Recent differentiable rendering techniques have become key tools to tackle many inverse problems in graphics and vision. Existing models, however, assume steady-state light transport, i.e., infinite speed of light. While this is a safe assumption for many applications, recent advances in ultrafast imaging leverage the wealth of information that can be extracted from the exact time of flight of light. In this context, physically-based transient rendering allows to efficiently simulate and analyze light transport considering that the speed of light is indeed finite. In this paper, we introduce a novel differentiable transient rendering framework, to help bring the potential of differentiable approaches into the transient regime. To differentiate the transient path integral we need to take into account that scattering events at path vertices are no longer independent; instead, tracking the time of flight of light requires treating such scattering events at path vertices jointly as a multidimensional, evolving manifold. We thus turn to the generalized transport theorem, and introduce a novel correlated importance term, which links the time-integrated contribution of a path to its light throughput, and allows us to handle discontinuities in the light and sensor functions. Last, we present results in several challenging scenarios where the time of flight of light plays an important role such as optimizing indices of refraction, non-line-of-sight tracking with nonplanar relay walls, and non-line-of-sight tracking around two corners.

CCS Concepts: • Computing methodologies → Rendering.

Additional Key Words and Phrases: ray tracing, differentiable rendering, Monte Carlo rendering, physically-based rendering, transient rendering

ACM Reference Format:

1 INTRODUCTION

Physically-based differentiable rendering deals with the computation of the derivatives of radiometric measurements, according to changes in scene parameters (see recent references [Kato et al. 2020; Zhao et al. 2020] for a wide overview of the field). It has recently become a key tool not only for inverse rendering or scene reconstruction problems that require gradient-based optimization, but also to enable the integration of physics-based simulations in machine learning pipelines, computing the loss function in rendering space. While many different approaches have been presented in recent years (e.g., [Li et al. 2018; Zhang et al. 2020, 2019]), all of them assume steady-state configurations, where time delays due to the propagation of light are ignored.

Transient imaging, on the other hand, has already enabled many non-trivial applications in computer vision and scene recognition [Jarabo et al. 2017], such as visualizing light in motion [Heide et al. 2013; Velten et al. 2013], looking around corners [Liu et al. 2019a; O’Toole et al. 2018], looking through turbid media [Heide et al.
Compute the time-resolved radiance derivatives as a function of the scene is currently limited to expensive and fragile finite differences [Gkioulekas et al. 2016; Iseringhausen and Hullin 2020], which hinders the applicability of current transient rendering in inverse problems or machine learning approaches. To overcome this, we introduce \textit{path-space differentiable transient rendering}. Our goal is to differentiate the transient path integral [Jarabo et al. 2014] to help unlock the potential of differential approaches in transient scenarios. We make the observation that, when tracking the time of flight of light, scattering events at path vertices are no longer independent, as assumed in recent work [Zhang et al. 2020]. As a consequence, we are no longer dealing with two-dimensional evolving manifolds, which in turn means that we cannot apply the Reynolds transport relation recursively between path vertices [Zhang et al. 2020].

We thus rely on the \textit{generalized transport theorem} [Seguin and Fried 2014], applied on the \textit{higher-dimensional manifold} that results from all path vertices contributing to the temporal domain, and introduce a novel \textit{correlated importance} term, which links the time-integrated contribution of a path to its light throughput and allows us to handle discontinuous light and sensor functions. From our theoretical formulation, we then demonstrate that, despite the higher dimensionality of transient light transport, our practical formulation of the boundary integral converges to Zhang’s original formulation of differentiable steady-state path tracing [2020]. In that sense, our work can also be seen as a generalization of the existing path-space formulation, lifting the steady-state assumption of independent scattering events while being able to handle discontinuities in the light and sensor functions. In addition, we show how to incorporate our work in a Monte Carlo framework, defining estimators for both the interior and the boundary integrals.

We validate our results against the baseline transient path integral technique and using finite differences. In addition, we demonstrate how our formulation enables several novel applications in challenging scenarios, where the time of flight of light plays an important role and traditional non-line-of-sight (NLOS) imaging has been inapplicable. In particular, we show optimization results for indices of refraction; NLOS tracking with a nonplanar, wavy relay wall; and tracking the motion of a hidden object around two corners. These last two examples are the first demonstrations of NLOS imaging under such challenging scenarios, and demonstrate the potential of differential transient rendering for solving difficult inverse problems. We hope that our code and datasets\footnote{1http://velab.kaist.ac.kr/siggraphasia2021/} will help future developments on the field.

2 RELATED WORK

For a wide overview of the applications of transient imaging, we refer the reader to existing surveys on the topic [Jarabo et al. 2017; Satat et al. 2016]. In the following, we discuss related works that focus on transient and differentiable rendering.

