Self-Calibrating, Fully Differentiable NLOS Inverse Rendering

Kiseok Choi
KAIST
South Korea
kschoi@vclab.kaist.ac.kr

Inchul Kim
KAIST
South Korea
ickim@vclab.kaist.ac.kr

Dongyoung Choi
KAIST
South Korea
dychoi@vclab.kaist.ac.kr

Julio Marco
Universidad de Zaragoza - I3A
Spain
juliom@unizar.es

Diego Gutierrez
Universidad de Zaragoza - I3A
Spain
diegog@unizar.es

Min H. Kim
KAIST
South Korea
minhkim@vclab.kaist.ac.kr

ABSTRACT

Existing time-resolved non-line-of-sight (NLOS) imaging methods reconstruct hidden scenes by inverting the optical paths of indirect illumination measured at visible relay surfaces. These methods are prone to reconstruction artifacts due to inversion ambiguities and capture noise, which are typically mitigated through the manual selection of filtering functions and parameters. We introduce a fully-differentiable end-to-end NLOS inverse rendering pipeline that self-calibrates the imaging parameters during the reconstruction of hidden scenes, using as input only the measured illumination while working both in the time and frequency domains. Our pipeline extracts a geometric representation of the hidden scene from NLOS volumetric intensities and estimates the time-resolved illumination at the relay wall produced by such geometric information using differentiable transient rendering. We then use gradient descent to optimize imaging parameters by minimizing the error between our simulated time-resolved illumination and the measured illumination. Our end-to-end differentiable pipeline couples diffraction-based volumetric NLOS reconstruction with path-space light transport and a simple ray marching technique to extract detailed, dense sets of surface points and normals of hidden scenes. We demonstrate the robustness of our method to consistently reconstruct geometry and albedo, even under significant noise levels.

CCS CONCEPTS

• **Computing methodologies → Image and video acquisition**.

KEYWORDS

Non-line-of-sight imaging, image reconstruction, computational imaging

ACM Reference Format:
Kiseok Choi, Inchul Kim, Dongyoung Choi, Julio Marco, Diego Gutierrez, and Min H. Kim. 2023. Self-Calibrating, Fully Differentiable NLOS Inverse Rendering. In *SIGGRAPH Asia 2023 Conference Papers (SA Conference Papers '23), December 12–15, 2023, Sydney, NSW, Australia.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3610548.3618140

SA Conference Papers '23, December 12–15, 2023, Sydney, NSW, Australia

ACM ISBN 979-8-4007-0315-7/23/12.
https://doi.org/10.1145/3610548.3618140

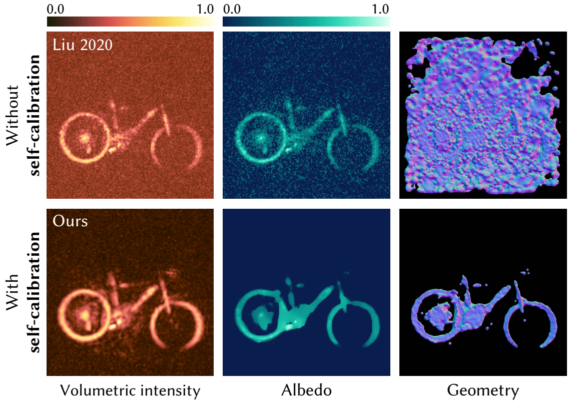

Figure 1: We present a self-calibrating, fully-differentiable NLOS inverse rendering pipeline for the reconstruction of hidden scenes. Our method only requires transient measurements as input and relies on differentiable rendering and implicit surface estimation from NLOS volumetric outputs to obtain the optimal NLOS imaging parameters that yield accurate surface points, normals, and albedo reconstructions of the hidden scene. The top row shows the reconstructed volumetric intensity, albedo, and 3D geometry of a real scene [Liu et al. 2020], failing to reconstruct geometry estimation due to noise interference. The bottom row demonstrates our results after optimization of the imaging parameters.

1 INTRODUCTION

Time-gated non-line-of-sight (NLOS) imaging algorithms aim to reconstruct hidden scenes by analyzing time-resolved indirect illumination on a visible relay surface [Faccio et al. 2020; Jarabo et al. 2017; Satat et al. 2016]. These methods typically emit ultra-short illumination pulses on the relay surface, and estimate the hidden scene based on the time of flight of third-bounce illumination reaching the sensor [Lindell et al. 2019; Liu et al. 2019; O'Toole et al. 2018; Velten et al. 2012; Xin et al. 2019].

The majority of existing methods estimate hidden geometry by backprojecting captured third-bounce illumination into a voxelized space that represents the hidden scene [Laurenzis and Velten 2014], lacking information about surface orientation and self-occlusions [Iseringhausen and Hullin 2020]. Moreover, captured data contains higher-order indirect illumination and high-frequency noise from different sources that introduce undesired artifacts in the reconstructions. Performing a filtering step over the data or the reconstructed volume is the most common solution to mitigate errors and enhance the geometric features [Arellano et al. 2017; Buttafava et al. 2015; Liu et al. 2019; O'Toole et al. 2018; Velten et al. 2012]; however, this requires manual design and selection of filter parameters, as their impact in the reconstruction quality is highly dependent on the scene complexity, environment conditions, and hardware limitations.

Recent physically-based methods proposed an alternative technique that avoids the issues linked to backprojection. By merging a simplified but efficient three-bounce transient rendering formula with an optimization loop, the computed time-resolved illumination at the relay wall resulting from an optimized geometry reconstruction is compared to the measured illumination. However, geometric representations introduced by existing works limit the detail in the reconstructions [Iseringhausen and Hullin 2020] or fail to reproduce the boundaries of hidden objects [Tsai et al. 2019].

Alternatively, the recent development of accurate transient rendering methods [Jarabo et al. 2014; Pediredla et al. 2019; Royo et al. 2022] has fostered differentiable rendering pipelines in path space [Wu et al. 2021; Yi et al. 2021], which have the potential to become key tools in optimization schemes. However, differentiable methods are currently bounded by memory limitations since the need to compute the derivatives of time-resolved radiometric data severely limits the number of unknown parameters that can be handled. The difficulty of handling visibility changes in a differentiable manner, as well as the large number of parameters that need to be taken into account, are two limiting factors shared as well with steady-state differentiable rendering [Li et al. 2018; Zhao et al. 2020], that are further aggravated in the transient regime. As a result, NLOS imaging methods that rely on differentiable rendering are therefore limited to simple operations such as tracking the motion of a single hidden object with a known shape [Yi et al. 2021].

