used a bidirectional path sampling scheme for forward time-gated rendering. Although our framework of differential time-gated path integral is fully compatible with bidirectional path sampling, we found the unidirectional version presented above to work well in all our experiments.

Antithetic sampling. Many commonly used path-length importance W_τ—such as Gaussian functions—have derivatives $\mathrm{d}W_\tau(s)/\mathrm{d}s$ that are odd functions. When W_τ is near-delta, the derivative involves high-magnitude positive and negative regions that can yield high variance when estimating the *interior* integral even with our ellipsoidal next-event estimation discussed above.

To overcome this, we adopt the antithetic sampling method introduced by Zhang et al. [2021a] for handling near-specular BSDFs. Assume the derivative $\mathrm{d}W_\tau(s)/\mathrm{d}s$ to satisfy

$$\frac{\mathrm{d}}{\mathrm{d}s} W_\tau(\bar{\tau} + s) = -\frac{\mathrm{d}}{\mathrm{d}s} W_\tau(\bar{\tau} - s), \tag{25}$$

for some constant $\bar{\tau} \in \mathbb{R}_{>0}$ and any $0 < s < \bar{\tau}$. When performing our ellipsoidal next-event estimation, after drawing one point $\boldsymbol{p}_0$ and a length $\tau \in \mathbb{R}_{>0}$, we select a second length $\tau^* = 2\bar{\tau} - \tau$. Then, we sample two points $\boldsymbol{q}, \boldsymbol{q}^* \in \mathcal{B}$ in a correlated fashion (i.e., by using identical random numbers) such that the two paths $\bar{\boldsymbol{p}}_+ = (\boldsymbol{p}_0, \boldsymbol{q}, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ and $\bar{\boldsymbol{p}}^*_+ = (\boldsymbol{p}_0, \boldsymbol{q}^*, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ have total lengths τ and τ^*, respectively (see Figure 6). When W_τ is near-delta, we will typically have $\boldsymbol{q}, \boldsymbol{q}^*$ in close proximity, and

$$\frac{\mathrm{d}}{\mathrm{d}\theta} W_\tau(\|\bar{\boldsymbol{x}}_+\|) \approx -\frac{\mathrm{d}}{\mathrm{d}\theta} W_\tau(\|\bar{\boldsymbol{x}}^*_+\|), \tag{26}$$

where $\bar{\boldsymbol{x}}^*_+ = \bar{\mathrm{X}}(\bar{\boldsymbol{p}}^*_+, \theta)$. This, in turn, ensures that the contributions of the two paths $\bar{\boldsymbol{p}}_+$ and $\bar{\boldsymbol{p}}^*_+$ (to the *interior* term) largely cancel out each other.

We will demonstrate the effectiveness of our antithetic sampling in §6.1.

5.2 Estimating the Visibility-Boundary Integral

We now consider the problem of estimating the *boundary* integral in Eq. (13). Unlike the *interior* integral that can be estimated using path sampling strategies developed for forward time-gated rendering, this *boundary* integral is unique to differentiable time-gated rendering.

In the following, we introduce a new estimator that utilizes a multi-directional sampling technique introduced by Zhang et al. [2020] and the ellipsoidal next-event estimation described in §5.1 for near-delta path-length importance functions.

Multi-directional sampling. Recall that the *boundary* integral is over the material boundary path space $\partial\hat{\Omega}$ comprised of light transport paths $\bar{\boldsymbol{p}} = (\boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ containing exactly one material boundary segment $\overline{\boldsymbol{p}_{K-1}\, \boldsymbol{p}_K}$ (for some $0 < K \leq N$).

Since the material boundary segment breaks a material boundary path into a source and a detector subpath, we rename the vertices of

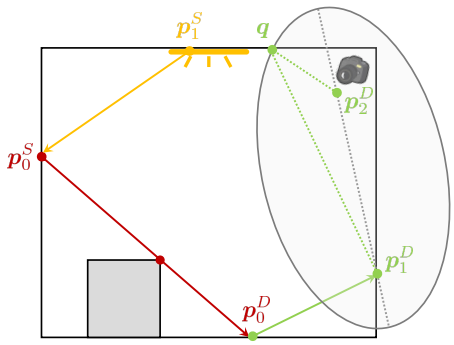

Fig. 7. **Multi-directional sampling of *boundary* paths:** To estimate the *boundary* integral in its multi-directional form given by Eq. (28), we sample material boundary paths as follows. We first draw the material boundary segment $\overline{\boldsymbol{p}_0^{\mathrm{S}}\,\boldsymbol{p}_0^{\mathrm{D}}}$ (illustrated in red) followed by the source subpath $\bar{\boldsymbol{p}}^{\mathrm{S}}$ (shown in yellow). Based on total length of $\overline{\boldsymbol{p}_0^{\mathrm{S}}\,\boldsymbol{p}_0^{\mathrm{D}}}$ and $\bar{\boldsymbol{p}}^{\mathrm{S}}$, we then sample the detector subpath $\bar{\boldsymbol{p}}^{\mathrm{D}}$—which equals $(\boldsymbol{p}_0^{\mathrm{D}}, \boldsymbol{p}_1^{\mathrm{D}}, \boldsymbol{q}, \boldsymbol{p}_2^{\mathrm{D}})$ in this example—using ellipsoidal next-event estimation.

this path as $\bar{\boldsymbol{p}} = (\boldsymbol{p}_s^{\mathrm{S}}, \ldots, \boldsymbol{p}_0^{\mathrm{S}}, \boldsymbol{p}_0^{\mathrm{D}}, \ldots, \boldsymbol{p}_t^{\mathrm{D}})$, where $\overline{\boldsymbol{p}_0^{\mathrm{S}}\,\boldsymbol{p}_0^{\mathrm{D}}}$ is the material boundary segment, $\bar{\boldsymbol{p}}^{\mathrm{S}} \coloneqq (\boldsymbol{p}_s^{\mathrm{S}}, \ldots, \boldsymbol{p}_0^{\mathrm{S}})$ is the source subpath, and $\bar{\boldsymbol{p}}^{\mathrm{D}} \coloneqq (\boldsymbol{p}_0^{\mathrm{D}}, \ldots, \boldsymbol{p}_t^{\mathrm{D}})$ is the detector subpath. Further, we decompose $\Delta\hat{f}(\bar{\boldsymbol{p}})\, V(\boldsymbol{p}_0^{\mathrm{D}})$ into the product of contributions of $\overline{\boldsymbol{p}_0^{\mathrm{S}}\,\boldsymbol{p}_0^{\mathrm{D}}}$, $\bar{\boldsymbol{p}}^{\mathrm{S}}$, and $\bar{\boldsymbol{p}}^{\mathrm{D}}$, respectively:

$$\Delta\hat{f}(\bar{\boldsymbol{p}})\, V(\boldsymbol{p}_0^{\mathrm{D}}) = \underbrace{\hat{f}^{\mathrm{B}}(\boldsymbol{p}_0^{\mathrm{S}}, \boldsymbol{p}_0^{\mathrm{D}})}_{\text{bound seg.}} \underbrace{\hat{f}^{\mathrm{S}}(\bar{\boldsymbol{p}}^{\mathrm{S}}; \boldsymbol{p}_0^{\mathrm{D}})}_{\text{source subpath}} \underbrace{\hat{f}^{\mathrm{D}}(\bar{\boldsymbol{p}}^{\mathrm{D}}; \boldsymbol{p}_0^{\mathrm{S}})}_{\text{detector subpath}}. \tag{27}$$

Please refer to the work by Zhang et al. [2020] for the exact forms of $\hat{f}^{\mathrm{B}}$, $\hat{f}^{\mathrm{S}}$, and $\hat{f}^{\mathrm{D}}$ in this equation.