\textit{Transient rendering}. The term “transient rendering” was coined by Smith et al. [2008], who proposed to extend the rendering equation by including the temporal delay of light propagation. Based on this framework, several works proposed transient variations of Monte Carlo rendering [Adam et al. 2016; Ament et al. 2014; Jarabo 2012; Krivánek et al. 2013; O’Toole et al. 2014; Pitts et al. 2014]. Jarabo et al. [2014] presented a practical framework by introducing the \textit{transient path integral}, including an efficient reconstruction technique, as well as a novel sampling strategy by adding indirect shadow vertices, to control path lengths in participating media. This framework was later extended to include vector-based effects such as polarization or fluorescence [Jarabo and Arellano 2018]. Pediredla et al. [2019] introduced ellipsoidal connections for time-gated rendering, also using additional vertices to control path length, but for surfaces instead of media. Other authors have recently focused on increasing efficiency when rendering transient light transport, usually based on simplified models [Chen et al. 2020; Iseringhausen and Hullin 2020; Pan et al. 2019; Tsai et al. 2019], or by means of filtering [Marco et al. 2019]. All these works render time-resolved zeroth-order radiance; in contrast, we develop a differentiable transient path integral, and focus on the derivatives of transient light transport.

\textit{Analysis-by-synthesis in transient imaging}. Several works have leveraged transient rendering for inverse problems, using it as the generative model for gradient descent-based optimization. Iseringhausen and Hullin [2020] used an efficient three-bounce renderer for non-line-of-sight (NLOS) reconstruction based on finite differences, while Tsai et al. [2019] explicitly derived the light transport gradients of hidden surface properties. None of these works account for visibility changes, and are limited to the particular problem of three-bounce NLOS reconstruction. Gkioulekas et al. [2016] analyzed the use of transient rendering in the context of stochastic gradient descent for recovering heterogeneous media. In contrast, our framework is general and does not rely on explicitly computed gradients, while taking into account complex derivatives including singularities produced by, e.g., visibility changes.

\textit{Differentiable rendering}. While several special-purpose differentiable rendering systems have been proposed in the past (e.g., [Chen et al. 2019; Gkioulekas et al. 2016; Kato et al. 2018; Liu et al. 2019b]), in this work we focus on general-purpose differentiable rendering. It was first proposed in OpenDR [Loper and Black 2014], targeting scalability and efficiency; however, it was based on a simple physical model, which reduced its applicability to direct illumination only. Laine et al. [2020] included additional features such as antialiasing or texture filtering, but still limited only to direct illumination. Li et al. [2018] proposed a general-purpose, differentiable, and physically-based rendering model, computing the differential form of the rendering equation with Monte Carlo ray tracing, while taking into account global illumination and geometric discontinuities. The
authors proposed searching for the discontinuities on the integral using edge sampling. The discontinuity handling was later improved by the works of Loubet et al. [2019], who reparameterized the integral on top of the differentiable renderer Mitsuba 2 [Nimier-David et al. 2019] to ease computations at the price of introducing bias, and Bangaru et al. [2020], who combined edge and area sampling for increasing robustness and efficiency. Zhang et al. [2019] extended the differential rendering equation to the volumetric rendering equation, modeling light transport in participating media. These approaches work on the local domain of the rendering equation, which makes it difficult to extend them to bidirectional rendering algorithms. Zhang et al. [2020] introduced a differential path-space light transport integral suitable for bidirectional methods. Our work builds on top of such integral, and generalizes it to time-resolved light transport. Finally, orthogonal to these works, Nimier-David et al. [2020] proposed an adjoint differential light transport formulation that propagates the derivatives from the camera to the light sources, dramatically improving speed and memory requirements.

3 BACKGROUND

We introduce here the main aspects of both the transient path integral and path-space differentiable rendering. Table 1 summarizes the main symbols used throughout the paper.

3.1 Transient Path Integral

The path integral [Veitch 1997] models the intensity $I$ recorded by a virtual sensor as the integral of all light paths $\Omega$ contributing to a pixel, assuming that both the emission and sensor response are orders of magnitude larger than the propagation time of light. Incorporating the temporal domain we obtain the transient path integral [Jarabo et al. 2014] as

$$ I = \int_0^{\infty} \int_{\Omega} f_{\bar{x}}(\bar{x}, t) d\mu(t) d\mu(\bar{x}), $$

where $\Omega$ represents the space of temporal delays, $t = t_0\ldots t_k$ is the sequence of time delays on each vertex, $d\mu(t)$ denotes temporal integration at each vertex, $d\mu(\bar{x})$ is the differential measure, and $\bar{x} = x_0\ldots x_k$ is a path of $k + 1$ vertices. Vertices $x_0$ and $x_k$ lie on the light source and the sensor, respectively. For convenience, we define the path space as $\Omega = \cup_{k=0}^{\infty} \Omega_k$, with $\Omega_k$ being the space of all paths with $k$ vertices. The path contribution $f_{\bar{x}}(\bar{x}, t)$ is given by