To address these problems, we propose a novel self-calibrated, fully differentiable pipeline for NLOS inverse rendering that jointly optimizes system parameters and scene information to extract surface points, normals, and albedo of the hidden geometry. To this end, we combine diffractive phasor-field imaging in the frequency domain [Liu et al. 2020, 2019] with differentiable third-bounce transient rendering in the temporal domain. We leverage the volumetric output of phasor-field NLOS imaging to estimate geometric information of the hidden scene, which we then use on a transient rendering step to simulate time-resolved illumination at the relay wall. By minimizing the error between simulated and captured illumination, we provide a fully-differentiable pipeline for self-calibrating NLOS imaging parameters in an end-to-end manner.

Our optimized parameters provide accurate volumetric outputs from which we estimate surface points, normals and albedos of hidden objects, with more geometric detail than previous surface-based methods. Our method is robust in the presence of noise, providing consistent geometric estimations under varying capture conditions. Our code is freely available for research purposes[1].

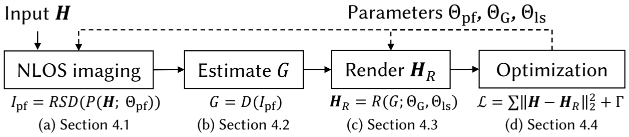
Input $\boldsymbol{H}$
Parameters Θ_{pf}, Θ_{G}, Θ_{ls}
NLOS imaging
Estimate G
Render $\boldsymbol{H}_R$
Optimization
$I_{\mathrm{pf}} = RSD(P(\boldsymbol{H};\ \Theta_{\mathrm{pf}}))$
(a) Section 4.1
$G = D(I_{\mathrm{pf}})$
(b) Section 4.2
$\boldsymbol{H}_R = R(G; \Theta_{\mathrm{G}}, \Theta_{\mathrm{ls}})$
(c) Section 4.3
$\mathcal{L} = \sum \|\boldsymbol{H} - \boldsymbol{H}_R\|_2^2 + \Gamma$
(d) Section 4.4

Figure 2: Overview of our self-calibrated, fully differentiable NLOS inverse rendering workflow (Sections 3 and 4). (a) We perform NLOS imaging using a phasor-field diffraction method, taking an initial matrix H of transient measurements as input, and outputting volumetric intensity I_{pf}. (b) We estimate G, an implicit geometric representation of the hidden scene, from I_{pf}. (c) We obtain the time-resolved illumination H_R from G using differentiable path-space transient rendering. (d) We optimize imaging parameters until the error between H and H_R converges with regularization terms Γ. Geometry G is computed during the forward pass, while Θ_{pf}, Θ_{ls}, and Θ_{G} are updated during the backward pass.

2 RELATED WORK

Active-light NLOS imaging methods provide 3D reconstructions of general NLOS scenes by leveraging temporal information of light propagation by means of time-gated illumination and sensors [Faccio et al. 2020; Jarabo et al. 2017].

Scene representation. While existing methods rely on inverting third-bounce transport, they may differ in their particular representation of scene geometry as volumetric density or surfaces. Volumetric approaches estimate geometric density by backprojecting third-bounce light paths onto a voxelized space [Ahn et al. 2019; Arellano et al. 2017; Buttafava et al. 2015; Gariepy et al. 2015; Gupta et al. 2012; La Manna et al. 2018; Velten et al. 2012]. Efficiently inverting the resulting discrete light transport matrix is not trivial; many dimensionality reduction methods have been proposed [Heide et al. 2019; Lindell et al. 2019; O'Toole et al. 2018; Xin et al. 2019; Young et al. 2020], but they are often limited in spatial resolution (as low as 64×64 in some cases) due to memory constraints. Surface methods, in contrast, rely on inverting third-bounce light transport onto explicit representations of the geometry [Iseringhausen and Hullin 2020; Plack et al. 2023; Tsai et al. 2019], usually starting with simple blob shapes, progressively optimizing the geometry until loss converges. In contrast, we estimate *implicit* geometric representations of the hidden scene based on surface points and normals by ray marching the volumetric output of NLOS imaging, inspired by recent work on neural rendering [Barron et al. 2021; Mildenhall et al. 2020; Niemeyer et al. 2022]. The combination of NLOS imaging with differentiable transient rendering over the estimated geometry allows us to self-calibrate imaging parameters in an end-to-end manner. For clarity, in this paper the term *explicit* surface refers to a polygonal surface mesh, while *implicit* surface denotes a representation based on surface points and their normals, without defining a surface mesh. Please, refer to Section 4.2 for a further detailed discussion on explicit/implicit surface representations.

Learning-based approaches. Other methods leverage neural networks instead, such as U-net [Grau Chopite et al. 2020], convolutional neural networks [Chen et al. 2020], or neural radiance fields

[1] https://github.com/KAIST-VCLAB/nlos-inverse-rendering.git

[Mu et al. 2022]. These learning-based methods are learned using object databases such as ShapeNet [Chang et al. 2015]. However, their parameters are trained with steady-state renderings of synthetic scenes composed of a single object behind an occluder in an otherwise empty space. As such, their performance is often degraded with real scenes, often overfitting to the training dataset, and becoming susceptible to noise. Our method does not rely on a pre-trained deep network to extract high-level features from synthetic steady-state rendering data; instead, we explicitly optimize virtual illumination functions and scene information by evaluating actual transient observations, without relying on neural networks. Recent works by Shen et al. [2021] and Fujimura et al. [2023] leverage transient observations similar to ours for optimizing multi-layer perceptrons for imaging. However, these methods cannot be utilized for calibrating the filtering parameters of volumetric NLOS methods due to the lack of evaluation of the physical observation of the transient measurements by an NLOS imaging and light transport model.

Wave-based NLOS imaging. Recent works have shifted the paradigm of third-bounce reconstruction approaches to the domain of wave optics [Lindell et al. 2019; Liu et al. 2019]. In particular, the phasor field framework [Liu et al. 2019] computationally transforms the data captured on the relay surface into illumination arriving at a virtual imaging aperture. This has enabled more complex imaging models (e.g., [Dove and Shapiro 2020a,b; Guillén et al. 2020; Marco et al. 2021; Reza et al. 2019]), and boosted the efficiency of NLOS imaging to interactive and real-time reconstruction rates [Liao et al. 2021; Liu et al. 2020; Mu et al. 2022; Nam et al. 2021]. However, these systems require careful calibration of all their parameters, including the definition of the phasor field and the particular characteristics of lasers and sensors, which makes using them a cumbersome process. Our fully self-calibrated system overcomes this limitation.

3 TIME-GATED NLOS IMAGING MODEL

We propose a differentiable end-to-end inverse rendering pipeline (shown in Figure 2) to improve the reconstruction quality of hidden scenes by optimizing the parameters of NLOS imaging algorithms without prior knowledge of the hidden scene. In the following, we describe our NLOS imaging model. Section 4 describes our optimization pipeline based on this NLOS imaging model.