Let $\bar{\boldsymbol{p}}_0^{\mathrm{S}}$ and $\bar{\boldsymbol{p}}_0^{\mathrm{D}}$ denote the source and detector subpaths with $\boldsymbol{p}_0^{\mathrm{S}}$ and $\boldsymbol{p}_0^{\mathrm{D}}$ excluded, respectively. That is, $\bar{\boldsymbol{p}}_0^{\mathrm{S}} \coloneqq (\boldsymbol{p}_s^{\mathrm{S}}, \ldots, \boldsymbol{p}_1^{\mathrm{S}})$ and $\bar{\boldsymbol{p}}_0^{\mathrm{D}} \coloneqq (\boldsymbol{p}_1^{\mathrm{D}}, \ldots, \boldsymbol{p}_t^{\mathrm{D}})$. Based on Eq. (27), we rewrite the *boundary* term in Eq. (13) in its *multi-directional form* as:

$$\begin{aligned}&\int_{\partial\hat{\Omega}} W_\tau(\|\bar{\boldsymbol{x}}\|)\, \Delta\hat{f}(\bar{\boldsymbol{p}})\, V(\boldsymbol{p}_0^{\mathrm{D}})\, \mathrm{d}\dot{\mu}(\bar{\boldsymbol{p}})\\ &= \int_{\mathcal{B}}\int_{\partial\mathcal{B}} \hat{f}^{\mathrm{B}}(\boldsymbol{p}_0^{\mathrm{S}}, \boldsymbol{p}_0^{\mathrm{D}}) \left(\int \hat{f}^{\mathrm{S}}(\bar{\boldsymbol{p}}^{\mathrm{S}}; \boldsymbol{p}_0^{\mathrm{D}})\, h(\boldsymbol{p}_0^{\mathrm{D}}, \bar{\boldsymbol{p}}^{\mathrm{S}})\, \mathrm{d}\bar{\boldsymbol{p}}_0^{\mathrm{S}}\right) \mathrm{d}\ell(\boldsymbol{p}_0^{\mathrm{D}})\, \mathrm{d}A(\boldsymbol{p}_0^{\mathrm{S}}),\end{aligned} \tag{28}$$

where

$$h(\boldsymbol{p}_0^{\mathrm{D}}, \bar{\boldsymbol{p}}^{\mathrm{S}}) \coloneqq \int W_\tau(\|\bar{\boldsymbol{x}}\|)\, \hat{f}^{\mathrm{D}}(\bar{\boldsymbol{p}}^{\mathrm{D}}; \boldsymbol{p}_0^{\mathrm{S}})\, \mathrm{d}\bar{\boldsymbol{p}}_0^{\mathrm{D}}. \tag{29}$$

In Eqs. (28) and (29), we omit the domains and measures of the path integrals on $\bar{\boldsymbol{p}}_0^{\mathrm{S}}$ and $\bar{\boldsymbol{p}}_0^{\mathrm{D}}$ for notational simplicity.

As demonstrated by Zhang et al. [2020], the multi-directional-form *boundary* integral of Eq. (28) allows sampling a material boundary path $\bar{\boldsymbol{p}}$ by starting with its boundary segment $\overline{\boldsymbol{p}_0^{\mathrm{S}}\,\boldsymbol{p}_0^{\mathrm{D}}}$, avoiding the need of expensive silhouette detections.

Ellipsoidal next-event estimation. Unlike steady-state differentiable rendering, the *boundary* integral of our differential time-gated path integral of Eq. (13) involves the path-length importance W_τ. When this function has a narrow support, it is desired to importance sample the path length τ when estimating the *boundary* integral.

To this end, we leverage the ellipsoidal reparameterization by applying Eq. (12) to Eq. (29). This allows us to perform the ellipsoidal next-event estimation presented in §5.1 when building the camera subpath.

Specifically, to estimate Eq. (28), we first draw a desired total path length $\tau \in \mathbb{R}$ based on the path-length importance W_τ. Then, we utilize the multi-directional path sampling approach introduce by Zhang et al. [2020] to draw the boundary segment $\overline{\boldsymbol{p}_0^{\mathrm{S}}\,\boldsymbol{p}_0^{\mathrm{D}}}$ followed by the source subpath $\bar{\boldsymbol{p}}^{\mathrm{S}}$ that starts with $\boldsymbol{p}_0^{\mathrm{S}}$. In this way, the desired length of the detector subpath $\bar{\boldsymbol{p}}^{\mathrm{D}}$ equals $\tau' = \tau - \|\overline{\boldsymbol{p}_0^{\mathrm{S}}\,\boldsymbol{p}_0^{\mathrm{D}}}\| - \|\bar{\boldsymbol{p}}^{\mathrm{S}}\|$. To construct this subpath, we start from $\boldsymbol{p}_0^{\mathrm{D}}$ and use our ellipsoidal next-event estimation to find a subpath satisfying the length constraint (see Figure 7).

5.3 Estimating the Path-Length-Boundary Integral

We now discuss the Monte Carlo estimation of the path-length-*boundary* component in Eq. (13), which is unique to differentiable time-gated rendering.

Recall that, due to our choice of the reference configuration presented in §4.1, any material light path $\bar{\boldsymbol{p}}$ and the corresponding light path $\bar{\boldsymbol{x}} = \bar{\mathrm{X}}(\bar{\boldsymbol{p}}, \theta_0)$ coincide and have identical geometric lengths (i.e., $\|\bar{\boldsymbol{p}}\| = \|\bar{\boldsymbol{x}}\|$). Thus, we have

$$\partial\hat{\Omega}_\tau(s)|_{\theta=\theta_0} = \left\{\bar{\boldsymbol{p}}_+ \in \hat{\Omega} \;:\; \|\bar{\boldsymbol{p}}_+\| = s\right\}, \tag{30}$$

for all path length $s \in \Delta\mathbb{R}[W_\tau]$. This allows us to sample material paths $\bar{\boldsymbol{x}}_+ \in \partial\hat{\Omega}_\tau(s)$ using our ellipsoidal next-event estimation discussed in §5.1 as follows.

Given a desired length s, we build a light path $(\boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ using standard unidirectional path tracing (starting from $\boldsymbol{p}_N$ on the detector), followed by sampling $\boldsymbol{p}_0$ on a light source. Then, we draw $\boldsymbol{q}$ from an elliptical arc determined by the constraint that $\|\overline{\boldsymbol{p}_0\,\boldsymbol{q}}\| + \|\overline{\boldsymbol{p}_1\,\boldsymbol{q}}\| = s - \sum_{n=2}^{N}\|\overline{\boldsymbol{p}_{n-1}\,\boldsymbol{p}_n}\| \eqqcolon \tau_0$. This leads to the following single-sample Monte Carlo estimator:

$$\left\langle \frac{\Delta W_\tau(s)\, \hat{f}(\bar{\boldsymbol{p}}_+)\, v_\tau(\boldsymbol{q})}{\mathbb{P}((\boldsymbol{p}_1, \ldots, \boldsymbol{p}_N))\, \mathbb{P}(\boldsymbol{p}_0)\, \mathbb{P}(\boldsymbol{q} \mid \boldsymbol{p}_0, \boldsymbol{p}_1)} \right\rangle, \tag{31}$$

where $\bar{\boldsymbol{p}}_+ = (\boldsymbol{p}_0, \boldsymbol{q}, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$. Further, $\mathbb{P}((\boldsymbol{p}_1, \ldots, \boldsymbol{p}_N))$ and $\mathbb{P}(\boldsymbol{p}_0)$ are, respectively, probability densities for building the subpath $(\boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ and drawing the point $\boldsymbol{p}_0$ on the surface of a light source. Both of these probability densities are computed based on the standard unidirectional path tracing process. Lastly, $\mathbb{P}(\boldsymbol{q})$ denotes the probability density for drawing $\boldsymbol{q}$ from a path-length discontinuity curve $\Delta\mathcal{B}(\tau_0, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) \subset \mathcal{B}$.

We note that, although the use of ellipsoidal next-event estimation is optional when estimating the *interior* and the visibility-*boundary* components, its use is mandatory when estimating the path-length-*boundary* component due to the need of sampling paths $\bar{\boldsymbol{p}}_+$ with fixed geometric lengths. Also, since no change of variable is applied, the Jacobian determinant J_i^e from Eq. (6) is absent from Eq. (31).

Antithetic sampling. Near-delta path-length importances W_τ can also cause high-variance estimates of the path-length-*boundary* component when, for example, using boxcar functions of Eq. (16) with $(\tau_{\max} - \tau_{\min})$ being close to zero. Similar to our handling of the *interior* component (discussed in §5.1), we address this problem

