Computers & Graphics 107 (2022) 84–92

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on CEIG 2022

Non-line-of-sight transient rendering

Diego Royo [a,*], Jorge García [a], Adolfo Muñoz [a], Adrian Jarabo [b]

[a] Universidad de Zaragoza, I3A, Zaragoza, 50017, Spain
[b] Meta Reality Labs, Zürich, Switzerland

ARTICLE INFO

Article history:
Received 12 May 2022
Received in revised form 1 July 2022
Accepted 4 July 2022
Available online 11 July 2022

Keywords:
Transient rendering
Light transport
Non-line-of-sight imaging

ABSTRACT

The capture and analysis of *light in flight*, or light in transient state, has enabled applications such as range imaging, reflectance estimation and especially non-line-of-sight (NLOS) imaging. For this last case, hidden geometry can be reconstructed using time-resolved measurements of indirect diffuse light emitted by a laser. Transient rendering is a key tool for developing such new applications, significantly more challenging than its steady-state counterpart. In this work, we introduce a set of simple yet effective subpath sampling techniques targeting transient light transport simulation in occluded scenes. We analyze the usual capture setups of NLOS scenes, where both the camera and light sources are focused on particular points in the scene. Also, the hidden geometry can be difficult to sample using conventional techniques. We leverage that configuration to reduce the integration path space. We implement our techniques in a modified version of Mitsuba 2 adapted for transient light transport, allowing us to support parallelization, polarization, and differentiable rendering.

1. Introduction

Transient imaging aims to capture and analyze how light propagates through a scene with an ultra-high temporal resolution, reaching the order of picoseconds, breaking the common assumption of the infinite speed of light. This extended dimensionality on the temporal domain provides a richer description of how light propagates and interacts with matter throughout the scene, and in consequence about the scene itself [1,2].

As such, transient imaging has had a great impact in the area of computer graphics and computer vision, with applications such as material classification [3] and recovery [4], vision through scattering media [5,6], visualizing light in motion [7], or reconstruction of partially or fully occluded geometry [8–11]. In this work, we focus on the latter topic, the so-called *non-line-of-sight (NLOS) imaging* problem [12].

The goal of NLOS imaging is to reconstruct a fully or partially occluded scene by measuring the back-scattering of the hidden scene in a secondary visible surface (Fig. 1). This is a fundamental problem in many fields, including autonomous driving, medical imaging, space exploration, and others. Unfortunately, transient imaging, in general, requires expensive hardware, including pulsed illumination and ultrafast cameras, difficult to calibrate and operate. *Transient light transport simulation* emerges as an alternative tool for developing, prototyping and testing such systems, before building them.

However, time-resolved simulation (i.e. transient rendering) is significantly more expensive than traditional steady-state rendering: Instead of generating a 2D image $I(x, y)$, we generate a 3D video of the scene $I(x, y, t)$ that records the scene over time t (see Fig. 2). This requires several orders of magnitude more samples for reducing variance and introduces an additional challenge: The light paths generated with classic steady-state sampling techniques are suboptimal for transient light transport.

In this work, we build upon the transient path integral formulation proposed by Jarabo and colleagues [13] and develop a new set of sampling techniques particularly targeted to NLOS configurations (Fig. 1). Our contributions can be summarized as follows:

- We propose two simple yet effective techniques for generating both the eye and light subpaths, that generate intermediate *virtual* irradiance meters and light sources.
- We also add an importance sampling technique that focuses on finding the hidden geometry, as it can be difficult using conventional sampling techniques. Together, these three techniques improve the convergence of NLOS transient path tracing with little cost.
- We implement our techniques inside a custom transient-aware version of Mitsuba 2 [14]. It provides many useful features such as CPU and GPU parallelism, simulation of the polarization of the light, and could be used for differentiable transient rendering. We compare our results with a reference implementation [13], with lower computational complexity, faster convergence and lower rendering times.

* Corresponding author.
E-mail address: droyo@unizar.es (D. Royo).

https://doi.org/10.1016/j.cag.2022.07.003

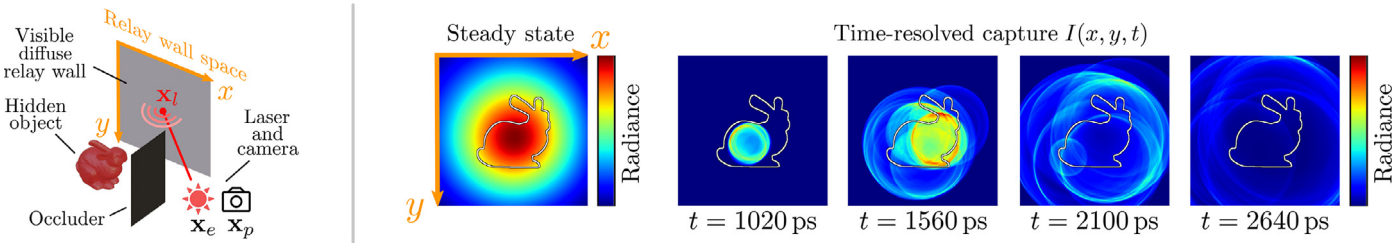

Fig. 1. The non-line-of-sight (NLOS) imaging problem aims to recover 3D geometry of hidden objects. Some methods use ultra-fast lasers and detectors for this task, exploiting the time-resolved properties of light propagation through the scene seen in the relay wall space. We simulate a data capture setup (left): a laser pulse is emitted towards a visible diffuse wall, scattering illuminating the occluded geometry. Light reflects back to the relay wall and is captured by the camera sensor (right).

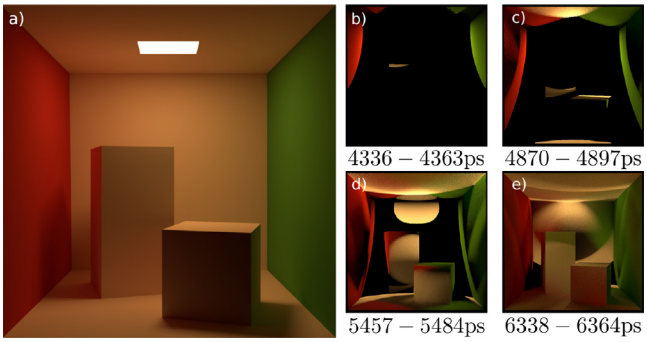

Fig. 2. *Cornell box* scene, with two diffuse boxes and a small area light. Images show (a) steady-state and (b–e) transient renders. Adding all the time-resolved frames of the video achieves the steady-state image in much less than a millisecond.

- Finally, we demonstrate the usefulness of our data by using it to reconstruct NLOS scenes using the phasor fields framework proposed by Liu et al. [10].

2. Related work

Time-resolved imaging. Ultra-fast imaging has found direct applications in computer graphics and vision [15]. Originally, Raskar and Davis [16] introduced a basic theoretical framework for transient light transport analysis. This would later lead to applications such as range sensing [17,18], light transport decomposition [1], vision through turbid media [5,6], reconstruction of hidden geometry (NLOS imaging) [19,8] or reflectance acquisition [4]. The seminal femto-photography technique presented by Velten et al. [20,7] used a streak camera to capture time-resolved videos with an effective exposure time of one picosecond per frame. Heide et al. [21] proposed a cheaper setup using Photonic Mixing Devices (PMDs), which sacrifices temporal and spatial resolution. Other setups include gated sensors [22]. More recently, many have embraced the use of single-photon avalanche diodes (SPADs) [23,24], as they can detect the time of arrival of individual photons with a temporal precision of tens of picoseconds, and can be used at long distances or under ambient lighting. Using simulation, our rendering system can work with arbitrary spatial and temporal resolutions.