3.1 Phasor-based NLOS imaging

In a standard NLOS imaging setup (see Figure 3), a laser beam is emitted towards a point $\mathbf{x}_l$ on a visible relay wall, which reflects light towards the hidden scene and then is reflected back to the wall. The hidden scene is imaged based on the time of flight of the time-resolved illumination, captured at points $\mathbf{x}_s$ on the relay wall in the form of a measurement matrix $\boldsymbol{H}(\mathbf{x}_l, \mathbf{x}_s, t)$.

The recent diffractive phasor-based framework by Liu et al. [2020; 2019] intuitively turns the grid of measured points $\mathbf{x}_s$ on the relay wall into a virtual aperture; this allows to formulate the reconstruction of NLOS scenes as a virtual *line-of-sight* (LOS) problem.

We define $\boldsymbol{H}(\mathbf{x}_l, \mathbf{x}_s, \Omega)$ as a set of phasors at the relay wall, obtained by Fourier transform of the measurement matrix $\boldsymbol{H}(\mathbf{x}_l, \mathbf{x}_s, t)$. In practice, since this function $\boldsymbol{H}$ is noisy, we apply a filtering operation as

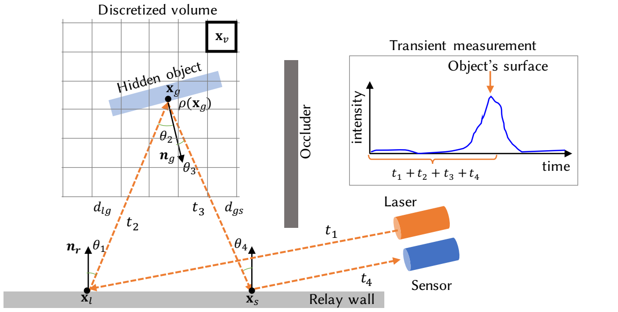

Figure 3: NLOS imaging setup. A laser emits a pulse of light, which travels to the relay wall, then to the hidden geometry, back to the relay wall, and reaches the sensor after a travel time of $t = t_1 + t_2 + t_3 + t_4$. The inset shows the sensor response; the peak at t indicates the presence of a hidden object.

$$\boldsymbol{H}_{\text{pf}}(\mathbf{x}_l, \mathbf{x}_s, \Omega) = \mathcal{P}(\mathbf{x}_l, \mathbf{x}_s, \Omega)\, \boldsymbol{H}(\mathbf{x}_l, \mathbf{x}_s, \Omega), \tag{1}$$

where $\mathcal{P}(\mathbf{x}_l, \mathbf{x}_s, \Omega)$ represents a virtual illumination function that acts as a filter over $\boldsymbol{H}$, typically defined as a spatially-invariant illumination function [Liu et al. 2020, 2019]. The hidden scene can then be imaged as an intensity function $I_{\text{pf}}(\mathbf{x}_v, t)$ on a voxelized space via Rayleigh-Sommerfeld Diffraction (RSD) operators as

$$I_{\text{pf}}(\mathbf{x}_v, t) = \left| \int_{-\infty}^{\infty} e^{i\frac{\Omega}{c}t} \iint_{S\,L} \frac{e^{-i\frac{\Omega}{c}(d_{lv}+d_{vs})}}{d_{lv}d_{vs}} \boldsymbol{H}_{\text{pf}}(\mathbf{x}_l, \mathbf{x}_s, \Omega)\, \mathrm{d}\mathbf{x}_l \mathrm{d}\mathbf{x}_s \frac{\mathrm{d}\Omega}{2\pi} \right|^2, \tag{2}$$

where L and S define the illuminated and measured regions on the relay wall, respectively; $d_{lv} = \|\mathbf{x}_l - \mathbf{x}_v\|$ and $d_{vs} = \|\mathbf{x}_v - \mathbf{x}_s\|$ are voxel-laser and voxel-sensor distances (see Figure 3); and Ω represents frequency.

Classic NLOS reconstruction methods reconstruct hidden geometry by evaluating $\boldsymbol{H}(\mathbf{x}_l, \mathbf{x}_s, t)$ at the time of flight of third-bounce illumination paths between scene locations and points on the relay surface [Arellano et al. 2017; Gupta et al. 2012; O'Toole et al. 2018]. This is analogous to evaluating $I_{\text{pf}}(\mathbf{x}_v, t)$ at $t = 0$, where the RSD propagators have traversed an optical distance $\|\bar{\mathbf{x}}\| = d_{lv} + d_{vs}$. We incorporate a similar third-bounce strategy in our path integral formulation as described in the following. Due to the challenges of estimating surface albedo due to diffraction effects during the NLOS imaging process [Guillén et al. 2020; Marco et al. 2021], we assume an albedo term per surface point that approximates the averaged reflectance observed from all sensor points.

3.2 Path-space light transport in NLOS scenes

To formally describe transient light transport in an efficient manner, we rely on the transient path integral formulation [Jarabo et al. 2014; Royo et al. 2022]. Transient light transport $\boldsymbol{H}(\mathbf{x}_l, \mathbf{x}_s, t) \in \mathbb{R}$ can then be expressed as

$$\boldsymbol{H}(\mathbf{x}_l, \mathbf{x}_s, t) = \int_{\mathcal{T}} \int_{\psi} \mathcal{K}(\bar{\mathbf{x}}, \mathbf{t})\mathrm{d}\mu(\bar{\mathbf{x}})\mathrm{d}\mu(\mathbf{t}), \tag{3}$$

where $\mathcal{K}$ is the radiometric contribution in transient path-space; $\mathrm{d}\mu(\bar{\mathbf{x}})$ is the differential measure of path $\bar{\mathbf{x}}$; $\mathcal{T}$ represents the domain of temporal measurements; $\mathbf{t} = t_l \ldots t_s$ is the sequence of time-resolved measurements on each vertex; $\mathrm{d}\mu(\mathbf{t})$ denotes temporal integration at each vertex; $\bar{\mathbf{x}} = \mathbf{x}_l \ldots \mathbf{x}_s$ is a set of discrete transient path time intervals of $k+1$ vertices; and $\psi = \cup_{k=1}^{\infty} \psi_k$ is the entire space of paths with any number of vertices, with ψ_k being the space of all paths with k vertices. For convenience and without losing generality, we ignore the fixed vertices at the laser and sensor device in our formulae.

In practice, $\boldsymbol{H}$ is obtained by the spatio-temporal integration of transient measurements during a time interval τ, which accounts for the contribution of all paths $\bar{\mathbf{x}}$ with time of flight

$$t = \mathrm{tof}(\bar{\mathbf{x}}) = \sum\nolimits_{i=1}^{k} \frac{||\mathbf{x}_i - \mathbf{x}_{i-1}||}{c}, \tag{4}$$

where c is the speed of light, $\mathbf{x}_0 \equiv \mathbf{x}_l$, and $\mathbf{x}_k \equiv \mathbf{x}_s$. We assume no scattering delays at the vertices.