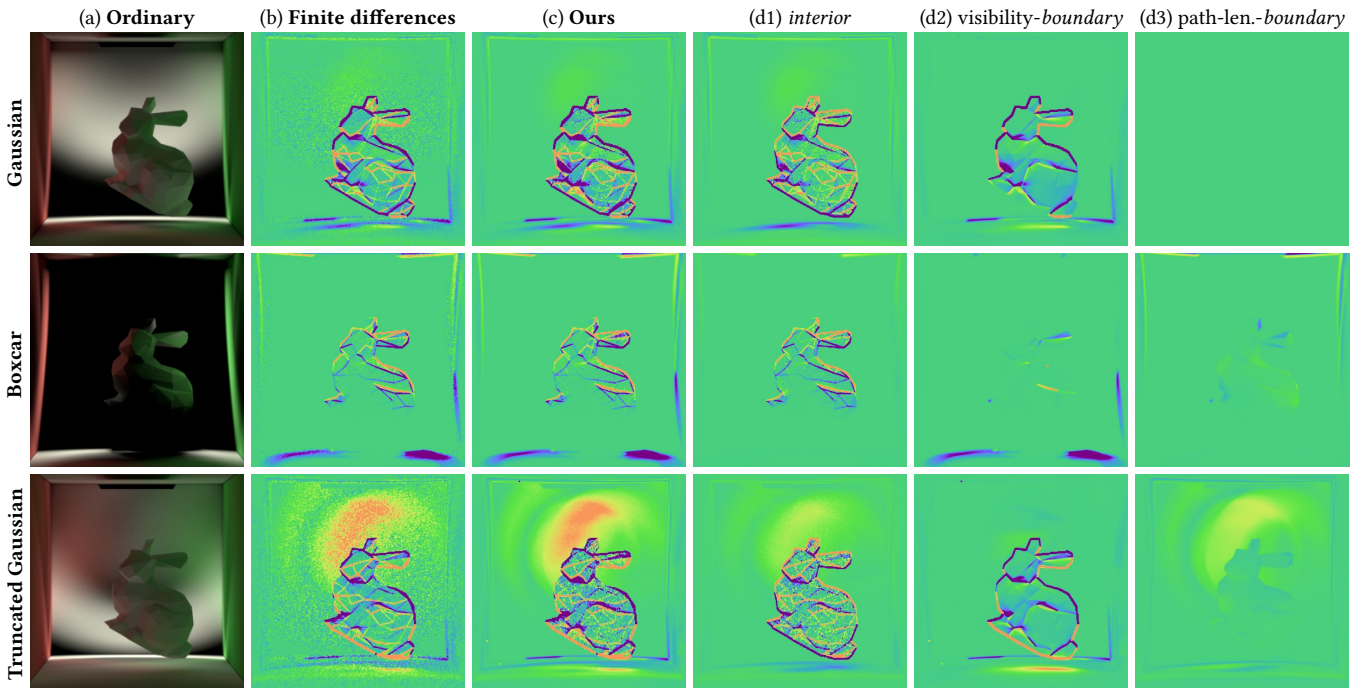

Fig. 8. **Validation** of derivatives estimated with our Monte Carlo estimators. All examples share identical scene configurations except for using varying path-length importance functions (Gaussian, boxcar, and truncated Gaussian for the top, middle, and bottom rows, respectively). For each example, we show the ordinary time-gated rendering in (a) and derivatives with respect to the vertical position of the bunny estimated using finite differences in (b) and our method in (c). Additionally, we visualize the *interior*, visibility-*boundary*, and path-length-*boundary* contributions computed by our estimators in (d1), (d2), and (d3), respectively. The sum of these contributions give the full derivative estimates in (c).

by building correlated paths. In case of boxcar filters, we build $\bar{\boldsymbol{p}}_+ = (\boldsymbol{p}_0, \boldsymbol{q}, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ and $\bar{\boldsymbol{p}}^*_+ = (\boldsymbol{p}_0, \boldsymbol{q}^*, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$ with total lengths τ_{max} and τ_{min}, respectively. This allows the contributions of these paths given by Eq. (31) to largely cancel out each other, offering significant variance reduction.

6 RESULTS

In what follows, we validate our implementation in §6.1. Additionally, we demonstrate in §6.2 the potential usefulness of our technique using proof-of-concept synthetic inverse-rendering examples that simulate both line-of-sight and non-line-of-sight imaging scenarios.

Please refer to the supplemental material for animated versions of the inverse-rendering results.

6.1 Validation and Evaluation

Validation. We validate our estimators by comparing derivative estimates with references generated using finite differences in Figure 8. In this figure, all results share a scene with a diffuse bunny inside a Cornell box. We show derivatives (with respect to the vertical position of the bunny) computed using finite differences and our method in columns (b) and (c), respectively. Additionally, we demonstrate in (d1)–(d3) how the *interior*, visibility-*boundary*, and path-length-*boundary* components of our differential time-gated path integral of Eq. (13) contribute to the final derivatives.

Each row of Figure 8 uses a different type of path-length importance W_τ. Specifically, results in the top row are generated with a Gaussian path-length importance. In this case, the path-length-*boundary* component vanishes as W_τ is smooth. The second and third rows have W_τ being boxcar (16) and truncated Gaussian functions, respectively. In both cases, assuming W_τ to have the support $(\tau_{\text{min}}, \tau_{\text{max}}) \subset \mathbb{R}$, the path-length-*boundary* component involves paths with lengths τ_{min} and τ_{max}. We allocate an equal number of samples to estimate the *interior*, visibility-*boundary*, and path-length-*boundary* components and the fraction of running time is about $1:1:3$, since the ellipsoidal NEE is required for the path-length-*boundary* component and it brings more overhead.

Each finite-difference result in Figure 8 takes more than 72 hours to generate (using an AWS EC2 `c5a.12xlarge` instance with 48 CPU cores) but still contains visible Monte Carlo noise. Our technique, on the other hand, provides cleaner and unbiased gradient estimates in significantly less time.

Differentiable-rendering evaluation. We demonstrate the effectiveness of our ellipsoidal next-event estimation and antithetic sampling (described in §5) in Figure 9. In this example, we show time-gated rendering of a glossy cube with W_τ being a truncated Gaussian function with a narrow support. The top and bottom rows of this figure show, respectively, estimated *interior* and path-length-*boundary* components. When using standard unidirectional path tracing, as

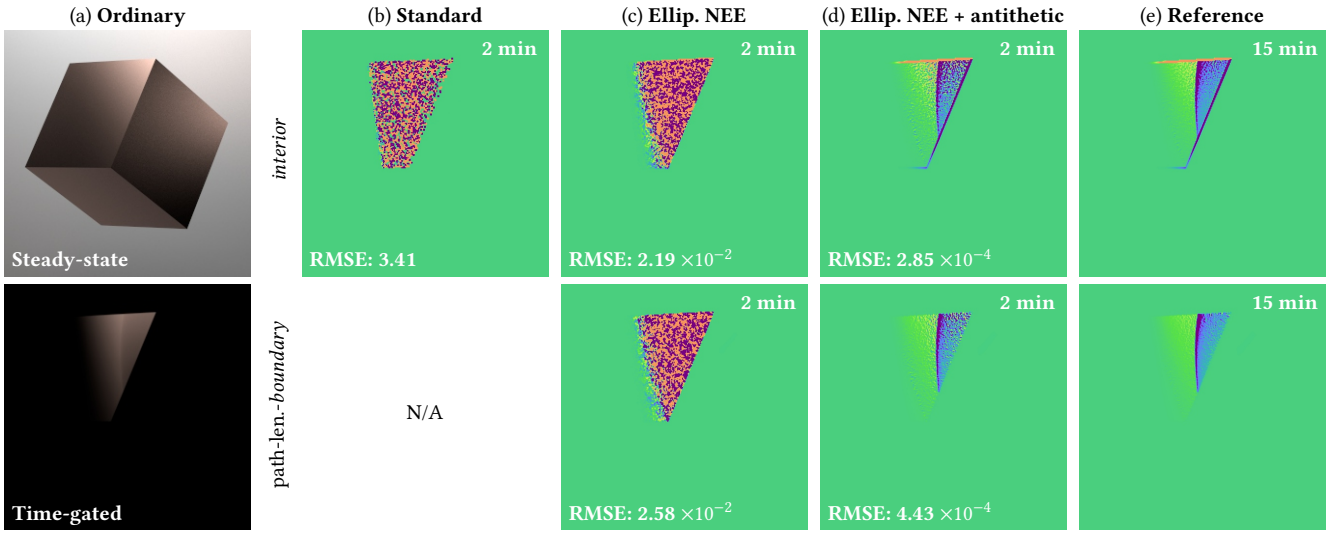

Fig. 9. We utilize **ellipsoidal next-event estimation** and **antithetic sampling** to efficiently estimate the *interior* and the path-length-*boundary* integrals when the path-length importance W_τ is near-delta. This example contains a glossy cube lit by an area light, where W_τ is a narrow truncated Gaussian function, and the derivatives are computed with respect to the vertical position of the cube.

shown in Figure 9-b, the estimated *interior* component suffers from extremely high noise since the chance for randomly sampling a path with desired length is very low. Further, the standard path sampling does not work for the path-length-*boundary* component since the estimation requires sampling light paths with specific lengths. On the contrary, our ellipsoidal next-event estimation produces estimates with less error in equal time, as shown in Figure 9-c. When coupled with antithetic sampling by generating correlated pairs of paths $\bar{\boldsymbol{p}}_+$ and $\bar{\boldsymbol{p}}_+^*$, as demonstrated in Figure 9-d, significant reduction of variance can be obtained.

Inverse-rendering evaluation. Many prior works (e.g., [Iseringhausen and Hullin 2020]) solve inverse time-gated rendering problems by considering only low-order reflections (e.g., direct illumination) and/or estimating derivatives using finite differences. We demonstrate the importance of supporting full interreflection using an inverse-rendering example in Figure 10. Using a non-line-of-sight (NLOS) configuration, this example has a time-of-flight camera located at the opening of a corridor that has two segments joined in the middle. We search for global translations of the four vertices that are at the far end of the second segment and invisible to the camera (as illustrated in Figure 10-a). This problem is challenging as changes of these translations have subtle effects on the time-of-flight images. Thus, having unbiased and low-variance gradient estimates is crucial.

We solve this inverse problem using three configurations for which we use the Adam method [Kingma and Ba 2014] with identical initializations and learning rates. We also adjust the sample count so that each iteration takes approximately equal time.

Our first configuration limits the number of reflections to three (top row). This causes the time-gated renderings to suffer severe energy loss, causing the inverse-rendering optimization (with gradients estimated with our method) to diverge.