NLOS imaging. One of the most challenging applications of time-resolved imaging is non-line-of-sight (NLOS) imaging. We distinguish between two categories: passive and active. Passive methods rely on environment illumination to localize or estimate rough motion and structure [25,26]. We focus on active methods that inject light into the NLOS scene in a controlled way by illuminating an intermediate relay wall. There exist several different works leveraging active sensing of the image for NLOS reconstruction, based on filtered backprojection [8,23,22,9], optimization [27–29], deconvolution [11,30], deep learning [31], or by posing the problem as a virtual waves propagation problem [10].

Transient rendering. This term was originally used by Smith et al. [32]. The authors propose to extend the rendering equation as a recursive operator by incorporating the time delays caused by light propagation. Based on their contribution, other works emerged that proposed transient variations of Monte Carlo rendering [33,34]. Other authors focused on the modeling and simulation of Photonic Mixer Devices (PMDs) [35], a type of time-of-flight sensor. Later, Jarabo et al. [13] introduced the transient path integral, which we base our work on, that includes propagation and scattering temporal delays.

On top of such formulation, several steady-state rendering algorithms have been adapted, such as bidirectional path tracing [13] or progressive photon beams [36], as well as simplified models [37,38] or extensions to include polarization [39], or also include fluorescence [40]. The implementation of our technique is an extension of the Mitsuba2 library [14] so it can also account for polarization.

These ideas have also been applied to the NLOS domain: Hullin [41] and Heide et al. [42] simulate transient light transport to speed up the expensive data capture. Our proposed technique is also tailored towards NLOS setups, but mainly focused on noise reduction for pure transient rendering. Other authors avoid accounting for paths with more than three bounces, simplifying the path integral to obtain efficient results for learning-based reconstruction [28,29]. Our method is not limited in this way. Rendering longer paths could allow to tackle more challenging problems, such as seeing behind two corners.

Recently, Yi et al. introduced a novel framework able to differentiate the transient path integral [43], with applications such as NLOS tracking using non-planar relay walls or around two corners. While untested, our transient rendering implementation is built on top of Mitsuba2 which is differentiable.

Last, another approach for efficient transient rendering with Monte-Carlo techniques is through sophisticated importance sampling techniques that account for the temporal profile of light transport, both in participating media [13] and surfaces [44]. Our core contribution lies within this domain: we develop a new set of simple but effective importance sampling techniques for transient rendering, which are specific to NLOS setups, and are particularly challenging for more traditional sampling schemes.

3. Transient path tracing

The goal of a transient rendering algorithm is to extend the idea of a 2D image by adding a third, temporal dimension. Thus, obtaining an image $I(x, y, t)$ that contains the radiance of each pixel (x, y) over time t. If the light emission and geometry do not change, steady-state is achieved after enough time with

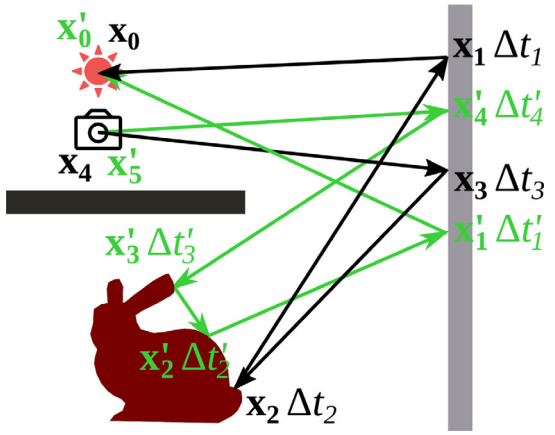

Fig. 3. Two example paths $\bar{\mathbf{x}}, \bar{\mathbf{x}}' \in \Omega$ (black, green) that join the camera with a light source, with three and four bounces on the geometry respectively. Extending the path integral formulation to transient state adds the propagation delays between points, and the possible scattering delays at each point $\Delta t_j, \Delta t'_j \in \Delta T$.

$\partial I / \partial t = 0$. In a typical scene, steady-state is achieved in less than a millisecond (see Fig. 2). As this is smaller than the exposure time of a conventional camera, it is reasonable to assume the speed of light to be infinite in a conventional path tracer. However, this is not our case. For this reason, we use the transient path tracing algorithm proposed by Jarabo et al. [13], along with a histogram-based density estimation technique to generate the final $I(x, y, t)$ result. In this section, we describe the general background which we base our contributions on.

Temporal delays. First, we analyze the behavior of light in flight. Suppose an object emits a ray of light at time t_0, bouncing k times on the scene until it reaches the camera at time t_k. The duration of the light path is determined by the propagation times (i.e. time of flight) between surface intersections from points $\mathbf{x}_j$ to $\mathbf{x}_{j+1}$, noted as $\text{tof}(\mathbf{x}_j \leftrightarrow \mathbf{x}_{j+1})$, plus the microscopic temporal delays when interacting with each of the surfaces Δt_j. In most cases, these delays are negligible even when compared to propagation times. The time of arrival to the camera sensor can be calculated as

$$t_i^- = \sum_{j=0}^{i-1} \left(\text{tof}(\mathbf{x}_j \leftrightarrow \mathbf{x}_{j+1}) + \Delta t_j\right), \quad t_i = t_i^- + \Delta t_i, \tag{1}$$

where $\text{tof}(\cdot)$ depends on the medium's index of refraction η. Assuming it is constant between bounces, it can be computed as $\text{tof}(\mathbf{x}_j \leftrightarrow \mathbf{x}_{j+1}) = \|\mathbf{x}_j - \mathbf{x}_{j+1}\| \eta / c$ with c being the speed of light.

Transient path integral. To obtain the value of a pixel $I(x, y, t)$ we need to look at all the paths that reach pixel (x, y) at time t. A path $\bar{\mathbf{x}}$ is defined with the list of scene points that it travels through, $\bar{\mathbf{x}} = \mathbf{x}_0 \dots \mathbf{x}_k$, where $\mathbf{x}_0$ is the emitter and $\mathbf{x}_k$ is the camera position, as shown in Fig. 3. Note that the exact same path can propagate light at different time instants if the emission time t_0 is different (e.g. multiple pulses or one longer pulse) or if the scattering delays $\Delta \mathbf{x}_j$ are different (e.g. fluorescent materials). These delays are defined as $\bar{\Delta t} = \Delta t_0 \dots \Delta t_{k-1}$. More formally, the value of $I(x, y, t)$ is obtained using the transient path integral:

$$I(x, y, t) = \int_{\Omega(x,y)} \int_{\Delta T(\bar{\mathbf{x}}, t)} f(\bar{\mathbf{x}}, \bar{\Delta t})\, d\mu(\bar{\Delta t})\, d\mu(\bar{\mathbf{x}}), \tag{2}$$

where $f(\cdot)$ is the light measured for that specific path and scattering delays, which is integrated over the whole path space Ω and delay space ΔT. It does not integrate over individual vertices $\mathbf{x}_0 \dots \mathbf{x}_k$: the differential measure $d\mu(\bar{\mathbf{x}})$ denotes area integration for surface vertices and volume integration for vertices in participating media. Similarly, $d\mu(\bar{\Delta t})$ denotes temporal integration at each path vertex, which could account for additional phenomena e.g. scattering delays that change based on the index of refraction. Finally, $\Omega(x, y)$ and $\Delta T(\bar{\mathbf{x}}, t)$ denote paths that pass through pixel (x, y) and arrive at time t respectively. These paths might be none, due to the propagation delays and t. Note that in a conventional path tracer we do not need to integrate over ΔT: the propagation speed is assumed to be infinite, so the propagation and scattering delays for each path are zero and thus $f(\cdot)$ only depends on $\bar{\mathbf{x}}$.