Incorporating the third-bounce strategy of NLOS reconstruction methods in our path integral formulation, we can express $\mathcal{K}$ in a closed form as

$$\mathcal{K}(\bar{\mathbf{x}}, \mathbf{t}) = \Lambda(\mathbf{x}_l \rightarrow \mathbf{x}_g, t_l)\rho(\mathbf{x}_g)\mathfrak{T}(\bar{\mathbf{x}}, \mathbf{t})\Phi(\mathbf{x}_g \rightarrow \mathbf{x}_s, \mathrm{tof}(\bar{\mathbf{x}})), \tag{5}$$

where Λ is the emitted light from the laser, Φ is the time-dependent sensor sensitivity function, ρ represents surface reflectance, and $\mathfrak{T}(\bar{\mathbf{x}}, \mathbf{t})$ is the path throughput defined by

$$\mathfrak{T}(\bar{\mathbf{x}}, \mathbf{t}) = V(\mathbf{x}_l, \mathbf{x}_g)\frac{|\cos\theta_1||\cos\theta_2|}{d_{lg}^2} V(\mathbf{x}_g, \mathbf{x}_s)\frac{|\cos\theta_3||\cos\theta_4|}{d_{gs}^2}, \tag{6}$$

where V is the binary visibility function between two vertices, $d_{lg} = \|\mathbf{x}_l - \mathbf{x}_g\|$ and $d_{gs} = \|\mathbf{x}_g - \mathbf{x}_s\|$, and θ_{1-4} refer to the angles between the normals of both the relay wall and surface geometry, and the path segments in $\bar{\mathbf{x}}$ (see Figure 3). Note that the three-bounce illumination is expressed in the path space as $\bar{\mathbf{x}} \equiv \mathbf{x}_l \rightarrow \mathbf{x}_g \rightarrow \mathbf{x}_s$.

Neither the emitted light Λ nor the sensor sensitivity Φ are ideal Dirac delta functions. Yi et al. [2021] and Hernandez et al. [2017] provide the following models for the laser and sensor behavior

$$\Lambda(t) = \frac{I_l}{\sigma_l\sqrt{2\pi}} e^{-t^2/(2\sigma_l^2)}, \tag{7}$$

$$\Phi(t) = \kappa_s e^{-\kappa_s t} * \frac{1}{\sigma_s\sqrt{2\pi}} e^{-(t-\mu_s)^2/(2\sigma_s^2)}, \tag{8}$$

where σ_l is the standard deviation of the Gaussian laser pulse, I_l is the laser intensity, κ_s is the sensor sensitivity decay rate, σ_s is the standard deviation of the sensor jitter, and μ_s is the offset of the sensor jitter. Since we are only interested on reproducing the combined effect of the laser and sensor models Λ and Φ on the path throughput (Equation 6), we replace them by a single joint laser-sensor correction function as

$$\begin{aligned}\Psi(t) &= \Phi(t) * \Lambda(t)\\ &= \kappa_s e^{-\kappa_s t} * \frac{I_l}{\sigma_{ls}\sqrt{2\pi}} e^{-t^2/(2\sigma_{ls}^2)}.\end{aligned} \tag{9}$$

Note that the convolution of the two Gaussian functions of Equations 7 and 8 yields a single Gaussian with a joint model parameter $\sigma_{ls} = \sqrt{\sigma_l^2 + \sigma_s^2}$. We set the sensor jitter offset as $\mu_s = 0$, with the assumption that a uniform distribution of shifts is equally present in all transient measurements. Please refer to the supplemental material for more details on derivation. Our inverse rendering optimization seeks optimal parameters of this model automatically based on physically-based transient rendering.

4 DIFFERENTIABLE TIME-GATED NLOS INVERSE RENDERING

In the following, we describe in detail our self-calibrated, end-to-end differentiable inverse rendering pipeline, where the forward pass provides high-detailed reconstructions of the geometry G, while the backward pass optimizes per-voxel surface reflectance as albedo Θ_G, as well as system parameters Θ_{pf} and Θ_{ls} to improve the forward pass reconstruction. For clarity, from here on, we redefine our functions in terms of their parameters to be optimized. Refer to the supplemental material for a summary of the different symbols.

4.1 Virtual illumination for RSD propagation

The inputs to our system are the known locations of the illumination $\mathbf{x}_l$ and the sensor $\mathbf{x}_s$, a matrix $\boldsymbol{H}$ of transient measurements, and an *arbitrary* virtual illumination function $\mathcal{P}(\Theta_{\mathrm{pf}}) \equiv \mathcal{P}(\mathbf{x}_l, \mathbf{x}_s, \Omega)$ (Equation 1), where Θ_{pf} represents the optimized parameter space for $\mathcal{P}$. Based on previous works [Liu et al. 2020, 2019; Marco et al. 2021], we define $\Theta_{\mathrm{pf}} = \{\sigma_{\mathrm{pf}}, \Omega_{\mathrm{pf}}\}$ to model a central frequency with a zero-mean Gaussian envelope as $\mathcal{P}(\Theta_{\mathrm{pf}}) = e^{i\Omega_{\mathrm{pf}}t} e^{-t^2/(2\sigma_{\mathrm{pf}}^2)}$, where $\sigma_{\mathrm{pf}}, \Omega_{\mathrm{pf}}$ represent the standard deviation and central frequency, respectively. Note that this equation is fully differentiable. In the forward pass we first compute the filtered matrix $\boldsymbol{H}_{\mathrm{pf}}$ (Equation 1) using the optimized virtual illumination $\mathcal{P}(\Theta_{\mathrm{pf}})$, having $\boldsymbol{H}_{\mathrm{pf}} = P(\boldsymbol{H}; \Theta_{\mathrm{pf}})$ (Figure 2a). We then compute a first estimation of the volumetric intensity I_{pf} of the hidden scene by evaluating RSD propagation (Equation 2) at $t = 0$, as $I_{\mathrm{pf}} = RSD(\boldsymbol{H}_{\mathrm{pf}})$. Next, we show how to estimate both the geometry G and the time-resolved transport $\boldsymbol{H}_R$ at the relay wall.

4.2 Implicit surface geometry

Our next goal is to estimate an implicit surface representation G (points $\mathbf{x}_g$ and normals $\mathbf{n}_g$) by means of a differentiable function D as $G = D(I_{\mathrm{pf}})$ (Figure 2b) that takes our volumetric intensity function I_{pf} as input.