$$ f_{\bar{x}}(\bar{x}, t) = L_{\bar{e}}(x_0 \rightarrow x_1, t_0) \mathcal{Z}(\bar{x}, t) W_e(x_{k-1} \rightarrow x_k, t_k + \text{tof}(\bar{x})), $$

where $\text{tof}(\bar{x}) = c^{-1} \sum_{i=0}^{k} |x_i - x_{i-1}|/\eta_i + \sum_{i=0}^{k} t_i$ is the total delay of path $\bar{x}$ (i.e., its time of flight), with $c$ the speed of light and $\eta_i$ the index of refraction of the medium between vertices $x_i$ and $x_{i+1}$. The temporal delay on the emission $L_{\bar{e}}(x_0 \rightarrow x_1, t_0)$ is continuous in real-world applications, ranging from a few femto-seconds in ultrashort lasers to continuous light modulation in time-of-flight cameras. The sensor sensitivity $W_e(x_{k-1} \rightarrow x_k, t_k + \text{tof}(\bar{x}))$ models the temporal response of each specific timestamp, in addition to the spatio-angular response of the pixel. Last, the light throughput $\mathcal{Z}(\bar{x}, t)$ is obtained from the product of the scattering function at inner vertices $\rho$, the geometry function $G$, and the visibility term $V$ for each path segment as

$$ \mathcal{Z}(\bar{x}, t) = \prod_{i=1}^{k} \rho(x_i, t_i) \prod_{i=0}^{k-1} G(x_i \leftrightarrow x_{i+1}) V(x_i \leftrightarrow x_{i+1}), $$

where the time dependency is explicitly incorporated in the scattering function as $\rho(x_i, t_i)$.

3.2 Path-space Differentiable Rendering

Differentiable rendering deals with the computation of the derivatives of the image $I(\theta)$ with respect to a set of parameters $\theta$ describing the scene. Unfortunately, the path space $\Omega(\theta)$ presents a large number of discontinuities due to occlusions and surface discontinuities, resulting into potential interactions between the different parameters; a naive derivation of $\Omega(\theta)$ cannot handle such evolving discontinuities, and therefore require sophisticated methods to handle them [Bangaru et al. 2020; Li et al. 2018; Loubet et al. 2019].

Zhang et al. [2020] observed that, given a fixed set of parameters $\theta$, the vertices $x_{0\ldots k}$ of a path $\bar{x} \in \Omega(\theta)$ lie on a surface manifold $M(\theta) \subset \mathbb{R}^2$ that evolves with $\theta$. In particular, it follows a vector field defined by the local velocities $v(x_{0\ldots k}, \theta) = \nabla_{\theta} x_{0\ldots k}(\theta)$, with

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\bar{x} = x_0\ldots x_k$</td>
<td>Light path of $k + 1$ vertices</td>
</tr>
<tr>
<td>$t = t_1\ldots t_k$</td>
<td>Time delays on $k + 1$ vertices</td>
</tr>
<tr>
<td>$\Omega$</td>
<td>Space of all light paths</td>
</tr>
<tr>
<td>$\Omega_k$</td>
<td>Space of light paths of $k + 1$ vertices</td>
</tr>
<tr>
<td>$\mathcal{Z}$</td>
<td>Space of temporal delays</td>
</tr>
<tr>
<td>$c$</td>
<td>Speed of light in vacuum</td>
</tr>
<tr>
<td>$\eta_i$</td>
<td>Refractive index of the medium between $x_i$ and $x_{i-1}$</td>
</tr>
<tr>
<td>tof $(\bar{x})$</td>
<td>Total delay of path $\bar{x}$</td>
</tr>
<tr>
<td>$L_{\bar{e}}(x_0 \rightarrow x_1, t)$</td>
<td>Light source emission function</td>
</tr>
<tr>
<td>$W_e(x_{k-1} \rightarrow x_k, t)$</td>
<td>Sensor sensitivity function</td>
</tr>
<tr>
<td>$S_e(\bar{x})$</td>
<td>Correlated importance as a function of light path $\bar{x}$</td>
</tr>
<tr>
<td>$\rho(x_i, t_i)$</td>
<td>Scattering function at vertex $x_i$</td>
</tr>
<tr>
<td>$G(x_i \leftrightarrow x_{i+1})$</td>
<td>Geometric function</td>
</tr>
<tr>
<td>$V(x_i \leftrightarrow x_{i+1})$</td>
<td>Visibility function</td>
</tr>
<tr>
<td>$\mathcal{Z}(\bar{x}, t)$</td>
<td>Light throughput, the product of $\rho$, $G$, and $V$</td>
</tr>
</tbody>
</table>

Table 1. Main symbols used in the paper.
\[ x_{0,k}(\theta) \in M(\theta) \] a path vertex for a parameter set \( \theta \) (in the following, we omit the dependence on \( \theta \) on paths and vertices for clarity).