For our case, the time-resolved path contribution function $f(\bar{\mathbf{x}}, \bar{\Delta t})$ accounts for light emission L_e, path throughput $\mathfrak{T}$ and sensor importance W_e:

$$f(\bar{\mathbf{x}}, \bar{\Delta t}) = L_e(\mathbf{x}_0 \rightarrow \mathbf{x}_1, \Delta t_0)\, \mathfrak{T}(\bar{\mathbf{x}}, \bar{\Delta t})\, W_e(\mathbf{x}_{k-1} \rightarrow \mathbf{x}_k, \Delta t_k). \tag{3}$$

The temporal sensor importance W_e defines spatial, angular and time-based sensitivity. Likewise, L_e models light sources using Δt_0 to control the emission instant and duration (e.g. Gaussian or delta laser pulse). The transient path throughput is defined as

$$\mathfrak{T}(\bar{\mathbf{x}}, \bar{\Delta t}) = \left[\prod_{i=1}^{k-1} \rho(\mathbf{x}_i, \Delta t_i)\right] \left[\prod_{i=0}^{k-1} G(\mathbf{x}_i, \mathbf{x}_{i+1})\, V(\mathbf{x}_i, \mathbf{x}_{i+1})\right], \tag{4}$$

where $G(\mathbf{x}_i, \mathbf{x}_{i+1})$ is the geometry term modeling the effects of inverse square falloff and Lambert's cosine law, and $V(\mathbf{x}_i, \mathbf{x}_{i+1})$ is the visibility term between $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$. The scattering kernel ρ, which defines how much light is scattered at each vertex, includes a temporal delay parameter Δt_i to account for potential time delays at each scattering vertex $\mathbf{x}_i$. Note that there are not any other time-related parameters, as we assume that the geometry is stationary, at least relative to the speed of light.

Monte Carlo path tracing. Providing an analytical solution to Eq. (2) is impossible due to the complexity of the search space for a scene. One of the most common solutions to this is to use numerical integration methods such as a Monte Carlo estimator:

$$I \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(\bar{\mathbf{x}}_i, \bar{\Delta t}_i)}{p(\bar{\mathbf{x}}_i, \bar{\Delta t}_i)}. \tag{5}$$

In other words, we generate N random paths $(\bar{\mathbf{x}}_i, \bar{\Delta t}_i)$, drawn from a spatio-temporal probability distribution $p(\cdot)$, which is defined by the sampling strategy for path and scattering delays. As discussed by Jarabo et al. [13], the different natures of the propagation delays in $\bar{\mathbf{x}}$ and the scattering delays in $\bar{\Delta t}$ make challenging to find samples $f(\bar{\mathbf{x}}_i, \bar{\Delta t}_i)$ that have a non-zero contribution. This becomes especially important when either Δt_i, L_e or W_e approach delta functions. For this reason, we reuse all generated samples along the whole temporal domain, by using histogram density estimation to reconstruct the transient image $I(x, y, t)$.

4. Non-line-of-sight path tracing

In this section, we introduce our contributions by adapting the transient path tracing algorithm to simulate NLOS data capture. The setup described in Fig. 1 uses a single laser beam to generate a time-resolved image $I(x_s, y_s, t)$, where (x_s, y_s) parametrize the 2D space of the visible surfaces of the camera (the relay wall). We define this 2D-to-3D map as $\mathcal{M}_s(x_s, y_s) \rightarrow \mathbf{x}_s$, with $\mathbf{x}_s$ being a 3D point visible from the camera.

A common setup repeats this measurement for multiple laser positions on the visible surfaces, which results in a five-dimensional signal $I(x_l, y_l, x_s, y_s, t)$, with (x_l, y_l) parametrizing the 2D space of laser positions. For simplicity, we will refer to the three-dimensional case $I(x_s, y_s, t)$, since extending to five dimensions just requires repeating the same experiment, changing the

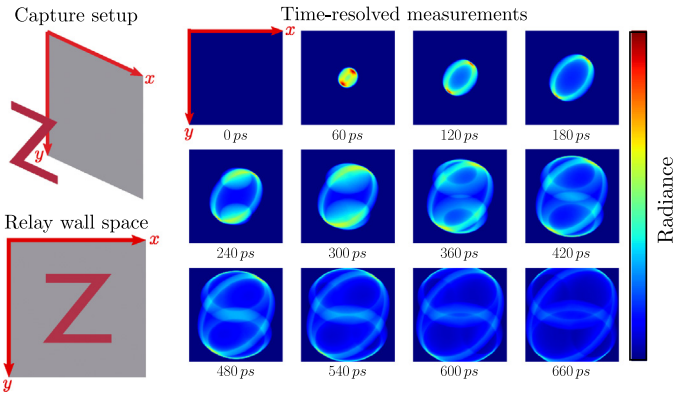

Fig. 4. Transient image of a *Z letter* captured in an NLOS setup. Each time instant shows the radiance at each point, reflected by the relay wall and captured by the sensor with 10 ps of exposure time.

orientation of the laser. Analogous to $\mathcal{M}_s(x_s, y_s)$, we define this mapping as $\mathcal{M}_l(x_l, y_l) \rightarrow \mathbf{x}_l$.

Our implementation is based on a forward path tracer, in which the paths start from the sensor and each path vertex is stochastically selected from the previous one. The probability distribution corresponding to a forward path tracer (the denominator in Eq. (5)) with delta scattering delays $\bar{\Delta t}_i$ can be expressed as:

$$p(\bar{\mathbf{x}}, \Delta t) = p(\mathbf{x}_k) \prod_{i=0}^{k-1} p(\mathbf{x}_i | \mathbf{x}_{i+1}), \tag{6}$$

as the path is traced from the camera $\mathbf{x}_k$ to the light source $\mathbf{x}_0$. Our contributions are three sampling techniques that affect the probability distribution functions associated to different vertices along this path. These are tailored for NLOS scenes, which are challenging due to the main part of the geometry being hidden both from the sensor and the light source.

Relay wall space. The key difference between conventional line-of-sight (LOS) images and their NLOS counterparts lies in the topology of (x, y). While LOS images aim to reconstruct a continuous signal, where (x, y) are continuous on the image plane, NLOS captures use sparse points in the visible surfaces, following arbitrary distributions. The reason is that typical setups are based on well-collimated sensors that only measure light coming from differential solid angle centered on a specific direction ω_s. Therefore, for each measurement in (x, y), the sensor importance W_e is a singular function in the spatio-temporal domain, as

$$W_e(\mathbf{x}_{k-1} \rightarrow \mathbf{x}_k, \Delta t_k) = \delta(\mathbf{x}_p - \mathbf{x}_k)\, \delta(\omega_s - \omega_{k-1})\, K_t(\Delta t_k), \tag{7}$$

with $\delta(\cdot)$ the Dirac delta, $\mathbf{x}_p$ the position of the sensor directed at $\omega_s = \text{norm}(\mathcal{M}_s(x_s, y_s) - \mathbf{x}_p)$ and $\omega_{k-1} = \text{norm}(\mathbf{x}_{k-1} - \mathbf{x}_k)$ normalized directions, and $K_t(\Delta t_k)$ the temporal importance. The distribution of these vertices for each path, i.e. $p(\mathbf{x}_{k-1}|\mathbf{x}_k)$, is uniform and regular along the relay wall space.