We keep an implicit representation of our hidden surface geometry G without creating meshed (explicit) surface geometry during the whole optimization. The key idea is to use the volumetric data computed at each forward pass to estimate *projections* of the geometry (i.e., points and normals) visible from the perspective of each sensor point $\mathbf{x}_s$ on the relay wall and use those to perform path-space differentiable transient rendering at $\mathbf{x}_s$.

We first estimate the geometry observed by $\mathbf{x}_s$ by sampling rays towards our volumetric intensity I_{pf}, and build an implicit representation of the closest surface along each ray. Using information from neighboring rays, we then estimate the normals required to compute the path-space throughput of $\mathfrak{T}$ (Equation 6). Using the implicit geometry computed for every sensing point $\mathbf{x}_s$, we then compute time-resolved illumination at $\mathbf{x}_s$ as we describe later in this subsection.

Points. As Figure 4a shows, for each sensor point $\mathbf{x}_s$ we sample rays uniformly using concentric hemispherical mapping [Shirley

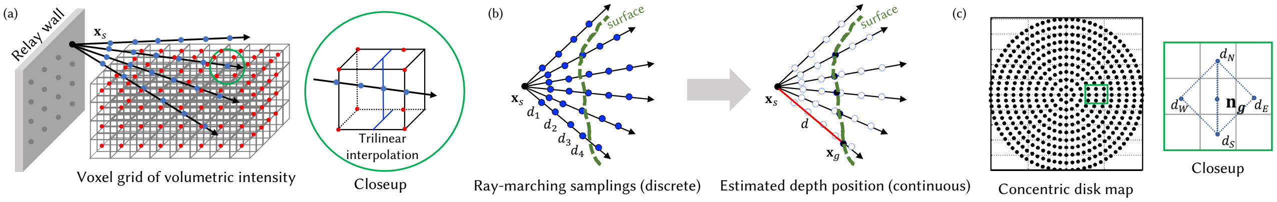

Figure 4: Geometry estimation procedure. (a) We ray-march from sensor points $\mathbf{x}_s$, and estimate the intensity at each point along the ray by trilinear interpolation of I_{pf}. (b) From the discrete ray-marching samplings, we obtain a continuous depth function. (c) Normals are computed based on the distances at neighboring ray samples in the concentric hemispherical mapping.

and Chiu 1997]. We then sample points along each ray with ray marching, and estimate the intensity at each sampled point (blue in Figure 4a) by trilinear interpolation of neighbor voxel intensities of I_{pf} (red). From the interpolated volumetric intensities $I_{\mathrm{pf}}(d_i)$ (Figure 4b, left), we estimate the distance d_{gs} between $\mathbf{x}_s$ and the hidden surface vertex $\mathbf{x}_g$ (Figure 4b, right), assuming $\mathbf{x}_g$ is located at the maximum intensity along the ray. To find d_{gs} in free space from the ray-marched intensities in a differentiable manner, we use `softargmax` function: $d_{gs} = \frac{\sum_i \omega_i d_i}{\sum_i \omega_i}$, where d_i is a ray-marched distance from $\mathbf{x}_s$, and $\omega_i = e^{\beta I_{\mathrm{pf},i}}$ is a probability density function of d_i, and $I_{\mathrm{pf},i}$ is the volume intensity at distance d_i along the ray. β is a hyperparameter that determines the sensitivity in blending neighboring probabilities, set to 1e+3 in all our experiments. If I_{pf} falls below a threshold, we assume that no surface has been found; we set this threshold to 0.05 for synthetic scenes, and 0.2 for real scenes throughout the paper. Our procedure implicitly estimates surface points $\mathbf{x}_g$ at distances $d = \left\|\mathbf{x}_s - \mathbf{x}_g\right\|$ by observing via ray marching the grid of phasor-field intensities I_{pf} from the perspective of the sensing points $\mathbf{x}_s$.

Normals. As shown in Figure 4c, we estimate the normal $\mathbf{n}_g$ at vertex $\mathbf{x}_g$ based on the distances d_N, d_S, d_E, d_W at neighboring ray samples in the concentric hemispherical mapping. We compute the normals of two triangles $\triangle d_N d_E d_S$ and $\triangle d_S d_W d_N$ via two edges' cross product and compute $\mathbf{n}_g$ as the normalized sum of the normals of those two triangles.

Surface albedo. Besides points and normals—updated implicitly during each forward pass—, computing path contribution $\mathcal{K}$ (Equation 5) at sensor points $\mathbf{x}_s$ requires computing per-point monochromatic albedo ρ. We estimate albedos by evaluating the physical observation of the transient measurements in the backward pass.

4.3 Differentiable transient rendering

The next step during the forward pass is to obtain time-resolved illumination $\boldsymbol{H}_R$ at $\mathbf{x}_s$ through transient rendering. In our pipeline (Figure 2c), we represent this step as $\boldsymbol{H}_R = R(G; \Theta_{\mathrm{G}}, \Theta_{\mathrm{ls}})$, where $R()$ computes third-bounce time-resolved light transport at sensing points $\mathbf{x}_s$. We use the rays sampled from $\mathbf{x}_s$ (Figure 4b) to compute the radiometric contribution $\mathcal{K}(\bar{\mathbf{x}}, \mathbf{t})$ of the implicit surface points $\mathbf{x}_g$ estimated by those rays, following Equations 5 through 9.

Visibility. Differentiating the binary visibility function V, necessary to compute the path throughput $\mathfrak{T}$ (Equation 6), is challenging. However, note that we estimate an implicit surface at $\mathbf{x}_g$ based on volumetric intensities, which strongly depend on the illumination from the laser reaching the surface and going back to the sensor without finding any occluder. Based on this, we avoid computing the visibility term by assuming the volumetric intensities are a good estimator of the geometry visible from the perspective of both laser and sensor positions on the relay wall.

Transient rendering. The radiometric contribution $\mathcal{K}(\bar{\mathbf{x}}, \mathbf{t})$ (Equation 5) yields time-resolved transport in path space for a single path $\bar{\mathbf{x}} \equiv \mathbf{x}_l \rightarrow \mathbf{x}_g \rightarrow \mathbf{x}_s$. Our goal is to obtain a set of discrete transient measurements $\boldsymbol{H}_R$ from all paths arriving at each sensing point $\mathbf{x}_s$, such that $\boldsymbol{H}_R$ is comparable to the captured matrix $\boldsymbol{H}$. To this end, we first discretize $|\mathcal{K}(\bar{\mathbf{x}}, \mathbf{t})|$ into neighboring bins τ using a differentiable Gaussian distribution function as $\hat{\mathcal{K}}(\bar{\mathbf{x}}, \tau) = |\mathcal{K}(\bar{\mathbf{x}}, \mathbf{t})| \exp\left(-\frac{(\tau-t)^2}{2\sigma_t^2}\right)$, where τ is a transient bin index, t is continuous time of $\bar{\mathbf{x}}$ (Equation 4), and σ_t is set to 0.62 to make the FWHM of the Gaussian distribution cover a unit time bin.