The manifold \( M(\theta) \) can be assumed \( C^0 \)-continuous except in the set \( \partial M(\theta) \) that includes the boundaries of \( M(\theta) \) and the discontinuity curves of \( f(\hat{x}) \), i.e., the jump discontinuity points in \( f(\hat{x}) \), \( \partial M \), and \( \Delta M(f) \). For clarity, in the following we omit the dependence on \( \theta \) of the extended boundary. We denote \( \partial \Omega(\theta) \subset \Omega(\theta) \) the subspace of paths with at least one vertex \( x_{0,k} \in \partial M(\theta) \), i.e., lying on the boundary manifold. Tracking this particular subset of the manifold allows to explicitly account for the derivatives due to discontinuities in the path space.

By following the Reynolds transport theorem [Cermelli et al. 2005] from fluid mechanics, the derivatives of the path integral can be computed by defining the integral on the evolving manifold \( M(\theta) \) as the sum of the so-called interior and boundary integrals, as

\[
\begin{align*}
\frac{d}{d\theta} I(\theta) &= \int_{\Omega} f(\hat{x}) - f(\bar{x}) \sum_{i=0}^{k} \kappa(x_i)V_M(x_i) \, d\mu(\bar{x}) \\
&\quad + \int_{\partial M} \Delta f(\hat{x})' V_{\partial M}(x_i) d\mu_{\partial M}(\bar{x}),
\end{align*}
\]

where \( f(\hat{x}) \) is the normal derivative of \( f(\hat{x}) \) as a function of \( \theta \), \( \Delta f(\hat{x})' \) is the boundary contribution function for a vertex \( x_i \in \partial M \) on the boundary, \( \kappa(x_i) \) is the total curvature at \( x_i \), and \( V_M(x_i) \) and \( V_{\partial M}(x_i) \) are the scalar normal velocities defined as \( V_M(x_i) = \n(x_i) \cdot \nu(x_i) \) with \( \n(x_i) \) the normal vector at \( x_i \). The normal derivative of \( f(\hat{x}) \) is defined as \( \hat{f}(\bar{x}) = f(\hat{x}) - (\nu(\hat{x}) - V_M(\hat{x})) \) grad \( f(\hat{x}) \), with \( \text{grad}_M(f(\hat{x})) \) the manifold gradient of \( f(\hat{x}) \) (see Figure 2.a). Note that for zero tangential velocity of the evolving manifold with respect to the local parameterization we get \( \hat{f}(\bar{x}) = \bar{f}(\bar{x}) \). Figure 2.b depicts the derivatives of the path integral as interior and boundary integrals.

**4 DIFFERENTIABLE TRANSIENT RENDERING**

In this section we obtain the differentiable version of the transient path integral in Equation (1). Following the path-space model proposed by Zhang et al. [2020], we compute two different terms for the path integral, the interior and boundary terms.

In the original steady-state formulation of Zhang and colleagues, all scattering events in path \( \hat{x} \) are considered independent, which allows the authors to apply the two-dimensional Reynolds transport theorem recursively. However, in transient state this is no longer the case; the total time of flight of the path tof(\hat{x}) needs to be taken into account, and as a consequence scattering events are no longer independent. To deal with this and derive our differential transient path integral, we rely on the generalized transport theorem [Seguin and Fried 2014] applied on a higher-dimensional manifold.

**Differential transient path integral.** We first define an intermediate path contribution term \( f_T \) that only depends on spatial variables as

\[
f_T(\hat{x}) = \int_{T} f_T(\hat{x}, t) d\mu(t),
\]

which leads to the following expression for the transient path integral defined in Equation (1):

\[
I = \int_{\Omega} f_T(\hat{x}) d\mu(\bar{x}).
\]

Similar to previous works [Chen et al. 2020; Jarabo et al. 2014; Tsai et al. 2019], we assume that scattering delays in materials (due to e.g., multiple scattering or electromagnetic phase shift) are negligible compared with the temporal resolution of sensors and the propagation delays. This allows us to approximate \( \rho(x_i, t) \approx \rho(x_i) \delta(t_i) \), therefore removing the temporal dependence on the light throughput \( \mathbb{I}(\hat{x}) \) defined in Equation (3). Moreover, the integral domain of Equation (5) no longer depends on scattering delays, and thus it reduces to a one-dimensional time domain. Therefore, when the temporal response of the source \( L_e \) and sensor \( W_e \) are independent, the path contribution \( f_T \) becomes

\[
f_T(\hat{x}) = \mathbb{I}(\hat{x}) \int_{-\infty}^{\infty} L_e(x_0 \rightarrow x_1, t) W_e(x_{k-1} \rightarrow x_k, t + \text{tof}(\hat{x})) dt
\]

\[
= \mathbb{I}(\hat{x}) S_{e}(\hat{x}),
\]

where we introduce a novel correlated importance function \( S_{e}(\hat{x}) \). Note that \( S_{e}(\hat{x}) \) has an implicit dependence on time through \( \text{tof}(\hat{x}) \); this dependence will be relevant when we obtain its derivatives later in this section, since it allows us to handle discontinuous light and sensor functions (see Figure 3).