In our examples we show a planar relay wall space, used in classical NLOS setups, but any parametrizable 2D surface can be used. Thus, each (x, y) coordinate represents the projection of one point in the relay wall. A rectangular sampling grid is typically used so, in this case, the image resolution matches the number of sampled points in the horizontal and vertical directions. Fig. 4 shows an example image $I(x, y, t)$ at different instants t using histogram estimation.

Laser sampling. In a typical NLOS capture setup, light is emitted from a laser towards the visible surfaces (the relay wall), therefore resulting in emission in a differential solid angle, modeled as

$$L_e(\mathbf{x}_0 \rightarrow \mathbf{x}_1, \Delta t_0) = \delta(\mathbf{x}_e - \mathbf{x}_0)\, \delta(\omega_l - \omega_0)\, E(\Delta t_0), \tag{8}$$

with $\mathbf{x}_e$ the position of the laser, $\omega_l = \text{norm}(\mathbf{x}_e - \mathcal{M}_l(x_l, y_l))$ and $\omega_0 = \text{norm}(\mathbf{x}_0 - \mathbf{x}_1)$, and $E(\Delta t_0)$ the emitted radiance of the laser over time. However, paths in the search space Ω generated using conventional sampling techniques have a zero-probability of finding the light source through an indirect interaction given the delta functions in Eq. (8), as it is only illuminating a differential spot in the relay wall. One could use a small area light instead, but this approach still results in a very slow convergence rate.

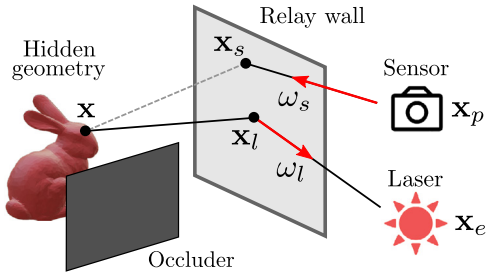

Fig. 5. Laser sampling overview. We cannot draw explicit connections to the emitter $\mathbf{x}_e$ directly as it is occluded. Instead, our approach samples the illuminated point in the relay wall $\mathbf{x}_l$.

Our laser sampling technique is inspired by next-event estimation, which draws explicit connections between every path vertex and light source. We cannot apply next-event estimation directly, as we need to connect the path vertices to the relay wall and not the laser origin. For this reason, we need to search for the laser light source through a bounce in the geometry.

Consider a laser emitter targeted at $\mathcal{M}_l(x_l, y_l) = \mathbf{x}_l$, also illustrated in Fig. 5. We can analytically compute the contribution from the laser $\mathbf{x}_e$ to any other point $\mathbf{x}$ through the relay wall $\mathbf{x}_l$:

$$\hat{L_e}(\mathbf{x}_0 \rightarrow \mathbf{x}, \Delta t_0) = \delta(\mathbf{x}_e - \mathbf{x}_0)\, \mathfrak{T}(\mathbf{x}_l \rightarrow \mathbf{x}, \Delta t_l)\, E(\Delta t_0), \tag{9}$$

where $\mathfrak{T}(\mathbf{x}_l \rightarrow \mathbf{x}, \Delta t_l)$ denotes partial throughput from $\mathbf{x}_l$ to $\mathbf{x}$:

$$\mathfrak{T}(\mathbf{x}_l \rightarrow \mathbf{x}, \Delta t_l) = \rho(\mathbf{x}_l, \Delta t_l)\, G(\mathbf{x}_l, \mathbf{x})\, V(\mathbf{x}_l, \mathbf{x}), \tag{10}$$

with $V(\cdot)$ the visibility term. The geometry term $G(\cdot)$ models light scattering on the relay wall as

$$G(\mathbf{x}_l, \mathbf{x}) = \frac{|\mathbf{n} \cdot \text{norm}(\mathbf{x}_l - \mathbf{x})|\ |\mathbf{n}_l \cdot \text{norm}(\mathbf{x} - \mathbf{x}_l)|}{\|\mathbf{x}_l - \mathbf{x}\|_2}, \tag{11}$$

with $\mathbf{n}$ and $\mathbf{n}_l$ used to denote the surface normals at $\mathbf{x}$ and $\mathbf{x}_l$ respectively. In order, Eq. (11) considers the scattering kernel, Lambert's cosine law and inverse square falloff respectively. This technique is similar to creating a virtual point light source located at the relay wall. While bidirectional path tracing can also be used for this purpose, our approach is much simpler, as it fits the purpose of NLOS scenes introducing very little overhead. Only one point $\mathbf{x}_l$ is illuminated, thus this vertex must be included in a path for its contribution to be non-zero. In our laser sampling technique, we always sample this vertex, so the probability distribution for the vertex $\mathbf{x}_1$, illuminated by a light source in $\mathbf{x}_0$, that is,

$$p(\mathbf{x}_1|\mathbf{x}_2) = p(\mathbf{x}_1) = \delta(\mathbf{x}_l - \mathbf{x}_1), \tag{12}$$

is a delta function independent of $p(\mathbf{x}_2)$. However, performing only laser sampling produces biased results that ignore longer paths, such as interreflections in the hidden geometry. For this reason, our laser sampling technique is combined with the classical sampling strategy of the forward path tracer using multiple importance sampling (MIS) at every interaction.

Hidden geometry sampling. The main goal of NLOS capture is to measure the back-scattering of the hidden geometry (see Fig. 4). This geometry can only be found through an indirect bounce on the relay wall (either from the point of view of the sensor or the emitter) and can be far from the wall. In a forward path tracer, the generated vertices usually follow a probability distribution $p(\mathbf{x}_{j-1}|\mathbf{x}_j)$ that either targets the reflectance distribution (which, in general, is unlikely to reach the geometry) or the light source location (which is also far from the hidden geometry). Thus, such a strategy presents a low probability of finding the hidden geometry given its small projected solid angle to a point in the relay wall.

For this sampling technique, we generate a random vertex $\mathbf{x}$ on the surface of the hidden geometry. Its contribution can be computed with $\mathfrak{T}(\mathbf{x} \rightarrow \mathbf{x}_s, \Delta\mathbf{x})$ as per Eq. (10). We cast a shadow ray to estimate the visibility term $V(\mathbf{x}, \mathbf{x}_s)$. The probability of each sample is

$$p(\mathbf{x}_{k-2}|\mathbf{x}_{k-1}) = p(\mathbf{x}_{k-2}) = 1/A, \tag{13}$$

where A is the total surface area of the hidden geometry. Unlike laser sampling, this approach alone will not produce biased results. Nonetheless, using MIS along conventional reflectance sampling techniques can yield better results if the hidden geometry is close to the relay wall.

5. Results

Implementation details. In the real world, a camera sensor is not able to absorb any radiant flux over a delta time period, as the resulting energy would be zero. As a consequence, we use a discretization of the temporal domain, following previous works on transient rendering [34] For binning, samples for each pixel are accumulated on a time histogram ranging from times b_{start} to b_{end}, which can be seen as the temporal sensitivity of W_e. Each bin has equal width b_{width}, as detailed in Fig. 6. For an histogram with B bins, it must hold that $b_{\text{end}} = b_{\text{start}} + B \cdot b_{\text{width}}$. The ith bins of all the per-pixel histograms are combined to obtain one frame of the transient render.