Table 2. Performance statistics for the inverse-rendering results in Figures 10–17. The "*time*" numbers indicate average computation time per iteration (in seconds). The experiments are conducted on an AWS EC2 instance of type `c5a.12xlarge`.

Scene	Corridor (Fig. 10)	Cube (Fig. 11)	Branch (Fig. 12)	Height (Fig. 13)	Bunny (Fig. 15)	Sofa (Fig. 16)	Tree (Fig. 17)
# param.	2	1	1	121	2	59	1
# iter.	200	100	100	500	200	1000	200
time	4.5	5.7	1.0	70.3	17.1	58.2	15.5

The second configuration uses full interreflections and derivatives computed with finite differences (middle row). Although the forward renderings are accurate, the optimization still fails to find the correct solution due to the bias and high variance in gradient estimates.

Our third configuration uses full interreflections with derivatives estimated using our method (described in §4 and §5). Since our gradient estimates are unbiased and low-variance, the inverse-rendering optimization successfully finds the correct result.

Ellipsoidal NEE and antithetic sampling. Lastly, we demonstrate the effectiveness of our ellipsoidal next-event estimation and antithetic sampling for inverse-rendering applications. As shown in Figure 11, we use a scene similar to Figure 9. Using four target ToF images with near-delta Gaussian path-length importance functions, we search for the vertical position of the glossy cube.

Without our ellipsoidal NEE or antithetic sampling, the standard method (that relies on binning) suffers from very high variance, causing the optimization to diverge. In contrast, our ellipsoidal NEE and antithetic sampling offer significant variance reduction, allowing the inverse-rendering optimization to converge easily.

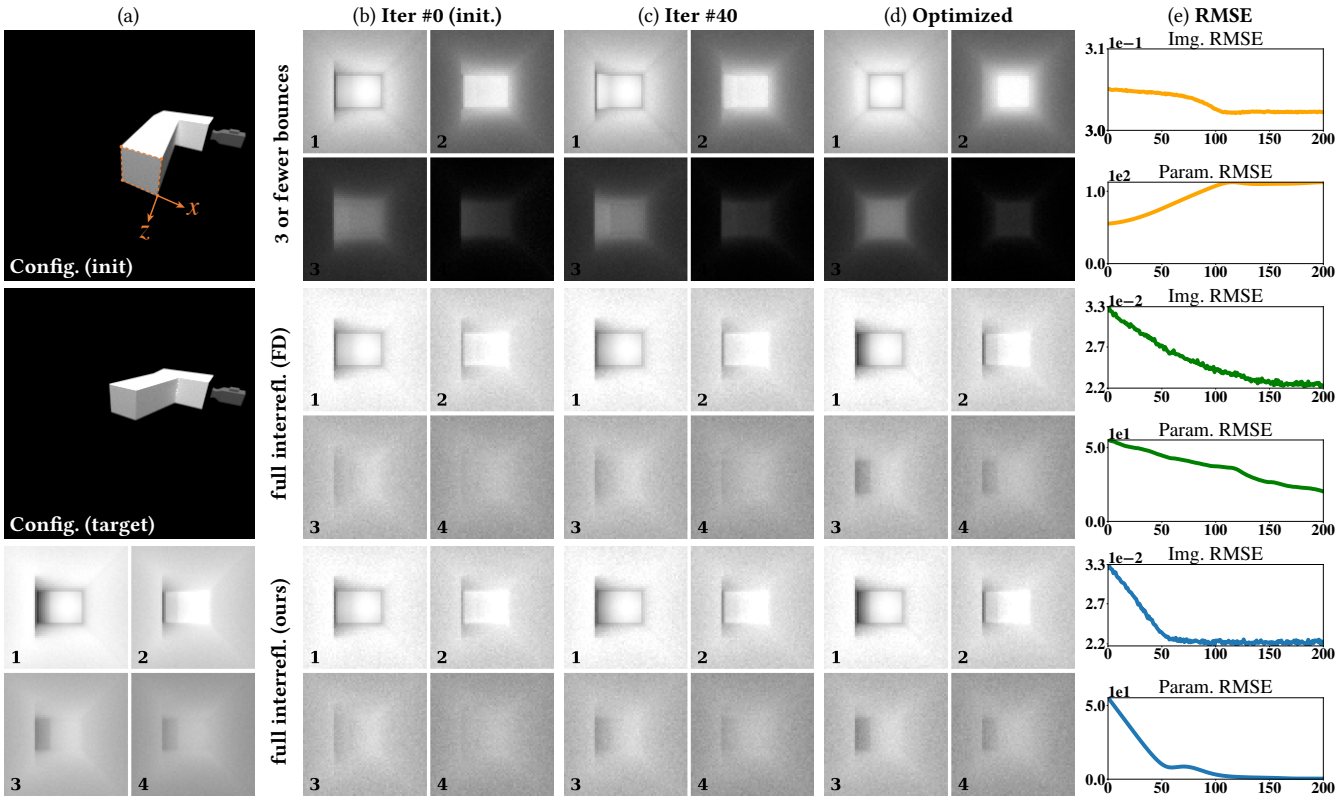

Fig. 10. **Inverse-rendering result (corridor):** As illustrated on the top of column (a), this example involves a diffuse corridor with two segments, and a small area light and a ToF camera are located near the opening of the first segment. Taking as input 20 target ToF images with four of them (marked as 1–4) shown at the bottom of column (a), we search for global translations in x- and z-directions of the four vertices at the end of the second segment. Columns (b–d) contain time-gated renderings illustrating inverse-rendering optimizations using 3-bounce and full interreflections. For the latter, we further compare optimization performance with gradients estimated using finite differences (FD) and our method. The running time per iteration of optimizations using 3-bounce (top row), full interreflections with FD (middle row), and our method (bottom row) are 4.10s, 19.65s, and 4.45s, respectively. Images in (b–d) contain Monte Carlo noise since we use low-sample (but unbiased) estimates for the optimization. The parameter RMSE information is used only for evaluation and not for optimization.

6.2 Inverse-Rendering Results

We now demonstrate the usefulness of our theory and algorithms using a few additional proof-of-concept inverse-rendering examples:

- The **branches** example (Figure 12) is modeled after that from the work by Zhang et al. [2020]. This scene contains a tree-like object casting soft shadows on the ground. Using five time-gated renderings of the shadow (with four shown in the figure) and Gaussian path-length importance, we optimize the rotation angle of the object. We note that, although ordinary time-gated renderings appear similar across iterations, their derivatives are much more distinct. Thus, having these derivatives is crucial for efficiently solving this inverse-rendering problem.
- The **height field** example (Figure 13) includes a diffuse surface lit by an area light from above. Using 12 time-gated renderings and Gaussian path-length importance, we jointly optimize the vertical position of each mesh vertex. The derivative images shown in this figure have zero values in most pixels because the position of one vertex only affects a small set of nearby pixels. Additionally, we show in Figure 14 a reconstruction result obtained using steady-state renderings (under otherwise identical configurations). In this case, since only a single steady-state image is used, the problem becomes significantly more under-constrained than the time-of-flight variant (Figure 13). This causes the optimization to get stuck at a local minimum, as demonstrated by the higher parameter RMSE.
- The **bunny** example (Figure 15) consists of a diffuse box that contains a bunny lit by a small area light. The bunny is made of rough glass, necessitating the handling of light transport beyond three bounces. Both the bunny and the light are blocked from the camera's point of view, mimicking a non-line-of-sight (NLOS) imaging configuration. We use 20 time-gated images with Gaussian path-length importance and optimize the translation of the bunny (along the x- and y-axis).
- The **sofa** example (Figure 16) uses another NLOS setting where a diffuse sofa-like object lit by a large area light is placed behind an occluder (when viewed from the camera). Using 20 time-gated