To compute the derivative of image \( I \) in Equation (6) we need to differentiate a high-dimensional integral, since our evolving manifolds are \( 2(k+1) \)-dimensional in path space \( \Omega \); more generally, \( \Omega \) is an \( m \)-dimensional subset of \( \mathbb{R}^n \), with \( m \) and \( n \) integers such that
\( n > m \). However, due to the influence of \( \text{tof}(\hat{x}) \) in path contribution \( f_T(\hat{x}) \), we cannot separate it into two-dimensional integrals on \( M \) as in previous work [Zhang et al. 2020], and as a consequence we cannot recursively apply Reynolds transport theorem for two-dimensional manifolds.

Instead, we turn to the generalized transport theorem [Seguin and Fried 2014] to compute the derivative of \( I \). As opposed to Reynolds transport theorem, the generalized transport theorem considers a higher-dimensional manifold \( \Omega \) defined as the product of multiple two-dimensional manifolds. Intuitively, each of these low-dimensional manifolds define a light interaction at each path vertex. Applying the generalized transport theorem allows to directly compute the derivative of the full path, leading to our differential transient path integral (see the supplemental document for the complete rigorous derivation):

\[
\frac{d}{d\theta} \int_{\Omega} f_T(\hat{x}) d\mu(\hat{x}) = \int_{\Omega} \left[ \frac{\partial f_T(\hat{x})}{\partial \theta} \right] d\mu(\hat{x}) + \int_{\partial\Omega} \Delta f_T(\hat{x}) V_{\Phi}(\hat{x}) d\mu(\partial\Omega)(\hat{x}),
\]  

(8)

where the first term represents the interior integral, and the second describes the boundary integral. Note that the boundary integral is now defined over the extended boundary \( \partial\Omega \); for clarity on the notation, we omit this dependence. Equation (8) has a similar structure to Zhang et al.’s steady-state differential path integral (4), but the term \( f_T(\hat{x}) \) now takes explicitly into account the temporal delays due to light’s time of flight. Our transient formulation can thus be seen as a generalization of Zhang’s formulation, but not limited to 2D manifolds. In the following, we derive individually the interior and boundary integrals.

**Interior term.** From Equations (7) and (8), the normal derivative of path contribution \( f_T(\hat{x}) \) with respect to \( \theta \) is

\[
\frac{\partial}{\partial \theta} f_T(\hat{x}) = \partial_r(\hat{x}) S_e(\hat{x}) + \partial_x(\hat{x}) S_e(\hat{x}).
\]  

(9)

The normal derivative of the light throughput \( \partial_r(\hat{x}) \) is obtained by the product rule in a similar fashion as Zhang et al.:

\[
\partial_r(\hat{x}) = \sum_{j=1}^{k-1} \left[ \sum_{i=1}^{j} \rho_i \prod_{i<j} G_{i-1} V_i + \sum_{j=1}^{k-1} \rho_j \prod_{i=j}^{k-1} G_i V_j \right] \prod_{i=j}^{k-1} G_i V_i,
\]  

(10)

where \( \rho_i = \rho(x_{i-1} \rightarrow x_i) \) and \( G_i = G(x_i \leftrightarrow x_{i+1}) \), and \( V_i = V(x_i \leftrightarrow x_{i+1}) \), for any index \( i \).

We then compute the normal derivative of our novel term \( S_e(\hat{x}) \) as

\[
\partial_S e(\hat{x}) = \frac{\partial S_e(\hat{x})}{\partial \text{tof}(\hat{x})} + \sum_{i=0,1,k-1,k} \frac{\partial S_e(\hat{x})}{\partial x_i} \frac{\partial}{\partial x_i}.
\]  

(11)

Considering \( S_e \) a function of \( t, x_0, x_1, x_{k-1}, \) and \( x_k \), the partial derivatives are evaluated as follows:

\[
\frac{\partial S_e(x)}{\partial t} = \int_{-\infty}^{\infty} \frac{\partial L_e(x_0, x_l, t')}{\partial t} W_e(x_{k-1}, x_l, t' + t) \left( x_0, x_l, t' \right) d\tau_{\text{tof}(x)}.
\]  

(12)

The normal derivative of the path time of flight is

\[
\text{tof}(\hat{x}) = \sum_{i=1}^{k} \frac{\partial}{\partial x_i} \left[ \|x_i - x_{i-1}\| \right]^2 + \frac{\partial}{\partial x_i} \left[ \|x_i - x_{i-1}\| \right] + \frac{\partial}{\partial x_i} \left[ \|x_i - x_{i-1}\| \right].
\]  

(13)

In both Equations (11) and (13) the derivatives of the vertices \( x_i \) lie on the velocity field of \( M \).