Also, as mentioned, trying to find a random path that falls in a specific bin with a non-zero contribution is inefficient. Instead, each random path is stored on its corresponding ith bin based on the time of arrival at the sensor t_k, with index $i = (t_k - b_{\text{start}})/b_{\text{width}}$ only if $b_{\text{start}} \leq t_k \leq b_{\text{end}}$. Histogram binning has a trade-off between high temporal resolution and faster convergence of the algorithm depending on the chosen bin width b_{width} and the number of samples per pixel. Note that histograms are the most naive and most inefficient forms of density estimation [45], and more effective forms for reconstructing the temporal domain in transient rendering have been proposed [13]. The application of these sophisticated density estimation techniques is orthogonal to our contribution and left as future work.

Our techniques have been implemented in the Mitsuba 2 rendering system. All of the following tests have been performed on an Intel Xeon Gold G140 at 3.7 GHz with 256 GB of main memory. However, the system can work with much less memory and computing power thanks to its efficiency. Table 1 shows an overview of the complexity of each of the scenes described. Generally, NLOS scenes in this work are simpler as they focus on computational imaging applications instead of photorealism.

Line-of-sight scenes. Fig. 2 shows a standard *Cornell Box* with two cubes and a small area light source in the ceiling. All elements consist of diffuse materials. The time-resolved images show a separation between the direct and indirect bounces. After some time, the color bleeding effect of the red and green walls can be seen on the white back wall. Fig. 7 shows two scenes. The *Bistro* scene showcases complex geometry with an area light

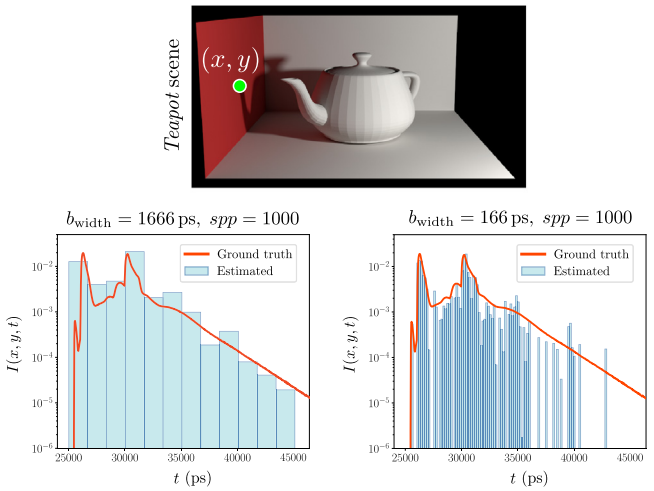

Fig. 6. Histogram density estimation for pixel (x, y) in the *Teapot* scene, using multiple configurations of bin widths b_{width} and samples per pixel *spp*. Ground truth temporal profile is obtained using much finer bin width and larger number of samples per pixel to approximate the continuous result.

Table 1
Triangle count for the scenes described in this work. Each type is ordered from less complex to more complex light interactions.

Type	Scene	# triangles
Line-of-sight	*Teapot*	6, 320
	Cornell box	33
	Bistro	2, 832, 120
	Contemporary bathroom	592, 186
Non-line-of-sight	*Z letter*	40
	NLOS letters	27
	Bunny	4, 968
	Lucy	499, 516
	White room	1, 219, 212

located in the sky. The *Bathroom* scene has a rectangular area light source located in the window. For this reason, objects that are closer to the emitter and camera appear first. Note how the reflection of the mirror appears after the rest of the scene has been illuminated, as the resulting paths are longer.

Non-line-of-sight scenes. Fig. 1 shows an example of simulated captured data. Current NLOS reconstruction algorithms are limited when working with complex geometry, which can produce multipath interference and result in a low-quality reconstruction. For this reason, we choose to use planar, simple objects. Fig. 4 uses a *Z letter* composed of two parallel lines joined by a diagonal line. Note that, for practical reasons in the NLOS cases, we do not take into account the time of flight of the first and last bounces that join the laser and sensor with the relay wall, and the laser is always pointed at the center of this wall. As the laser emitter illuminates the center of the relay wall, the diagonal part of the letter has the shortest path and thus appears first at 60 ps. Later, at 240 ps, paths that contain the two parallel lines start to appear, creating two additional wavefronts on the top and bottom halves.

Our laser and hidden geometry sampling techniques reduce the noise introduced by the Monte Carlo estimation, as seen in Figs. 8 and 9. Note how the forward path tracing algorithm finds either very few paths (Fig. 8) or none (Fig. 9) that contribute to the final result. Adding hidden geometry sampling improves this issue. However, the effects of laser sampling are much more noticeable, as the projected solid angle of the illuminated area in the relay wall is much smaller than the projected solid angle of the hidden geometry, making it a larger issue for forward

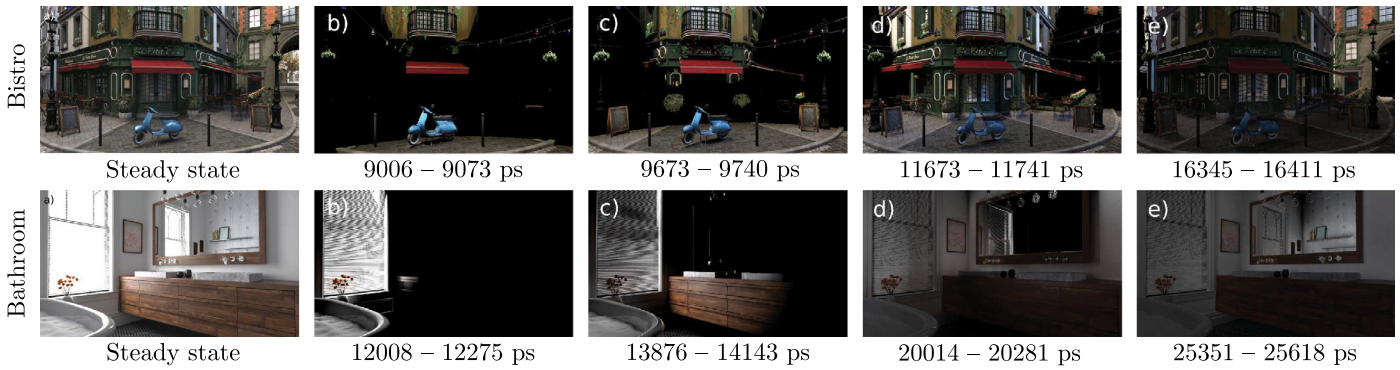

Fig. 7. Line-of-sight transient renders of the *Bistro* and the *Bathroom* scenes. Respectively, they are lit with an area light located in the sky and the bathroom window. The images show (a) steady-state and (b–e) transient renders. Note how the reflection on the bathroom mirror appears after the wall is illuminated, as its corresponding paths take more time.

Table 2
Render time for each of the proposed methods for the data shown in Fig. 8 (*Close*) and Fig. 9 (*Far*).