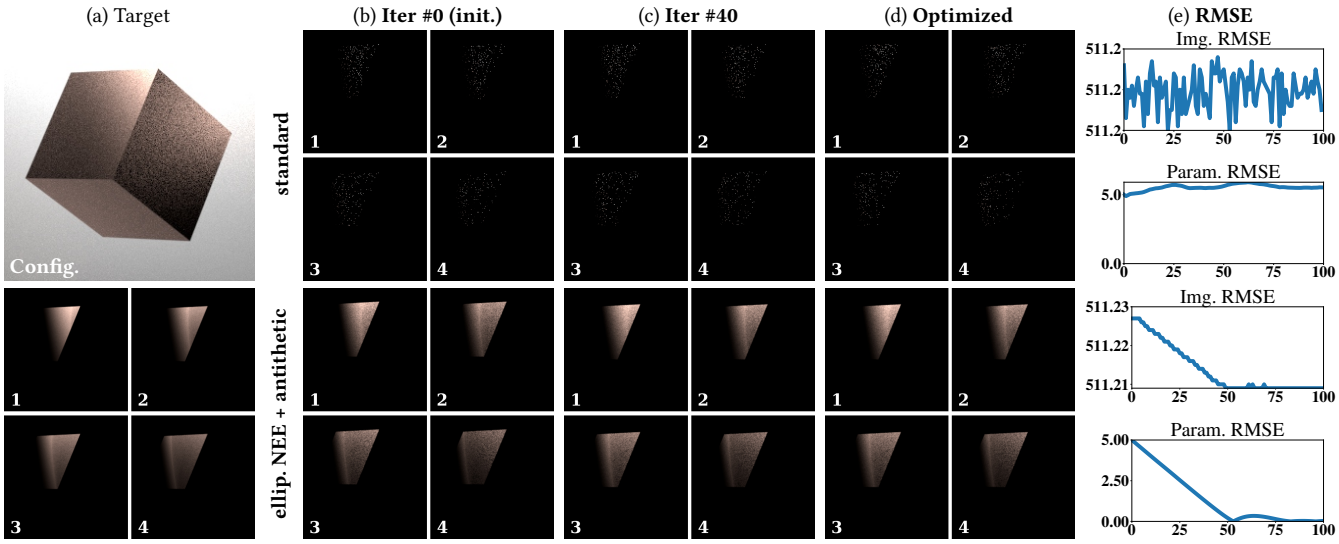

Fig. 11. **Inverse-rendering result (cube):** This example involves a glossy cube lit by an area light from the top. Taking as input four target ToF images (marked as 1–4) shown at the bottom of column (a), we search for the cube's vertical position. Without our ellipsoidal next-event estimation or antithetic sampling, the standard estimator produces very noisy estimates (taking 5.32s per iteration), causing the optimization to diverge (top row). With both techniques, on the contrary, much cleaner gradient estimates can be obtained (taking 5.73s per iteration), allowing the optimization to converge smoothly (bottom row).

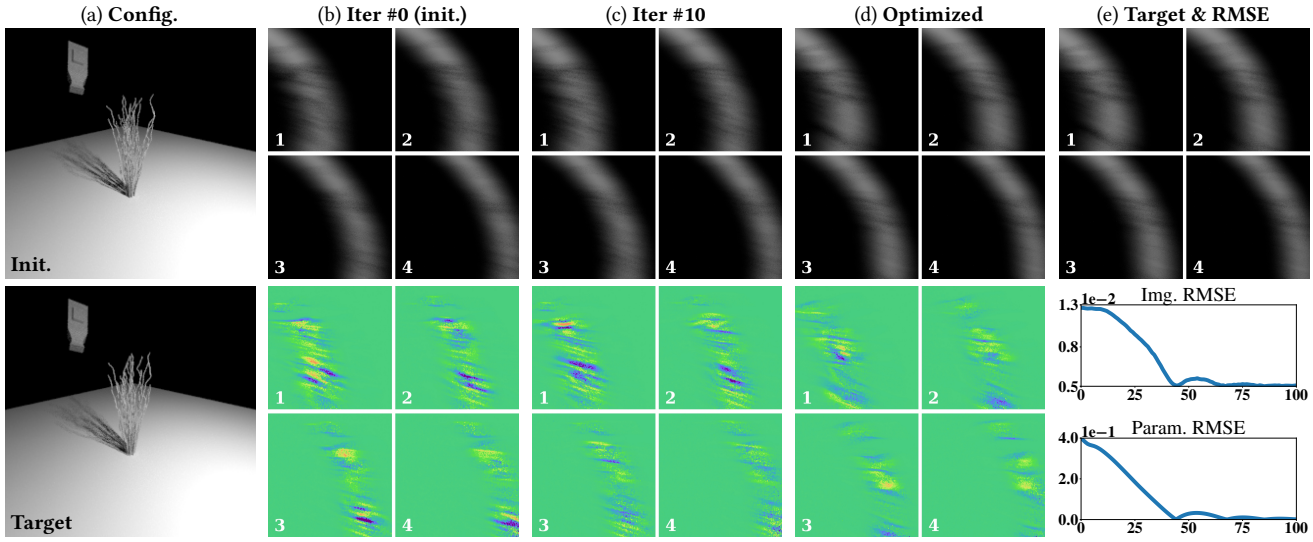

Fig. 12. **Inverse-rendering result (branches):** This example is modeled after the work by Zhang et al. [2020] and involves a tree-like object lit by an area light, casting soft shadows on the ground. We search for the rotation angle of the object, so that the difference (measured in image RMSE) between time-gated renderings of the shadow and the target time-of-flight (ToF) images is minimized. The initial and target configurations are visualized in (a), and four out of five target ToF images (generated using Gaussian path-length importance functions with identical variance and different mean) are shown in (e). Columns (b–d) show four time-gated renderings (marked as 1–4) and the corresponding derivatives produced by our technique during the optimization process.

images with boxcar path-length importance, we jointly optimize the shape and the surface albedo of this object.

- The **tree** example (Figure 17) has a glossy tree model [Telezhkin 2021] lit by an area light from above. Using 14 time-gated renderings and near-delta truncated Gaussian path-length importance, we optimize the horizontal position of the area light. Thanks to our ellipsoidal next-event estimation and antithetic sampling (discussed in §5), we can obtain low-variance derivatives efficiently, allowing the inverse-rendering optimization to converge nicely.

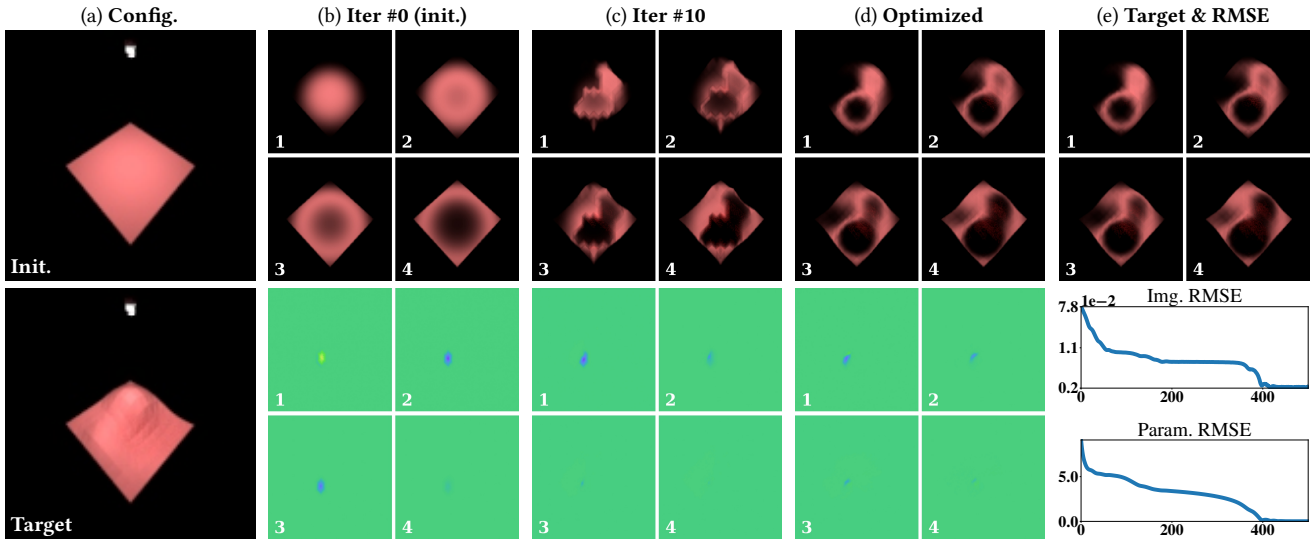

Fig. 13. **Inverse-rendering result (height field):** This example involves a terrain-like object lit by an area light. We search for the vertical position (i.e., height) of each mesh vertex using 12 ToF images with Gaussian path-length importance (among which four are shown). Columns (b–d) show four time-gated renderings (marked as 1–4) and the corresponding derivatives (with respect to one vertex position) produced by our technique during the optimization process.

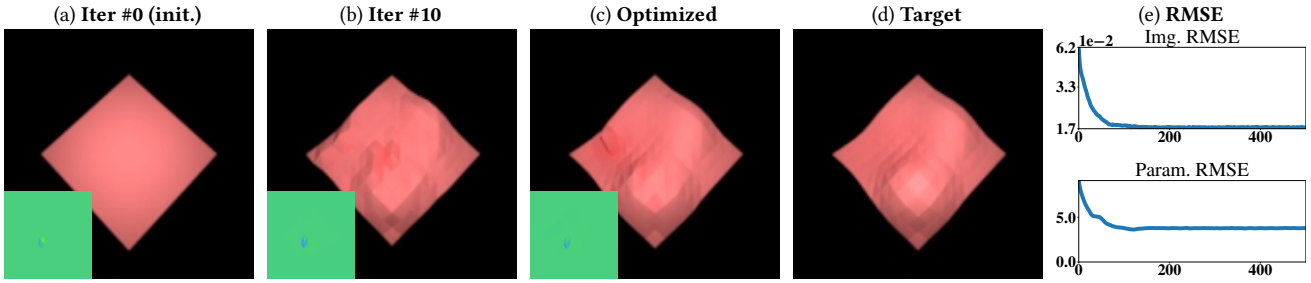

Fig. 14. **Steady-state reconstruction (height field):** When optimizing the height field using a single steady-state image, the problem becomes highly under-constrained, causing the inverse-rendering optimization to get stuck at a local minimum with high parameter RMSE. Each iteration takes approximately equal time (76.04s) compared with the above example using ToF images (70.34s). Derivative images are shown in the insets of columns (a–c).