To obtain the final expression for \( f_T(\hat{x}) \), we plug Equation (11) into Equation (9), which yields:

\[
\partial f_T(\hat{x}) = \partial f_s(\hat{x}) + \partial x(\hat{x}) \frac{\partial S_e(\hat{x})}{\partial \text{tof}(\hat{x})} \text{tof}(\hat{x}).
\]  

(14)

where \( \partial f_s(\hat{x}) \) is defined as

\[
\partial f_s(\hat{x}) = \partial_r(\hat{x}) S_e(\hat{x}) + \partial_x(\hat{x}) \sum_{i=0,1,k-1,k} \frac{\partial S_e(x)}{\partial x_i} x_i.
\]  

(15)

The second term in Equation (14) models the temporal derivatives for transient rendering (which do not exist in steady-state differentiable rendering), while \( \partial f_s(\hat{x}) \) models the spatial derivatives of the path contribution, similar to the steady-state differential path integral in Equation (4).

**Boundary term.** The main problem in the second term of Equation (8) (the boundary term) is to determine the domain of integration \( \partial\Omega \) [\( f_T \)], in which we will compute the path contribution \( \Delta f_T(\hat{x}) \), as well as how this domain evolves with \( \theta \) according to the velocity field \( V_{\partial\Omega}(\hat{x}) \). Recall that, similarly to its steady-state counterpart in Equation (4), \( \partial\Omega \) [\( f_T \)] represents the extended boundary of \( f_T \), where we include both the manifold boundaries \( \partial\Omega \) and the discontinuity set \( \Delta\Omega \) [\( f_T \)] as \( \partial\Omega \cup \Delta\Omega \) [\( f_T \)]. For our derivation, we consider the set of discontinuities \( \Delta\Omega \) [\( f_T \)] for paths of length \( k \); it follows that \( \Delta\Omega \) [\( f_T \)] = \( \bigcup_{k=1}^{\infty} \Delta\Omega \) [\( f_T \)].

We summarize here the main results of this derivation, and refer the reader to the supplemental material for the full details.

By the generalized transport theorem, the extended boundary of \( f_T \) is the union of the boundary of manifold \( \Omega_k \) and the discontinuities in the different terms of Equation (7), namely the visibility.
\[ \frac{dI}{d\theta} = \sum_{i=0}^{N_I} \frac{N_I}{N_T} \left( \hat{f}_T(p) \right) p_i(\hat{p}) + \sum_{i=0}^{N_B} \frac{N_B}{N_T} \Delta \hat{f}_T(p) V_{A2B}(\hat{p}) d\mu_{A2B}(\hat{p}). \]
The FD result is rendered with 4,096 spp at continuous (Figure 3), so $\Delta \Omega$ and sensors have a finite temporal response making $S$ the steady-state boundary integral is the existence of the discontinuity term $\Delta \Omega_k$. For most practical situations, both light sources and sensors have a finite temporal response making $S$ continuous (Figure 3), so $\Delta \Omega_k$ vanishes; the transient boundary integral estimation is then similar to its steady-state counterpart, except that we again need to keep track of the path time of flight $\text{tof}(\bar{x}(p_j, \theta))$ for binning.

6 RESULTS AND APPLICATIONS

We have implemented our differentiable transient rendering on top of the code provided by Zhang et al. [2020], by extending the bidirectional path tracer to keep track of the propagation delay of the paths. We compute the interior integral using automatic differentiation on top of next-event estimation and radiance-based multiple importance sampling. For the boundary integral, we use the multidirectional sampling of boundaries, next-event estimation, and the grid-based importance sampling proposed by Zhang et al. For temporal reconstruction we use histogram binning, though more advanced density estimation techniques [Jarabo et al. 2014] could be used without changes in our derivatives estimation. We refer to the supplemental material for pseudocode describing the computation of both the interior and boundary integrals.
We first compare our derivatives against the finite differences (FD) method on two different scenes. To obtain the FD, we use our transient renderer by computing radiance with the same (radiance-based) sampling routines we use for the interior integral. Figure 4, left, shows the Egg scene, with a vertical translation of the light source as the varying parameter. Figure 4, right, shows the Tower scene, where the tower rotates along its vertical axis. In both scenes our method matches the results from finite differences closely. In addition, Figure 5 provides equal-time comparisons, showing how our method provides less noisy results than FD.

We also evaluate in Figure 6 the evolution of variance with the number of samples per pixel. We observe that it follows a trend similar to baseline transient rendering [Jarabo et al. 2014].

We have also validated our scene derivatives when varying several parameters at the same time. Figure 7 shows the transient sequence for the Teapot and its scene derivatives with respect to three parameters. \( \theta_1 \) refers to a translation of the teapot, \( \theta_2 \) to a translation of the light source, and \( \theta_3 \) to the combined motion of the object and the light source. As the absolute difference row shows (amplified by a factor of 1000) \( \frac{\partial I}{\partial \theta_1} + \frac{\partial I}{\partial \theta_2} = \frac{\partial I}{\partial \theta_3} \), when \( \theta_3 \) is the combined motion of \( \theta_1 \) and \( \theta_2 \), which is a general key property of differentiable rendering.