	Forward path tracing	Laser sampling	Hidden geometry sampling	Both
Close	37 m 21 s	45 m 13 s	80 m 16 s	88 m 40 s
Far	34 m 12 s	40 m 30 s	68 m 32 s	85 m 00 s

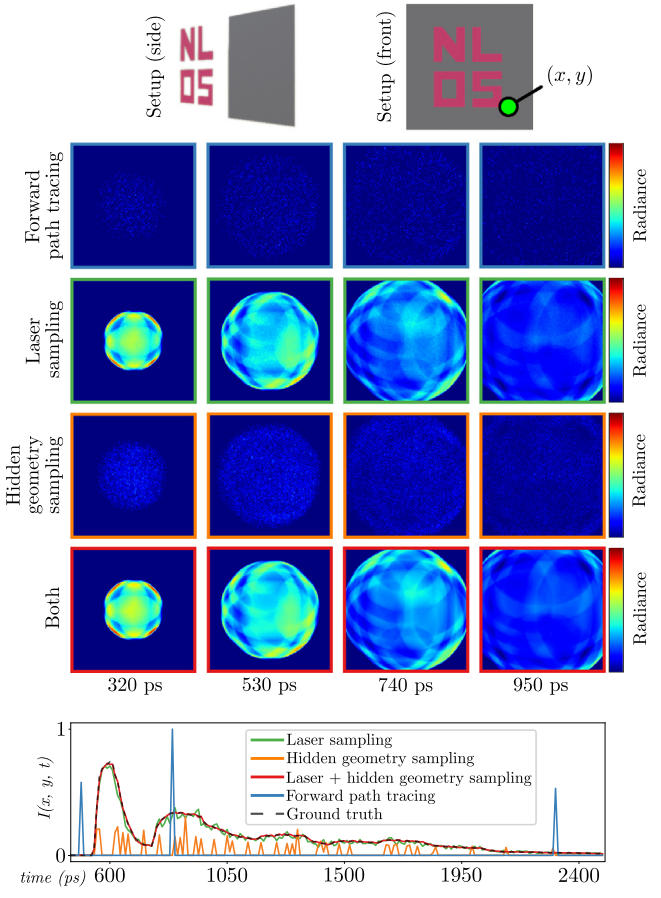

Fig. 8. Ablation test for our techniques in the *NLOS letters* scene, using 200,000 samples per pixel with a 256 × 256 resolution. The bottom row shows time-resolved radiance for the (x, y) pixel at the top. Without laser sampling, we use a light projector with a small solid angle to simulate illuminating a tiny area in the relay wall. The ground truth has been obtained using several orders of magnitude more samples per pixel.

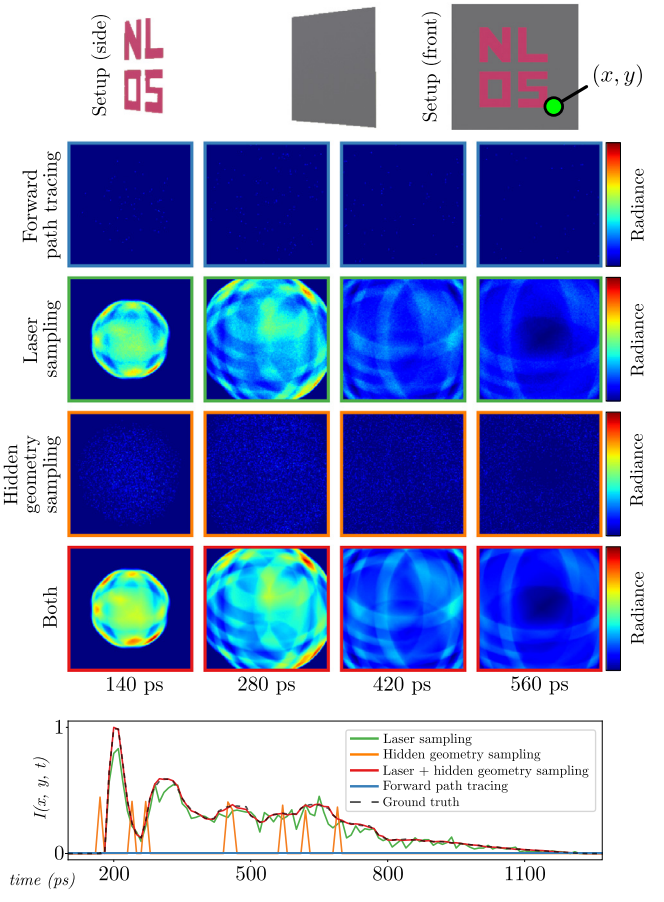

Fig. 9. Ablation test for our techniques in the *NLOS letters* scene, with the same configuration as described in Fig. 8. In this setup, the four hidden letters are separated three times as much from the relay wall. This makes our techniques much more effective when compared to conventional forward path tracing.

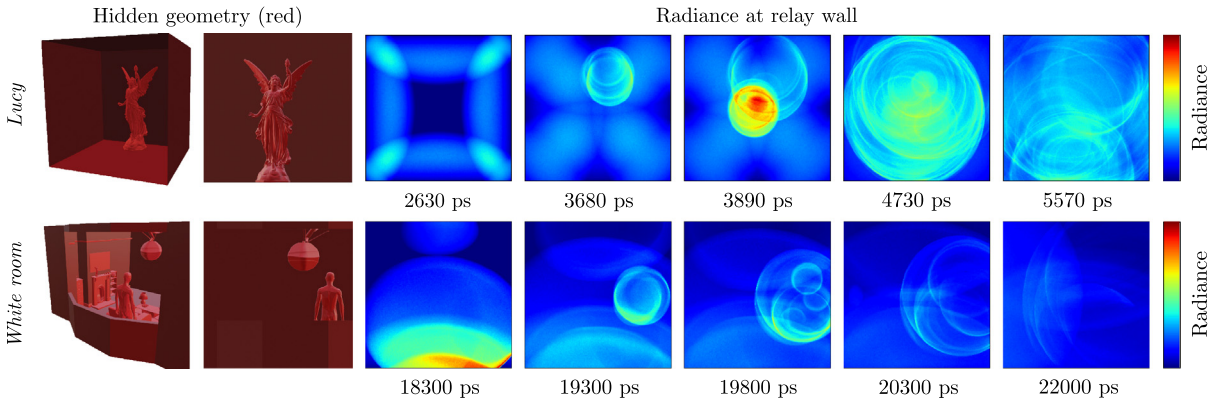

Fig. 10. NLOS renders of the *Lucy* and *White room* scenes, rendered with 1M samples per pixel with a 256 × 256 resolution. Both scenes are closed rooms, with the relay wall placed to close the four walls. The render times for each scene are 360.3 and 243.7 min, respectively. The front view (second column) depicts the scene viewed from the relay wall.

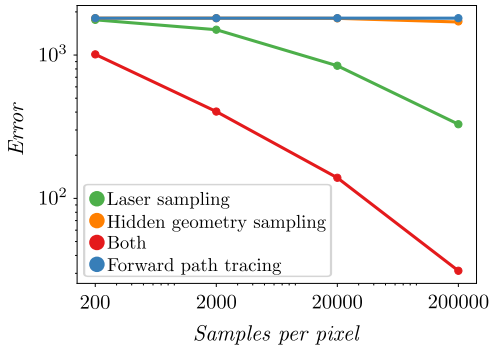

Fig. 11. Convergence analysis based on the number of samples per pixel in the *NLOS letters* scene. Error is computed as the mean squared error of each signal with respect to the ground truth mentioned in Fig. 9.

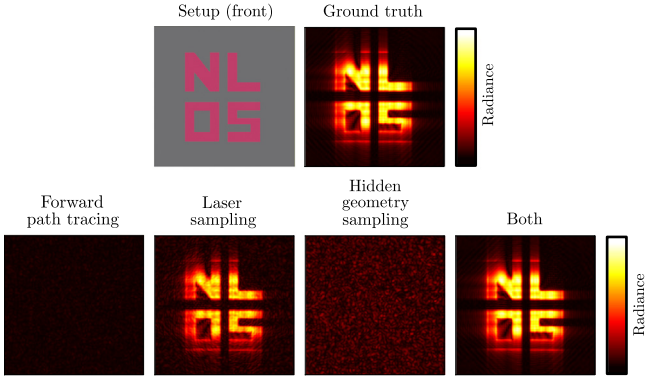

Fig. 12. Processing of the simulated data presented in Fig. 9 with the phasor fields-based NLOS reconstruction algorithm [10] for a scene computed with each of the proposed methods, for 200,000 samples per pixel.

path tracing. Bidirectional path tracing can also solve the light source issue, however, our method is simpler and can be implemented in a forward path tracing algorithm without adding a noticeable overhead. Finally, combining both laser and hidden geometry sampling greatly improves convergence, further reducing the variance of the results obtained with laser sampling. The effects of our sampling techniques are more noticeable if the hidden geometry is further away (Fig. 9). Note how moving the hidden geometry further away from the relay wall decreases the computation time, as more paths end prematurely, also when performing MIS.