For each example, we take multiple input time-gated renderings as the target and minimize image root-mean-square errors.

In Figures 12–17, column (a) illustrates the scene configurations of the initial and target states. Columns (b–d) demonstrate the initial, intermediate, and final states of our inverse-rendering optimization, where each column contains four time-gated renderings and the corresponding derivatives (with respect to one of the parameters). Lastly, Column (e) contains four target time-gated images and per-iteration image and parameter RMSE plots. The parameter RMSE information is used only for evaluation (and not for optimization).

In all five examples, derivatives of time-gated images computed with our method allow the inverse-rendering optimizations to converge smoothly. Please refer to Table 2 for performance statistics.

7 DISCUSSION AND CONCLUSION

Limitations and future work. Our theory neglects all volumetric light transport phenomena such as subsurface scattering. Thus, generalizing it to also handle volumetric light transport is an important topic for future research. Additionally, when the path-length importance is a Dirac delta function, as we derive in the supplemental document, the resulting *interior* integral involves second-order derivatives. Developing efficient Monte Carlo estimators for this case can be an interesting future topic.

Currently, our implementation is CPU-based and uses simple forward-mode automatic differentiation (autodiff). Developing new GPU-based systems leveraging reverse-mode autodiff and radiative backpropagation [Nimier-David et al. 2020; Vicini et al. 2021] will offer much better performance and allow efficient handling of millions of parameters.

Another important future direction is applying our differentiable time-gated rendering to real-world large-scale ToF problems involving complex object shapes and materials. Being able to compute arbitrary scene derivatives and handle general light transport phenomena, our theory has made the first step toward a gradient-based

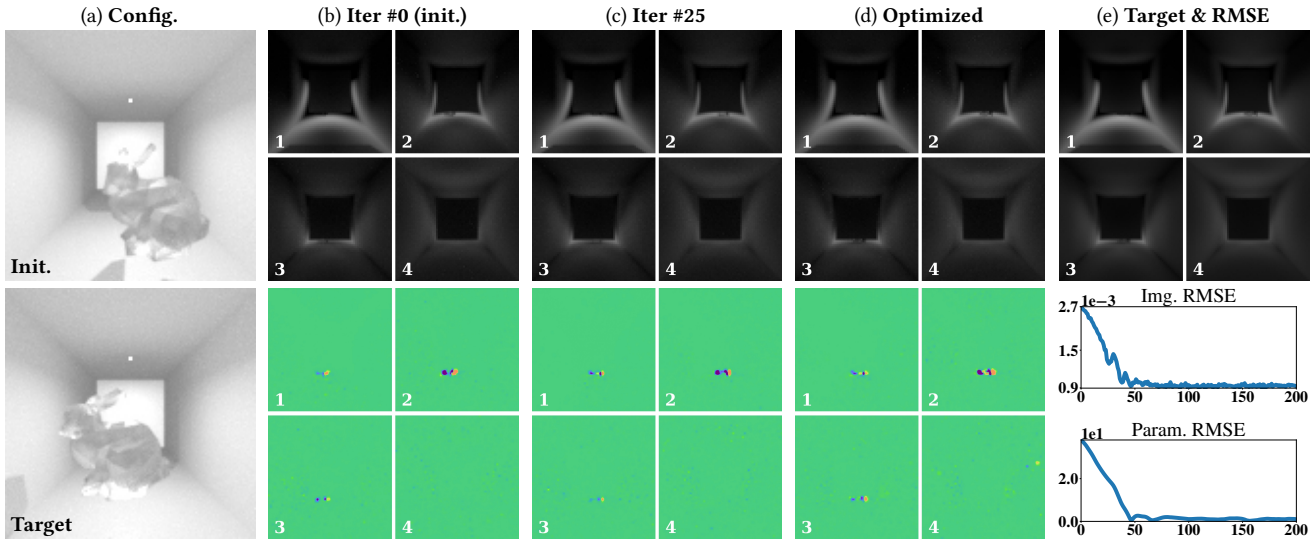

Fig. 15. **Inverse-rendering result (bunny):** This example contains a rough-glass bunny inside a diffuse box. As illustrated in (a), the bunny is occluded by a square patch and not directly visible to the time-of-flight camera. We search for the translation of the bunny (along the x- and y-axis), and four out of 20 target ToF images (using Gaussian functions as the path-length importance) are shown in (e). Columns (b–d) show four time-gated renderings (marked as 1–4) and the corresponding derivatives produced by our technique during the optimization process.

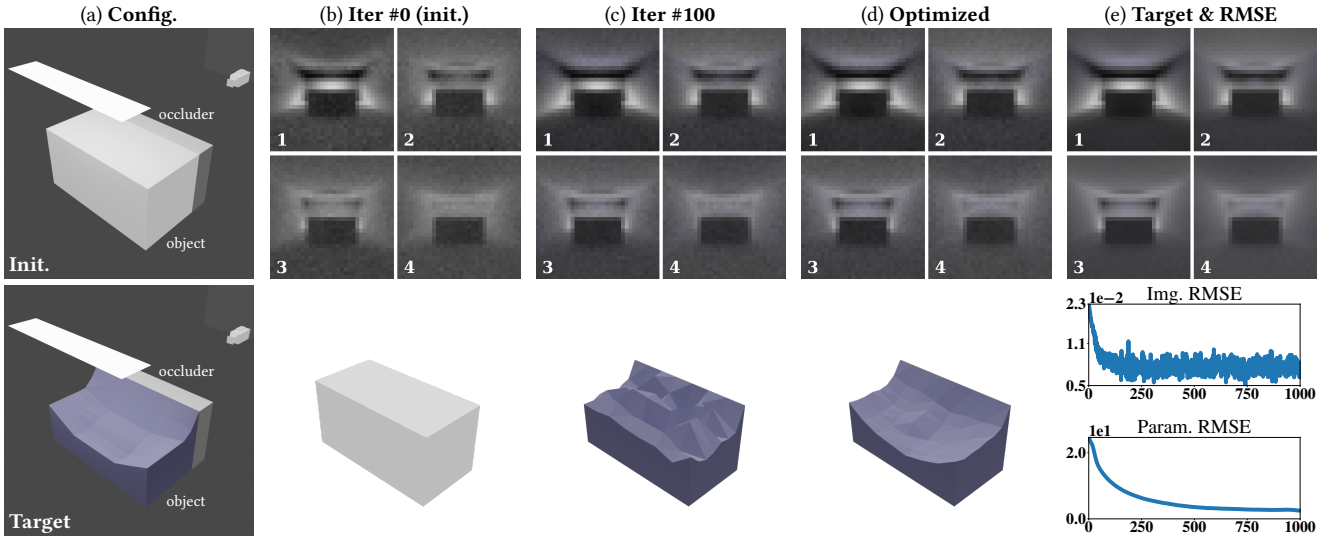

Fig. 16. **Inverse-rendering result (sofa):** This example involves a diffuse sofa-like object that is lit by an area light and not directly visible to the camera. We jointly optimize the shape and surface albedo of this object using 20 ToF images with boxcar path-length importance functions (with four shown). Additionally, columns (b–d) show four time-gated renderings (top) and re-renderings of the object (bottom).

solution to time-of-flight imaging problems, alleviating constraints of existing work such as limited light bounces and low-frequency surface reflectance.

Conclusion. In this paper, we introduced a theory of differentiable time-gated rendering that offers the generality to differentiate time-gated images with respect to arbitrary scene parameters including material optical properties and object geometries. At the core of our theory is the differential time-gated path integral comprised of an *interior* and a visibility-*boundary* component as well as a path-length-*boundary* integral unique to differentiable time-gated rendering. Additionally, we developed new Monte Carlo methods to estimate these integrals in an unbiased and efficient fashion. We validated our derivative estimates by comparing with finite-difference references. Further, we demonstrated the practical usefulness of our