The time-resolved measurement $\boldsymbol{H}_r(\mathbf{x}_l, \mathbf{x}_s, \tau)$ at temporal index τ is then approximated as the sum of the discrete path contributions $\hat{\mathcal{K}}(\bar{\mathbf{x}}, \tau)$ sampled through the concentric disk mapping as

$$\boldsymbol{H}_r(\mathbf{x}_l, \mathbf{x}_s, \tau) \approx \sum_{\bar{\mathbf{x}} \in \mathcal{X}} \hat{\mathcal{K}}(\bar{\mathbf{x}}, \tau), \tag{10}$$

where $\mathcal{X}$ is the set of paths $\bar{\mathbf{x}}$ that start at $\mathbf{x}_l$ and end in $\mathbf{x}_s$. After generating the rendered transient data $\boldsymbol{H}_r$, we then apply our joint laser-sensor model to it to obtain a sensed transient data $\boldsymbol{H}_R$:

$$\boldsymbol{H}_R(\mathbf{x}_l, \mathbf{x}_s, \tau) = \Psi(\tau) * \boldsymbol{H}_r(\mathbf{x}_l, \mathbf{x}_s, \tau) + \eta_s \tag{11}$$

where η_s is the intensity offset parameter that takes the ambient light and the dark count rate of the sensor into account.

4.4 Optimization of system parameters

Our final goal is to estimate the system parameters $\Theta = \{\Theta_{\mathrm{pf}}, \Theta_{\mathrm{ls}}, \Theta_G\}$ that minimize the loss between the measured matrix $\boldsymbol{H}$ and the rendered matrix $\boldsymbol{H}_R$ (Figure 2, red). We define this as

$$\min_{\Theta} \mathcal{L}(\boldsymbol{H}, \boldsymbol{H}_R), \tag{12}$$

which we minimize by gradient descent. The transient cost function $\mathcal{L}$ consists of a data term and regularization terms as

$$\mathcal{L}(\boldsymbol{H}, \boldsymbol{H}_R) = E_H + E_{I_{\mathrm{pf}}} + E_\rho. \tag{13}$$

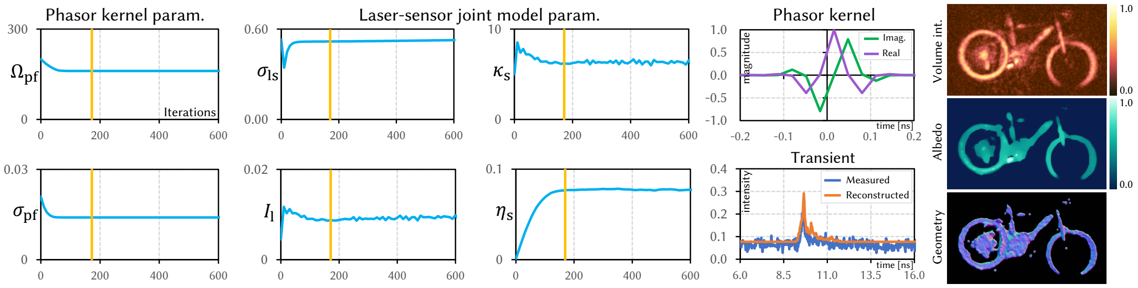

Figure 5: Convergence of the imaging parameters optimized by our method in the Bike real scene. From left to right: Phasor kernel parameters (Ω_{pf}, σ_{pf}), laser-sensor joint model parameters (σ_{ls}, I_{l}, κ_{s}, η_{s}), the converged phasor kernel (purple and green for real and imaginary parts), measured transients compared to our reconstructed one, and our reconstruction results after the optimization. The yellow line indicates when the optimization converges. The converged phasor kernel yields a high-quality reconstruction, while the laser and sensor parameters provide an accurate estimation of transient illumination.

The data term E_H computes an l_2 norm between the transient measurements $\boldsymbol{H}$ and $\boldsymbol{H}_R$:

$$E_H = \frac{1}{N_H} \sum_i \left\| \boldsymbol{H}_i - \boldsymbol{H}_{R,i} \right\|_2^2, \tag{14}$$

where N_H is the total number of elements of $\boldsymbol{H}$. The key insight of this loss term is that $\boldsymbol{H}_R$ is the byproduct of time-resolved illumination computed from our implicit geometry G, which was itself generated from volumetric intensities I_{pf} by means of RSD propagation of the ground truth $\boldsymbol{H}$. The difference between $\boldsymbol{H}$ and $\boldsymbol{H}_R$ is therefore a critical measure of the accuracy of geometry G and I_{pf}. By backpropagating the loss term through our pipeline, we optimize all system parameters, which improve the estimation of I_{pf}, G and therefore $\boldsymbol{H}_R$.

The term $E_{I_{\text{pf}}}$ in Equation 13 is a volumetric intensity regularization term that imposes sparsity, pursuing a clean image:

$$E_{I_{\text{pf}}} = \lambda_1 \frac{1}{N_{\text{pf,z}}} \sum_j \left| I_{\text{pf,z},j} \right|, \tag{15}$$

where $I_{\text{pf,z}}$ is the maximum intensity values of I_{pf} projected to the xz plane, $N_{\text{pf,z}}$ is the number of pixels of $I_{\text{pf,z}}$, and λ_1 is a loss-scale balance hyperparameter, which is set to 1e+2 in all our experiments.

The term E_ρ in Equation 13 is a regularization term that imposes smoothness, suppressing surface reflectance noise:

$$E_\rho = \lambda_2 \frac{1}{N_v} \sum_m \left| \nabla_{xy} \rho(\mathbf{x}_{v,m}) \right|, \tag{16}$$

where N_v is the number of voxels $\mathbf{x}_v$, and λ_2 is a loss-scale balance hyperparameter, which is set to 5e-3 in all our experiments. All terms E_H, $E_{I_{\text{pf}}}$, and E_ρ of the loss function are computed over batches of the transients and voxels at every iteration.

5 RESULTS

We implement our pipeline using PyTorch. Our code runs on an AMD 7763 CPU of 2.45 GHz equipped with a single NVIDIA GPU A100. 3D geometry is obtained from points and normals using Poisson surface reconstruction [Kazhdan and Hoppe 2013]. Please note that we do not perform any thresholding or masking of the data prior to this step. We evaluate our method on four real confocal datasets Bike, Resolution, SU, and 34, provided by O'Toole et al. [2018], Ahn et al. [2019] and Lindell et al. [2019]; on two real non-confocal datasets 44i and NLOS, provided by Liu et al. [2019]; and on four synthetic confocal datasets Erato, Bunny, Indonesian and Dragon, generated with the transient renderer by Chen et al. [2020]. The real datasets include all illumination bounces and different levels of noise depending on their exposure time. The synthetic datasets include up-to third-bounce illumination. In specific cases, we manually add Poisson noise to synthetic datasets to evaluate our robustness to signal degradation.