Table 2 shows the computation time for each of the mentioned cases. Our methods require more time, but provide better results proportionally. Fig. 11 shows a convergence analysis based on the number of samples per pixel used to render the *NLOS letters* scene. As stated before, the laser sampling technique alone is more effective than hidden geometry sampling, but combining both sampling strategies together provides better results by an order of magnitude.

The resulting render is used by NLOS imaging algorithms, obtaining successful reconstructions of the hidden geometry with a fraction of the cost of other capture setups, as seen in Fig. 12. It does not suffer from the noise introduced by real captures, which could be seen as good or bad. Nevertheless, other authors have used synthetic scenes that mimic reality [18], showing that the simulated data fits well with data captured with physical hardware and thus can be used as a replacement.

Lastly, Fig. 10 shows two complex scenes in closed rooms, based on the Z-NLOS dataset [46]. They are challenging scenes for NLOS reconstruction algorithms, mainly because of the multipath interference caused by indirect bounces. Note how the multiple wavefronts that appear in the relay wall are caused by the hidden geometry, and how their time is related to the distance of each geometry point to the emission source in the center of the relay wall.

Polarization. As our implementation is done in Mitsuba 2, it allows for the simulation of polarization of the light. Fig. 13 shows a *Cornell Box* scene with three spheres: two have a dielectric material and one has a conductor material. It shows the polarization state of the light using the four Stokes vector components, for both the steady and transient states. Note how light bouncing on the spheres changes its polarization over time.

6. Conclusions

In this work we have implemented a transient forward path tracing algorithm following the transient path integral [13] into Mitsuba 2, extending it to support simple sampling techniques targeting NLOS capture setups. Our technique reduces the path integration search space by targeting scene geometry hidden behind an intermediate wall and illuminated with a tiny area light, effectively improving the convergence of the algorithm by a large factor.

Point sampling in the relay-wall space allows simulating the behavior of capture devices typically used in non-line-of-sight environments. In our work, we have made use of flat shaped relay walls, but our technique can be applied to any parametrizable 2D surface. Still, the reconstruction framework by Liu et al. [10] is

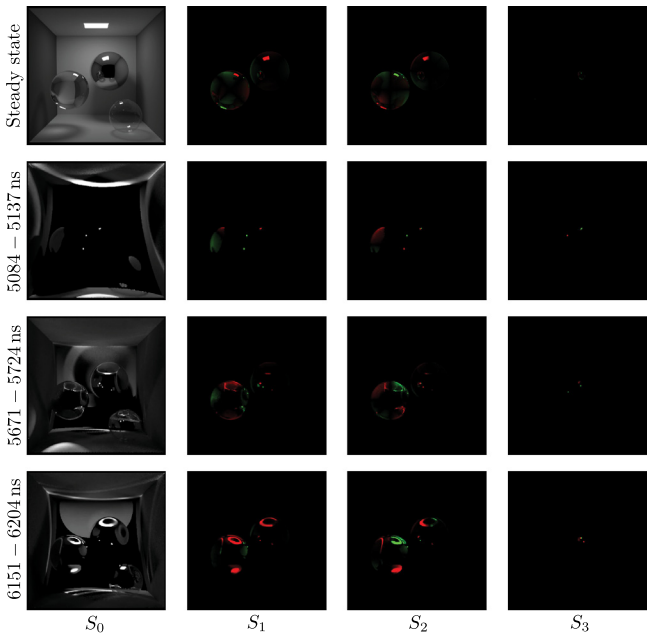

Fig. 13. *Cornell box* scene. It contains three spheres, from left to right: dielectric (diamond), conductor (aluminum) and another thin dielectric. Polarization is represented using Stokes vectors. S_0 represents irradiance i.e. the intensity of the beam represented using grayscale. S_1 uses green and red values to represent the predominance of horizontally vs vertically polarized light, respectively. Similarly, S_2 represents the predominance of linear vs perpendicular diagonal polarization, and S_3 represents the predominance of right-circularly vs left-circularly polarized light. The top row shows the steady-state render, while the bottom three rows show transient renders at different time instants.

only designed for planar surfaces, and simulating capture setups with curved or irregular relay walls could be used to test other reconstruction algorithms or develop new ones.

The proposed laser sampling technique, that creates a virtual point light (VPL) in the relay wall, greatly reduces variance in the simulation. As with all VPL-based methods, this is a form of bidirectional path tracing [47] with single-segment light subpaths. Of course, larger light subpaths could be used for larger variance reduction via MIS, but that would make the wavefront-based GPU implementation in Mitsuba 2 significantly more complex while introducing some overhead due to a quadratic growth on shadow connections during MIS. Going beyond, the guiding technique we propose for sampling the contribution of hidden objects could introduce additional complexity in the computation of the path contributions, resulting in a form of *tridirectional path tracing* [48].

Lastly, we have implemented our technique on top of the Mitsuba 2 rendering software, so in practice we take advantage of its parallelization and vectorization capabilities and its ability to include more sophisticated light transport information such as polarization, extending it to transient state. Additionally, Mitsuba 2 can also be used as a differentiable renderer, so our implementation, while still unexplored, could also apply differentiable rendering in the non-line-of-sight domain. For this and other future paths of research, we provide the source code of our approach.[1]

CRediT authorship contribution statement

Diego Royo: Conceptualization, Software, Validation, Visualization, Writing – original draft. **Jorge García:** Software, Validation, Visualization. **Adolfo Muñoz:** Conceptualization, Validation, Writing – review & editing, Supervision. **Adrian Jarabo:** Conceptualization, Validation, Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (project CHAMELEON, Grant No 682080), the EU MSCA-ITN programme (project PRIME, Grant No 956585) and MCIN/AEI/10.13039/501100011033 (project VIRTUALIGHT, Grant PID2019-105004GB-I00). Additionally, Diego Royo was supported by a Gobierno de Aragón (2021–2025) predoctoral grant.