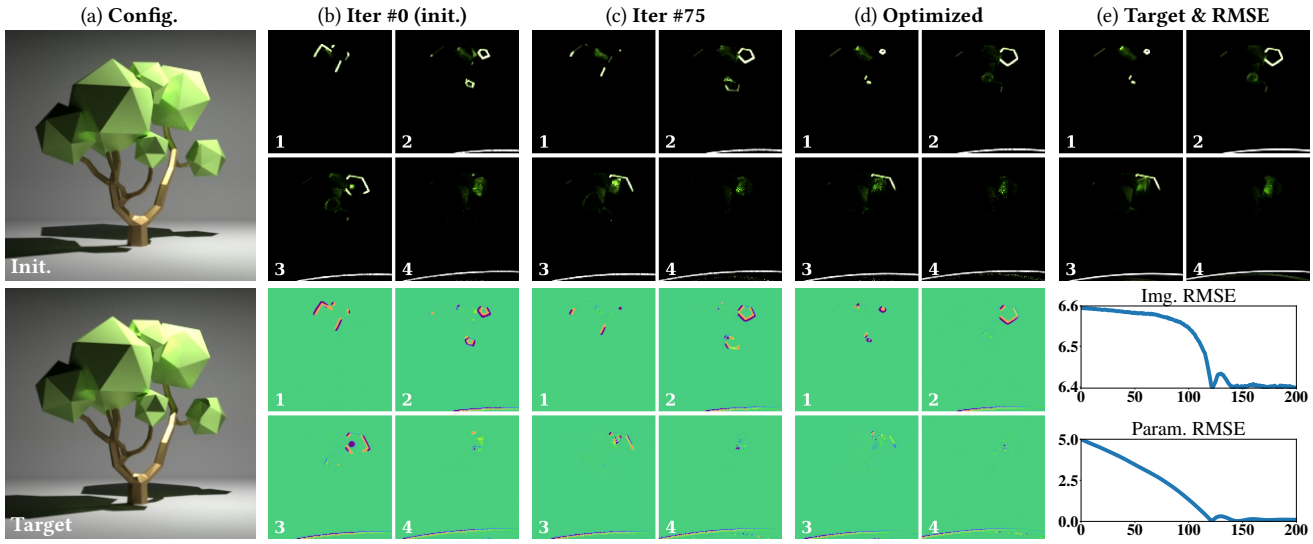

Fig. 17. **Inverse-rendering result (tree):** This example contains a glossy tree model [Telezhkin 2021] lit by a small area light. We search for the horizontal position of the light source, and four of 14 target ToF images (using near-delta truncated Gaussian functions as the path-length importance) are shown in (e). Columns (b–d) show four time-gated renderings (marked as 1–4) and the corresponding derivatives produced by our technique during the optimization process.

theory and algorithms via a few proof-of-concept inverse-rendering examples that mimic real-world time-gated imaging settings.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their comments and suggestions. This work was started while Lifan Wu and Guangyan Cai were students at UCSD, and the work was supported in part by NSF grant 1900927, an NVIDIA Fellowship, and the Ronald L. Graham Chair. We also acknowledge the recent award of NSF grant 2105806.

REFERENCES

Sai Bangaru, Tzu-Mao Li, and Frédo Durand. 2020. Unbiased Warped-Area Sampling for Differentiable Rendering. *ACM Trans. Graph.* 39, 6 (2020), 245:1–245:18.

Subrahmanyan Chandrasekhar. 1960. *Radiative transfer.* Courier Corporation.

Wenzheng Chen, Fangyin Wei, Kiriakos N. Kutulakos, Szymon Rusinkiewicz, and Felix Heide. 2020. Learned feature embeddings for non-line-of-sight imaging and recognition. *ACM Trans. Graph.* 39, 6 (2020).

BB Das, KM Yoo, and RR Alfano. 1993. Ultrafast time-gated imaging in thick tissues: a step toward optical mammography. *Optics letters* 18, 13 (1993), 1092–1094.

Ofer David, Norman S Kopeika, and Boaz Weizer. 2006. Range gated active night vision system for automobiles. *Applied optics* 45, 28 (2006), 7248–7254.

Ioannis Gkioulekas, Anat Levin, and Todd Zickler. 2016. An evaluation of computational imaging techniques for heterogeneous inverse scattering. In *European Conference on Computer Vision*. Springer, 685–701.

Yoav Grauer and Ezri Sonn. 2015. Active gated imaging for automotive safety applications. In *Video Surveillance and Transportation Imaging Applications 2015*, Vol. 9407. International Society for Optics and Photonics, 94070F.

Tobias Gruber, Frank Julca-Aguilar, Mario Bijelic, and Felix Heide. 2019. Gated2depth: Real-time dense lidar from gated images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1506–1516.

Felix Heide, Matthew O'Toole, Kai Zang, David B. Lindell, Steven Diamond, and Gordon Wetzstein. 2019. Non-line-of-sight imaging with partial occluders and surface normals. *ACM Trans. Graph.* 38, 3 (2019), 22:1–22:10.

Felix Heide, Lei Xiao, Wolfgang Heidrich, and Matthias B Hullin. 2014. Diffuse mirrors: 3D reconstruction from diffuse indirect illumination using inexpensive time-of-flight sensors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3222–3229.

Julian Iseringhausen and Matthias B. Hullin. 2020. Non-line-of-sight reconstruction using efficient transient rendering. *ACM Trans. Graph.* 39, 1 (2020), 8:1–8:14.

Adrian Jarabo. 2012. Femto-photography: Visualizing light in motion. *Universidad de Zaragoza* (2012).

Adrian Jarabo and Victor Arellano. 2018. Bidirectional Rendering of Vector Light Transport. *Computer Graphics Forum* (2018).

Adrian Jarabo, Julio Marco, Adolfo Muñoz, Raul Buisan, Wojciech Jarosz, and Diego Gutierrez. 2014. A Framework for Transient Rendering. *ACM Trans. Graph.* 33, 6 (2014), 177:1–177:10.

Adrian Jarabo, Belen Masia, Julio Marco, and Diego Gutierrez. 2017. Recent advances in transient imaging: A computer graphics and vision perspective. *Visual Informatics* 1, 1 (2017), 65–79.

James T. Kajiya. 1986. The rendering equation. *SIGGRAPH Comput. Graph.* 20, 4 (1986), 143–150.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

Ahmed Kirmani, Tyler Hutchison, James Davis, and Ramesh Raskar. 2009. Looking around the corner using transient imaging. In *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 159–166.

Ahmed Kirmani, Tyler Hutchison, James Davis, and Ramesh Raskar. 2011. Looking around the corner using ultrafast transient imaging. *International journal of computer vision* 95, 1 (2011), 13–28.

Joseph R Lakowicz, Henryk Szmacinski, Kazimierz Nowaczyk, Klaus W Berndt, and Michael Johnson. 1992. Fluorescence lifetime imaging. *Analytical biochemistry* 202, 2 (1992), 316–330.

Martin Laurenzis and Andreas Velten. 2014. Nonline-of-sight laser gated viewing of scattered photons. *Optical Engineering* 53, 2 (2014), 023102.

Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. 2018. Differentiable Monte Carlo ray tracing through edge sampling. *ACM Trans. Graph.* 37, 6 (2018), 222:1–222:11.

David B. Lindell, Gordon Wetzstein, and Matthew O'Toole. 2019. Wave-based non-line-of-sight imaging using fast f-k migration. *ACM Trans. Graph.* 38, 4 (2019), 116.

Xiaochun Liu, Sebastian Bauer, and Andreas Velten. 2020. Phasor field diffraction based reconstruction for fast non-line-of-sight imaging systems. *Nature Communications* 11, 1 (2020), 1–13.

Xiaochun Liu, Ibón Guillén, Marco La Manna, Ji Hyun Nam, Syed Azer Reza, Toan Huu Le, Adrian Jarabo, Diego Gutierrez, and Andreas Velten. 2019. Non-line-of-sight imaging using phasor-field virtual wave optics. *Nature* 572, 7771 (2019), 620–623.

Guillaume Loubet, Nicolas Holzschuch, and Wenzel Jakob. 2019. Reparameterizing discontinuous integrands for differentiable rendering. *ACM Trans. Graph.* 38, 6 (2019), 228:1–228:14.