Table 1: Ablation study of the impact of each component. MSE transient loss comparison with different configurations with the Bunny scene with two different albedos (Figure 8).

Component			MSE
Phasor kernel	Albedo	Laser-sensor model	transient
✔	–	–	6.817e-3
✔	–	✔	6.627e-3
–	✔	–	2.239e-3
–	✔	✔	2.217e-3
✔	✔	–	2.124e-3
✔	✔	✔	1.971e-3

5.1 Convergence of system parameters

In Figure 5, we show the convergence of our system parameters in a full optimization of the Bike real scene, showing as well the final reconstruction of both volumetric intensity and geometry. Phasor-field kernel parameters Ω_{pf} and σ_{pf} (first column) are responsible for improving the reconstruction quality by constructing a phasor kernel (fourth column, top) that yields high-detailed geometry. The laser and sensor parameters (second and third columns) improve the reconstruction of the transient measurements so that the transient simulation (fourth column, bottom, orange) resembles as much as possible the input data (blue). Refer to the supplemental material for more results of the progressive optimization.

We evaluate the impact of each component in our optimization pipeline: phasor kernel, albedo, and laser-sensor model, using a 256 × 256 × 201 voxel volume. As Table 1 shows, adding albedo and laser-sensor parameters improves the result over just using the

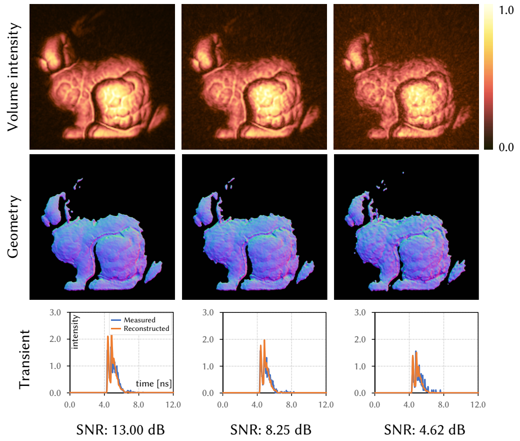

Figure 6: Evaluation of our surface reconstruction under increasing levels of Poisson noise (left to right). From top to bottom: intensity volume, reconstructed geometry, and measured vs. optimized transport. Our method reconstructs geometry reliably across a broad spectrum of noise levels. A lower signal-to-noise ratio (SNR) value indicates a higher level of noise, with an exponential increase in noise.

phasor parameters, while including the three components yields the best results. The impact of optimizing albedo is the most significant in this experiment.

5.2 Robustness to noise

To illustrate the robustness of our method to signal degradation, in Figure 6 we show reconstructions of the Bunny synthetic dataset under increasing levels of Poisson noise (from left to right) applied to the input transient data. The first row shows the final volumetric reconstruction after the optimization, while the second row shows the resulting surface estimation. The third row shows a comparison between the input transient illumination (blue) and our converged transient illumination at the same location that results from our estimated geometry (orange). The parameters optimized by our pipeline produce a volumetric reconstruction robust enough for our surface estimation method to obtain a reliable 3D geometry under a broad spectrum of noise levels. Note that while the volumetric outputs may show noticeable noise levels (first row), our pipeline optimizes the imaging parameters so that such volumetric outputs provide a good baseline for our geometry estimation method, which yields surface reconstructions that consistently preserve geometric details across varying noise levels (second row).

In Figure 7, we compare our method with existing volumetric approaches on two real confocal scenes, Resolution and Bike, captured under different exposure times. For each scene, first to fourth columns illustrate the compared methods: O'Toole et al. [2018], Lindell et al. [2019], Liu et al. [2020], and ours, respectively. First to fourth rows show the resulting volumetric intensity images under increasing exposure times of 10, 30, 60, and 180 minutes, respectively. Our method converges to imaging parameters that produce the sharpest results while significantly removing noise even under the lowest exposure time (top row). Other methods degrade notably at lower exposure times, failing to reproduce details in the resolution chart, or yielding noisy outputs in the Bike scene.

While LCT [O'Toole et al. 2018] allows to manually select an SNR filtering parameter α to improve results in low-SNR conditions, our experiments with different α values from 0.001 to 1.0 at different exposure levels validate that our automated calibration approach outperforms the LCT method, reproducing detailed geometric features (see supplemental material).

5.3 Inverse rendering

Our optimization pipeline estimates surface points, normals, and albedo by using only the input transient measurements. Figure 8 illustrates our volumetric intensity, as well as surface points, normals and albedo in the confocal synthetic scene Bunny made of two different surface albedos 1.0 (top) and 0.3 (bottom). Our method is consistent when estimating spatially-varying albedo, while not affecting the estimation of detailed surface points and normals.

Figure 9 demonstrates our inverse rendering results on real scenes. As shown in a confocal scene SU (first row) and two non-confocal scenes 44i (second row) and NLOS (third row), we correctly estimate the albedo of objects with uniform reflectance properties (second column), although they undergo different attenuation factors due to being at different distances from the relay wall. The result of the NLOS non-confocal scene (third row) shows the albedo throughout the entire surface is almost identical. Our estimation of surface points and normals (third and fourth columns) is able to accurately reproduce the structure of the hidden geometry.

In Figure 1, we illustrate the benefits of our inverse rendering optimization on the real scene Bike. The first row shows the first iteration of the optimization, which uses the volumetric output by Liu et al. [2020] with the default parameters of the illumination function. The resulting noise heavily degrades the geometry and normal estimation (top-right), and the albedo is wrongly estimated at empty locations in the scene despite the lack of a surface at such locations (top center). After our optimization converges (bottom row), the albedo is estimated only at surface locations, yielding a clean reconstruction of the bike's surface points and normals.

5.4 Geometry accuracy

In Figure 10, we compare the reconstructed geometry with surface normals in two real scenes (34 and SU) using D-LCT [Young et al. 2020], NeTF [Shen et al. 2021], a differentiable rendering approach [Plack et al. 2023], and our method. Existing methods fail to reproduce detailed surface features in both scenes, such as the subtle changes in depth of the numbers. Plack's method (fourth column) fails to reproduce the partially occluded U-shaped object and some regions of the S-shaped object in the SU scene. D-LCT (second column) succeeds in reproducing the U-shaped object but fails to reconstruct the detailed geometry of the boundary of the letters. While NeTF [Shen et al. 2021] (third column) is capable of reproducing the U-shaped object, their methodology, based on positional encoding and neural rendering, suppresses geometric details significantly, producing a coarse geometry. Plack's method faces similar challenges in reproducing geometric details due to the constraints imposed by the resolution of the explicit proxy geometry. Previous optimization-based methods that also rely on explicit geometry [Iseringhausen and Hullin 2020; Tsai et al. 2019] share similar limitations. Our method based on implicit surface representations is able to handle partial occlusions while reproducing detailed features of the surfaces, such as the depth changes on the numbers and the narrow segments of the letters.