References

[1] Wu D, Velten A, O'Toole M, Masia B, Agrawal A, Dai Q, et al. Decomposing global light transport using time of flight imaging. Int J Comput Vis 2014;107(2):123–38.
[2] Marco J, Jarabo A, Nam JH, Liu X, Cosculluela MA, Velten A, et al. Virtual light transport matrices for non-line-of-sight imaging. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021. p. 2440–9.
[3] Su S, Heide F, Swanson R, Klein J, Callenberg C, Hullin M, et al. Material classification using raw time-of-flight measurements. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. p. 3503–11.
[4] Naik N, Zhao S, Velten A, Raskar R, Bala K. Single view reflectance capture using multiplexed scattering and time-of-flight imaging. In: Proceedings of the 2011 SIGGRAPH Asia conference. 2011. p. 1–10.
[5] Heide F, Xiao L, Kolb A, Hullin MB, Heidrich W. Imaging in scattering media using correlation image sensors and sparse convolutional coding. Opt Express 2014;22(21):26338–50.
[6] Wu R, Jarabo A, Suo J, Dai F, Zhang Y, Dai Q, et al. Adaptive polarization-difference transient imaging for depth estimation in scattering media. Opt Lett 2018;43(6):1299–302.
[7] Velten A, Wu D, Jarabo A, Masia B, Barsi C, Joshi C, et al. Femto-photography: Capturing and visualizing the propagation of light. ACM Trans Graph 2013;32(4).
[8] Velten A, Willwacher T, Gupta O, Veeraraghavan A, Bawendi MG, Raskar R. Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging. Nature Commun 2012;3(1):1–8.
[9] Arellano V, Gutierrez D, Jarabo A. Fast back-projection for non-line of sight reconstruction. Opt Express 2017;25(10):11574–83.
[10] Liu X, Guillén I, La Manna M, Nam JH, Reza SA, Huu Le T, et al. Non-line-of-sight imaging using phasor-field virtual wave optics. Nature 2019;572(7771):620–3.
[11] O'Toole M, Lindell DB, Wetzstein G. Confocal non-line-of-sight imaging based on the light-cone transform. Nature 2018;555(7696):338–41.
[12] Maeda T, Satat G, Swedish T, Sinha L, Raskar R. Recent advances in imaging around corners. 2019, arXiv.
[13] Jarabo A, Marco J, Munoz A, Buisan R, Jarosz W, Gutierrez D. A framework for transient rendering. ACM Trans Graph 2014;33(6):1–10.
[14] Nimier-David M, Vicini D, Zeltner T, Jakob W. Mitsuba 2: A retargetable forward and inverse renderer. ACM Trans Graph 2019;38(6):1–17.
[15] Jarabo A, Masia B, Marco J, Gutierrez D. Recent advances in transient imaging: A computer graphics and vision perspective. Vis Inform 2017;1(1):65–79.
[16] Ramesh R, Davis J. 5D time-light transport matrix: What can we reason about scene properties? Tech. Rep., 2008.
[17] Gupta M, Nayar SK, Hullin MB, Martin J. Phasor imaging: A generalization of correlation-based time-of-flight imaging. ACM Trans Graph 2015;34(5):1–18.
[18] Marco J, Hernandez Q, Munoz A, Dong Y, Jarabo A, Kim MH, et al. DeepToF: Off-the-shelf real-time correction of multipath interference in time-of-flight imaging. ACM Trans Graph 2017;36(6):1–12.
[19] Kirmani A, Hutchison T, Davis J, Raskar R. Looking around the corner using ultrafast transient imaging. Int J Comput Vis 2011;95(1):13–28.
[20] Velten A, Wu D, Jarabo A, Masia B, Barsi C, Lawson E, et al. Relativistic ultrafast rendering using time-of-flight imaging. In: ACM SIGGRAPH 2012 talks. 2012. p. 1.

[1] https://github.com/diegoroyo/mitsuba2-transient-nlos

[21] Heide F, Hullin MB, Gregson J, Heidrich W. Low-budget transient imaging using photonic mixer devices. ACM Trans Graph 2013;32(4):1–10.
[22] Laurenzis M, Velten A. Nonline-of-sight laser gated viewing of scattered photons. Opt Eng 2014;53(2):023102.
[23] Buttafava M, Zeman J, Tosi A, Eliceiri K, Velten A. Non-line-of-sight imaging using a time-gated single photon avalanche diode. Opt Express 2015;23(16):20997–1011.
[24] O'Toole M, Heide F, Lindell DB, Zang K, Diamond S, Wetzstein G. Reconstructing transient images from single-photon sensors. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017. p. 1539–47.
[25] Baradad M, Ye V, Yedidia AB, Durand F, Freeman WT, Wornell GW, et al. Inferring light fields from shadows. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018. p. 6267–75.
[26] Batarseh M, Sukhov S, Shen Z, Gemar H, Rezvani R, Dogariu A. Passive sensing around the corner using spatial coherence. Nature Commun 2018;9(1):1–6.
[27] Xin S, Nousias S, Kutulakos KN, Sankaranarayanan AC, Narasimhan SG, Gkioulekas I. 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. 2019. p. 6800–9.
[28] Iseringhausen J, Hullin MB. Non-line-of-sight reconstruction using efficient transient rendering. ACM Trans Graph 2020;39(1):1–14.
[29] Tsai C-Y, Sankaranarayanan AC, Gkioulekas I. 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. 2019. p. 1545–55.
[30] Young SI, Lindell DB, Girod B, Taubman D, Wetzstein G. Non-line-of-sight surface reconstruction using the directional light-cone transform. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020. p. 1407–16.
[31] Chen W, Wei F, Kutulakos KN, Rusinkiewicz S, Heide F. Learned feature embeddings for non-line-of-sight imaging and recognition. ACM Trans Graph 2020;39(6):1–18.
[32] Smith A, Skorupski J, Davis J. Transient rendering. Technical Report, Citeseer; 2008, UCSC-SOE-08-26.
[33] Adam A, Dann C, Yair O, Mazor S, Nowozin S. Bayesian time-of-flight for realtime shape, illumination and albedo. IEEE Trans Pattern Anal Mach Intell 2016;39(5):851–64.
[34] Jarabo A. Femto-photography: Visualizing light in motion. 2012.
[35] Keller M, Kolb A. Real-time simulation of time-of-flight sensors. Simul Model Pract Theory 2009;17(5):967–78.
[36] Marco J, Guillén I, Jarosz W, Gutierrez D, Jarabo A. Progressive transient photon beams. In: Computer graphics forum, vol. 38. (6):Wiley Online Library; 2019, p. 19–30.
[37] Chen W, Ling H, Gao J, Smith E, Lehtinen J, Jacobson A, et al. Learning to predict 3d objects with an interpolation-based differentiable renderer. Adv Neural Inf Process Syst 2019;32.
[38] Pan X, Arellano V, Jarabo A. Transient instant radiosity for efficient time-resolved global illumination. Comput Graph 2019;83:107–13.
[39] Baek S-H, Heide F. Polarimetric spatio-temporal light transport probing. ACM Trans Graph 2021;40(6):1–18.
[40] Jarabo A, Arellano V. Bidirectional rendering of vector light transport. In: Computer graphics forum, vol. 37. (6):Wiley Online Library; 2018, p. 96–105.
[41] Hullin MB. Computational imaging of light in flight. In: Optoelectronic imaging and multimedia technology III, vol. 9273. International Society for Optics and Photonics; 2014, 927314.
[42] Heide F, O'Toole M, Zang K, Lindell DB, Diamond S, Wetzstein G. Non-line-of-sight imaging with partial occluders and surface normals. ACM Trans Graph 2019;38(3):1–10.
[43] Yi S, Kim D, Choi K, Jarabo A, Gutierrez D, Kim MH. Differentiable transient rendering. ACM Trans Graph 2021;40(6):1–11.
[44] Pediredla A, Veeraraghavan A, Gkioulekas I. Ellipsoidal path connections for time-gated rendering. ACM Trans Graph 2019;38(4):1–12.
[45] Silverman BW. Density estimation for statistics and data analysis, vol. 26. CRC Press; 1986.
[46] Galindo M, Marco J, O'Toole M, Wetzstein G, Gutierrez D, Jarabo A. A dataset for benchmarking time-resolved non-line-of-sight imaging. In: IEEE international conference on computational photography. IEEE; 2019, URL https://graphics.unizar.es/nlos.
[47] Veach E, Guibas LJ. Optimally combining sampling techniques for Monte Carlo rendering. In: Proceedings of the 22nd annual conference on computer graphics and interactive techniques. 1995. p. 419–28.
[48] Anderson L, Li T-M, Lehtinen J, Durand F. Aether: An embedded domain specific sampling language for Monte Carlo rendering. ACM Trans Graph 2017;36(4):1–16.