### 6.1 Applications

We begin by presenting the first demonstrations of two challenging NLOS scenarios where the time of flight of light plays a key role. We use ADAM optimizer with a scene-dependent learning rate (see Table 2). First, our Wavy scene in Figure 8 shows a variation of the classic confocal NLOS imaging setup; we lift the requirement of using a planar surface as relay wall, and use a wavy surface instead. In the hidden scene, the cyan bunny moves to a different target position. We jointly optimize the three coordinates \( x, y, \) and \( z \) defining its position. Our sampling resolution is \( 60 \times 60 \times 1200 \).

We include a comparison of our derivative computation and FD equipped with the same optimizer and learning rate. Our results are rendered with 17 and 34 spp for interior and boundary integrals, respectively, while FD results are rendered with \( 4 \times 17 \) spp. FD takes 61 min. per iteration, compared to 36 min. in our approach. Moreover, it can be seen how FD fails to optimize the parameters successfully. The oscillating nature of converge in our method is similar to a general stochastic gradient descent.

Next, we show a particularly difficult NLOS scenario: being able to track an object around two corners. Our Corners scene can be seen in Figure 9, where the hidden object moves along one axis. Despite the extremely challenging conditions of the scene, we are able to provide reasonable optimized results.

Figure 10 shows two different cases of parameter optimization. The Coffee scene (also shown in Figure 1) includes two transparent, colored objects: a glass teapot with reddish tea and an empty blue cup. Using our scene derivatives, we optimize the index of refraction of the tea in the teapot from an arbitrary initial value to the final

### References


ACM Trans. Graph., Vol. 40, No. 6, Article 286. Publication date: December 2021.
Differentiable Transient Rendering

Fig. 7. Validation of our derivatives when varying several parameters at the same time. a) Transient sequence and derivatives. $\theta_1$ corresponds to the translation of the teapot, $\theta_2$ to the translation of the light source, and $\theta_3$ to the combined translation of both teapot and light source. The absolute difference ($\times 1000$) shows how the sum of scene derivatives w.r.t. $\theta_1$ and $\theta_2$ is identical to the scene derivatives w.r.t. $\theta_3$. (b) Radiance and absolute difference plots. Dotted vertical lines indicate the exact frames shown above. The scene consists of 600 transient frames with an exposure of 5.0 ns. Our results are rendered with 683+1,382 spp (interior + boundary integrals, resp.) taking 94.7s per frame. The FD result is rendered with $4 \times 683$ spp taking 173.4s per frame.

target value. Last, the Stars scene is composed of two translucent, colored stars. The red star rotates around one axis, while the index of refraction of the blue star changes. In both scenes, the streak images show how changing the index of refraction affects the time of flight of light through the transparent objects.

Figure 10 also includes and equal-sample comparison with Zhang et al.’s steady-state renderer [2020]. It can be seen how the index of refraction converges faster with our method in the Coffee scene, while for Stars the steady-state approach fails to converge. Table 2 summarizes the main information for all scenes shown. Please refer to the supplemental video for the full transient sequences.

Fig. 8. Three-parameter optimization in an NLOS tracking scenario with a wavy relay wall. (a) The hidden bunny moves to the target position, which implies changes in three parameters: $\theta_1 = x$, $\theta_2 = y$, and $\theta_3 = z$. The yellow dots illustrate sampled positions. (b) RMSE of rendered transient images (top), as well as residuals for each parameter obtained by FD (middle) and our method (bottom). (c) Color-coded transient results: initial, optimized parameters after 69 iterations using our method, and target. (d) Streak images in $x - t$ for the coordinates indicated by the magenta line in (a).

Fig. 9. NLOS tracking around two corners. (a) Scene setup, where the hidden object moves along one axis. (b) RMSE of the image and residual of the optimized parameter. (c) Streak images in $x - t$ for the coordinates indicated by the magenta line in (a) for the initial situation, optimized parameter after 159 iterations, and target. Even in this challenging scenario, our method converges with just a small residual bias.
We have presented a framework for differential transient rendering, which allows us to compute derivatives of the rendering image with respect to scene parameters. The cost of computing these derivatives depends on the scene, due to the complexity of handling discontinuities beyond the geometric visibility term. As expected, when our time-dependent terms disappear, the temporal boundary in path space vanishes and our differential transient path integral converges to its steady-state counterpart.

**Relation to the steady-state differential path integral.** Our work can be seen as a generalization of the steady-state differential path integral [Zhang et al. 2020] to higher-dimensional domains, and handling discontinuities beyond the geometric visibility term. As expected, when our time-dependent terms disappear, the temporal boundary in path space vanishes and our differential transient path integral converges to its steady-state counterpart.