Julio Marco, Ibón Guillén, Wojciech Jarosz, Diego Gutierrez, and Adrian Jarabo. 2019. Progressive transient photon beams. In *Computer Graphics Forum*, Vol. 38. Wiley Online Library, 19–30.
Nikhil Naik, Shuang Zhao, Andreas Velten, Ramesh Raskar, and Kavita Bala. 2011. Single view reflectance capture using multiplexed scattering and time-of-flight imaging. *ACM Trans. Graph.* 30, 6 (2011), 171:1−−171:10.
Merlin Nimier-David, Sébastien Speierer, Benoît Ruiz, and Wenzel Jakob. 2020. Radiative backpropagation: an adjoint method for lightning-fast differentiable rendering. *ACM Trans. Graph.* 39, 4 (2020), 146:1–146:15.
Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. 2019. Mitsuba 2: A retargetable forward and inverse renderer. *ACM Trans. Graph.* 38, 6 (2019), 203:1–203:17.
Matthew O'Toole, Felix Heide, Lei Xiao, Matthias B. Hullin, Wolfgang Heidrich, and Kiriakos N. Kutulakos. 2014. Temporal frequency probing for 5D transient analysis of global light transport. *ACM Trans. Graph.* 33, 4 (2014), 87:1–87:11.
Matthew O'Toole, David B Lindell, and Gordon Wetzstein. 2018. Confocal non-line-of-sight imaging based on the light-cone transform. *Nature* 555, 7696 (2018), 338–341.
Xian Pan, Victor Arellano, and Adrian Jarabo. 2019. Transient instant radiosity for efficient time-resolved global illumination. *Computers & Graphics* 83 (2019), 107–113.
Mark Pauly, Thomas Kollig, and Alexander Keller. 2000. Metropolis light transport for participating media. In *Rendering Techniques 2000*. Springer, 11–22.
Adithya Pediredla, Akshat Dave, and Ashok Veeraraghavan. 2019a. Snlos: Non-line-of-sight scanning through temporal focusing. In *2019 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 1–13.
Adithya Pediredla, Ashok Veeraraghavan, and Ioannis Gkioulekas. 2019b. Ellipsoidal path connections for time-gated rendering. *ACM Trans. Graph.* 38, 4 (2019), 38:1–38:12.
Adithya Kumar Pediredla, Mauro Buttafava, Alberto Tosi, Oliver Cossairt, and Ashok Veeraraghavan. 2017. Reconstructing rooms using photon echoes: A plane based model and reconstruction algorithm for looking around the corner. In *2017 IEEE International Conference on Computational Photography (ICCP)*. 1–12.
Ramesh Raskar, Andreas Velten, Sebastian Bauer, and Tristan Swedish. 2020. Seeing around corners using time of flight. In *ACM SIGGRAPH 2020 Courses*. 1–97.
Osborne Reynolds. 1903. *Papers on mechanical and physical subjects: the sub-mechanics of the universe*. Vol. 3. The University Press.
Guy Satat, Matthew Tancik, and Ramesh Raskar. 2018. Towards photography through realistic fog. In *2018 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 1–10.
Charles Saunders, John Murray-Bruce, and Vivek K Goyal. 2019. Computational periscopy with an ordinary digital camera. *Nature* 565, 7740 (2019), 472–475.
Adam Smith, James Skorupski, and James Davis. 2008. Transient rendering. (2008).
Simon Telezhkin. 2021. Low poly tree 3D model. https://done3d.com/low-poly-tree/
Christos Thrampoulidis, Gal Shulkind, Feihu Xu, William T Freeman, Jeffrey H Shapiro, Antonio Torralba, Franco NC Wong, and Gregory W Wornell. 2018. Exploiting occlusion in non-line-of-sight active imaging. *IEEE Transactions on Computational Imaging* 4, 3 (2018), 419–431.
Chia-Yin Tsai, Kiriakos N Kutulakos, Srinivasa G Narasimhan, and Aswin C Sankaranarayanan. 2017. The geometry of first-returning photons for non-line-of-sight imaging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7216–7224.
Chia-Yin Tsai, Aswin C Sankaranarayanan, and Ioannis Gkioulekas. 2019. Beyond volumetric albedo–a surface optimization framework for non-line-of-sight imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1545–1555.
Eric Veach. 1997. *Robust Monte Carlo methods for light transport simulation*. Vol. 1610. Stanford University PhD thesis.
Delio Vicini, Sébastien Speierer, and Wenzel Jakob. 2021. Path Replay Backpropagation: Differentiating Light Paths using Constant Memory and Linear Time. *ACM Trans. Graph.* 40, 4 (2021), 108:1–108:14.
Shumian Xin, Sotiris Nousias, Kiriakos N Kutulakos, Aswin C Sankaranarayanan, Srinivasa G Narasimhan, and Ioannis Gkioulekas. 2019. A theory of Fermat paths for non-line-of-sight shape reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6800–6809.
Sean I. Young, David B. Lindell, Bernd Girod, David Taubman, and Gordon Wetzstein. 2020. Non-line-of-sight Surface Reconstruction Using the Directional Light-cone Transform. In *Proc. CVPR*.
Tizian Zeltner, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob. 2021. Monte Carlo Estimators for Differential Light Transport. *ACM Trans. Graph.* 40, 4 (2021).
Cheng Zhang, Zhao Dong, Michael Doggett, and Shuang Zhao. 2021a. Antithetic sampling for Monte Carlo differentiable rendering. *ACM Trans. Graph.* 40, 4 (2021), 77:1–77:12.
Cheng Zhang, Bailey Miller, Kai Yan, Ioannis Gkioulekas, and Shuang Zhao. 2020. Path-space differentiable rendering. *ACM Trans. Graph.* 39, 4 (2020), 143:1–143:19.
Cheng Zhang, Lifan Wu, Changxi Zheng, Ioannis Gkioulekas, Ravi Ramamoorthi, and Shuang Zhao. 2019. A differential theory of radiative transfer. *ACM Trans. Graph.* 38, 6 (2019), 227:1–227:16.
Cheng Zhang, Zihan Yu, and Shuang Zhao. 2021b. Path-space differentiable rendering of participating media. *ACM Trans. Graph.* 40, 4 (2021), 76:1–76:15.

A DERIVATION OF THE PATH-LENGTH BOUNDARY COMPONENT

In what follows, we derive the path-length *boundary* component of our differential time-gate path integral of Eq. (13).

Given $\bar{\boldsymbol{p}} = (\boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ with $N \geq 1$,[4] let

$$h(\bar{\boldsymbol{p}}) \coloneqq \int_{\mathcal{B}} W_\tau(\|\bar{\boldsymbol{x}}_+\|)\, \hat{f}(\bar{\boldsymbol{p}}_+)\, \mathrm{d}A(\boldsymbol{q}), \tag{32}$$

where $\bar{\boldsymbol{p}}_+ \coloneqq (\boldsymbol{p}_0, \boldsymbol{q}, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_N)$, and $\bar{\boldsymbol{x}}_+ \coloneqq \bar{\mathrm{x}}(\bar{\boldsymbol{p}}_+, \theta)$.

Differentiating Eq. (32) with respect to the scene parameter θ, according to the Reynolds transport theorem [Reynolds 1903], yields an *interior* integral and a *boundary* one capturing discontinuities of the integrand $W_\tau\, \hat{f}$ with respect to $\boldsymbol{p}_0$. We now examine discontinuities emerging from W_τ since those given by $\hat{f}$ have already been captured by the visibility-*boundary* component in Eq. (13):

$$\int_{\Delta\mathcal{B}[W_\tau]} \Delta W_\tau(\|\bar{\boldsymbol{x}}_+\|)\, \hat{f}(\bar{\boldsymbol{p}}_+)\, v_\tau(\boldsymbol{q})\, \mathrm{d}\ell(\boldsymbol{q}), \tag{33}$$

where $\Delta\mathcal{B}[W_\tau] \subset \mathcal{B}$ denotes the jump discontinuity points of $W_\tau(\cdot; \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ defined in Eq. (20), and $v_\tau(\boldsymbol{q})$ denotes the scalar change rate of $\boldsymbol{q}$ with respect to θ along the normal direction of the discontinuity curve.

Since the discontinuity curve $\Delta\mathcal{B}[W_\tau]$ of $W_\tau(\cdot; \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N) : \mathcal{B} \mapsto \mathbb{R}_{\geq 0}$ are determined by the discontinuities $\Delta\mathbb{R}[W_\tau]$ of the path-length importance $W_\tau : \mathbb{R}_{>0} \mapsto \mathbb{R}_{\geq 0}$, it holds that

$$\Delta\mathcal{B}[W_\tau] = \bigcup_{s \in \Delta\mathbb{R}[W_\tau]} \Delta\mathcal{B}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N), \tag{34}$$

where $\Delta\mathcal{B}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)$ is defined in Eq. (19).

By integrating Eq. (33) over the material path space $\hat{\boldsymbol{\Omega}}$, we obtain the path-length-*boundary* component of Eq. (13):

$$\begin{aligned}
&\sum_{s \in \Delta\mathbb{R}[W_\tau]} \int_{\hat{\Omega}} \int_{\Delta\mathcal{B}(s, \boldsymbol{p}_0, \ldots, \boldsymbol{p}_N)} \Delta W_\tau(\|\bar{\boldsymbol{x}}_+\|)\, \hat{f}(\bar{\boldsymbol{p}}_+)\, v_\tau(\boldsymbol{q})\, \mathrm{d}\ell(\boldsymbol{q})\, \mathrm{d}\mu(\bar{\boldsymbol{p}}) \\
&\quad = \sum_{s \in \Delta\mathbb{R}[W_\tau]} \int_{\partial\hat{\Omega}_\tau(s)} \Delta W_\tau(\|\bar{\boldsymbol{x}}_+\|)\, \hat{f}(\bar{\boldsymbol{p}}_+)\, v_\tau(\boldsymbol{q})\, \mathrm{d}\dot{\mu}_\tau(\bar{\boldsymbol{p}}_+).
\end{aligned} \tag{35}$$

We note that, our derivation above analyzes jump discontinuities of $\boldsymbol{q}$ with $\boldsymbol{p}_0, \ldots, \boldsymbol{p}_N$ fixed. Alternatively, one can also consider discontinuities of any different path vertex with all the others fixed. This would result in a different formula but ultimately identical derivatives.

[4] We focus on the more general case where $N \geq 1$. For the special case with $N = 0$, a similar result can be obtained by letting $\bar{\boldsymbol{p}}_+ \coloneqq (\boldsymbol{q}, \boldsymbol{p}_0)$.