**Limitations and future work.** Several exciting avenues of future work lie ahead. From a theoretical point of view, it would be interesting to include Dirac delta functions in the temporal domain for both the light and the sensor. From a more practical perspective, assuming non-negligible scattering delays would allow handling fluorescent materials [Gutierrez et al. 2008], thus extending the applicability of our method to time-resolved fluorescence lifetime imaging (FLI) [Satat et al. 2015]. This would require re-introducing the temporal dimension on the paths throughput, as well as the scattering delays inside the path time of flight. Extending our differentiable transient framework to participating media [Zhang et al. 2021] should be possible, adding indirect shadow vertices to control the path duration during sampling [Jarabo et al. 2014]. Exploiting this additional degree of freedom is a promising avenue for improving the applicability of our method to time-resolved fluorescence lifetime imaging (FLI) [Satat et al. 2015].

We have shown how to incorporate our framework in a Monte Carlo renderer, and demonstrated its application in several challenging scenarios. The cost of computing our derivatives depends on the scene, due to next-event estimation and importance sampling over edges in the boundary integral (similar to Zhang’s method). In the worst case (Tower) the overhead is 72% with respect to standard transient rendering, whereas for our NLOS results (Wavy and Corners) it is just 10%. Since a finite differences approach scales with the number of optimized parameters, our method is faster for all scenes shown.

**7 DISCUSSION AND CONCLUSIONS**

Table 2. Rendering and optimization data for our application scenarios. Since there is no geometry change in the Coffee scene, we do not need to perform boundary integral for this scene. Note that Wavy and Corners scenes requires high spp due to indirect-dominant NLOS scenario.

<table>
<thead>
<tr>
<th>Scene</th>
<th>Coffee</th>
<th>Stars</th>
<th>Wavy</th>
<th>Corners</th>
</tr>
</thead>
<tbody>
<tr>
<td># spp (interior)</td>
<td>41</td>
<td>1</td>
<td>17</td>
<td>85</td>
</tr>
<tr>
<td># spp (boundary)</td>
<td>0</td>
<td>4</td>
<td>34</td>
<td>85</td>
</tr>
<tr>
<td># frames</td>
<td>250</td>
<td>800</td>
<td>1200</td>
<td>1200</td>
</tr>
<tr>
<td># iter.</td>
<td>19</td>
<td>129</td>
<td>69</td>
<td>159</td>
</tr>
<tr>
<td>Time per iter.</td>
<td>28.5min.</td>
<td>5.7min.</td>
<td>36min.</td>
<td>32.8min.</td>
</tr>
<tr>
<td>Learning rate</td>
<td>0.07</td>
<td>0.07</td>
<td>0.07</td>
<td>40.0</td>
</tr>
</tbody>
</table>

Fig. 10. Two examples of parameter optimization involving indices of refraction. Left: The same Coffee scene as in Figure 1, depicting a glass teapot with red tea. We optimize the index of refraction of the tea \( \theta_1 \) from an arbitrary value (1.33) to the target value (1.41). Right: Stars scene showing two translucent glass stars. \( \theta_1 \) represents the single-axis rotation of the red star while \( \theta_2 \) represents the index of refraction of the blue star, changing from 1.5 to 2.5. For both scenes: (a) The top row shows steady-state images for the initial, optimized, and target parameters, as well as the difference between the last two. The bottom row shows the respective \( x - t \) streak images, corresponding to the magenta line. It can be observed how changing the index of refraction affects not only the BSDF of the object, but also the time of flight of light. (b) RMSE of the image and evolution of the residuals for each parameter, as well as equal-sample comparison of our results with Zhang et al.’s steady-state renderer [2020]. Although the error scales are not directly comparable, the optimization of the index of refraction in the steady-state approach converges more slowly in the Coffee scene, while failing to converge in Stars.
full transient sequence. Computing the derivatives in more complex scenarios with potentially thousands of parameters would benefit from faster differentiation strategies, which could be obtained by combining our work with recent advances on efficient backpropagation of derivatives [Nienert-David et al. 2020; Vicini et al. 2021].

ACKNOWLEDGMENTS

We want to thank other lab members for proofreading and the reviewers for their feedback. M. H. Kim acknowledges that this work is supported by Samsung Research Funding & Incubation Center of Samsung Electronics (SRFC-TT2001-04). D. Gutiérrez and A. Jarabo acknowledge additional support by the European Research Council (ERC) under the EU Horizon 2020 research and innovation program (project CHAMELEON, grant No 682080), the EU MSCA-ITN program (project PRIME, grant No 956585) and the Spanish Ministry of Science and Innovation (project PID2019-105046GB-I00).

REFERENCES


Felix Heide, Lei Xiao, Andreas Kolb, Matthias B Hulin, and Wolfgang Heidrich. 2014. Imaging in scattering media using correlation image sensors and sparse convolu-


Tech Report UCSC-SOE-08-28. School of Engineering, University of California, Santa Cruz.