In Figure 11, we provide quantitative comparisons between our estimated geometry and the geometry obtained from D-LCT [Young et al. 2020], NeTF [Shen et al. 2021] and Plack et al. [2023] for three synthetic scenes, Dragon, Erato, and Indonesian, using the Hausdorff distance map as an objective metric. In terms of geometric accuracy, we outperform all three methods in Erato, and Dragon, as shown in the RMSE table. Our improvements are especially noticeable in self-occluded regions and in the reproduction of detailed features. While Plack et al. [2023] yields a lower RMSE in the Indonesian scene, note that it fails to reproduce large regions on the sides of the geometry. Thus, RMSE is only computed on the reconstructed regions and may not fully represent the overall accuracy of the reconstruction.

6 DISCUSSION AND FUTURE WORK

We have presented an efficient and fully-differentiable end-to-end NLOS inverse rendering pipeline, which self-calibrates the imaging parameters using only the input-measured transient illumination. Our method is robust in the presence of noise while achieving enhanced scene reconstruction accuracy.

Even though forward automatic differentiation (AD) is known to be memory efficient, we implemented our pipeline using reverse AD, as we found it to be 20 times faster and showed better performance when optimizing a large number of parameters (such as per-voxel albedo), and supports a wider set of differentiable functions required for our context.

Phasor-field NLOS imaging can be performed analogously using temporal- or frequency-domain operators [Liu et al. 2020, 2019]. However, operating in the temporal domain introduces large memory constraints that are impractical on a differentiable pipeline. Our pipeline therefore operates in the frequency domain to perform NLOS imaging, which provides practical implementation of convolutions of complex-valued phasor-field kernels within GPU memory constraints. While we based volumetric NLOS imaging on phasor-based operators and kernels, an interesting avenue of future work may be optimizing alternative kernel parameterizations or implementing other differentiable NLOS imaging approaches.

ACKNOWLEDGMENTS

We want to thank the anonymous reviewers for their time and insightful comments. Min H. Kim acknowledges the main support of the Samsung Research Funding Center (SRFC-IT2001-04), in addition to the additional support of the MSIT/IITP of Korea (RS-2022-00155620, 2022-0-00058, and 2017-0-00072), Samsung Electronics, and the NIRCH of Korea (2021A02P02-001). This work was also partially funded by the Gobierno de Aragón (Departamento de Ciencia, Universidad y Sociedad del Conocimiento) through project BLINDSIGHT (ref. LMP30_21), and by MCIN/AEI/10.13039/501100011033 through Project PID2019-105004GB-I00.

REFERENCES

Byeongjoo Ahn, Akshat Dave, Ashok Veeraraghavan, Ioannis Gkioulekas, and Aswin C Sankaranarayanan. 2019. Convolutional approximations to the general non-line-of-sight imaging operator. In *Proc. International Conference on Computer Vision (ICCV)*. 7889–7899.

Victor Arellano, Diego Gutierrez, and Adrian Jarabo. 2017. Fast Back-Projection for Non-Line of Sight Reconstruction. *Optics Express* 25, 10 (2017).

Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. 2021. Mip-NeRF: A Multiscale Representation for Anti-Aliasing Neural Radiance Fields. *ICCV* (2021).

Mauro Buttafava, Jessica Zeman, Alberto Tosi, Kevin Eliceiri, and Andreas Velten. 2015. Non-line-of-sight imaging using a time-gated single photon avalanche diode. *Opt. Express* 23, 16 (2015).

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. 2015. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012* (2015).

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).

Justin Dove and Jeffrey H. Shapiro. 2020a. Nonparaxial phasor-field propagation. *Opt. Express, OE* 28, 20 (Sept. 2020), 29212–29229. https://doi.org/10.1364/OE.401203 Publisher: Optical Society of America.

Justin Dove and Jeffrey H. Shapiro. 2020b. Speckled speckled speckle. *Opt. Express, OE* 28, 15 (July 2020), 22105–22120. https://doi.org/10.1364/OE.398226 Publisher: Optical Society of America.

Daniele Faccio, Andreas Velten, and Gordon Wetzstein. 2020. Non-line-of-sight imaging. *Nature Reviews Physics* 2, 6 (2020), 318–327.

Yuki Fujimura, Takahiro Kushida, Takuya Funatomi, and Yasuhiro Mukaigawa. 2023. NLOS-NeuS: Non-line-of-sight Neural Implicit Surface. *arXiv* preprint arXiv:2303.12280v2 (2023).

Genevieve Gariepy, Nikola Krstajić, Robert Henderson, Chunyong Li, Robert R Thomson, Gerald S Buller, Barmak Heshmat, Ramesh Raskar, Jonathan Leach, and Daniele Faccio. 2015. Single-photon sensitive light-in-fight imaging. *Nature Communications* 6 (2015).

Javier Grau Chopite, Matthias B. Hullin, Michael Wand, and Julian Iseringhausen. 2020. Deep Non-Line-of-Sight Reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ibón Guillén, Xiaochun Liu, Andreas Velten, Diego Gutierrez, and Adrian Jarabo. 2020. On the Effect of Reflectance on Phasor Field Non-Line-of-Sight Imaging. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 9269–9273.

Otkrist Gupta, Thomas Willwacher, Andreas Velten, Ashok Veeraraghavan, and Ramesh Raskar. 2012. Reconstruction of hidden 3D shapes using diffuse reflections. *Opt. Express* 20, 17 (2012).

Felix Heide, Matthew Oâ€™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.

Quercus Hernandez, Diego Gutierrez, and Adrian Jarabo. 2017. A Computational Model of a Single-Photon Avalanche Diode Sensor for Transient Imaging. *arXiv* preprint arXiv:1703.02635 (2017).

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

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).

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.

Michael Kazhdan and Hugues Hoppe. 2013. Screened poisson surface reconstruction. *ACM Transactions on Graphics (TOG)* 32, 3 (2013), 29.

Marco La Manna, Fiona Kine, Eric Breitbach, Jonathan Jackson, Talha Sultan, and Andreas Velten. 2018. Error backprojection algorithms for non-line-of-sight imaging. *IEEE transactions on pattern analysis and machine intelligence* 41, 7 (2018), 1615–1626.

Martin Laurenzis and Andreas Velten. 2014. Feature selection and back-projection algorithms for nonline-of-sight laser–gated viewing. *Journal of Electronic Imaging* 23, 6 (2014), 063003.

Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. 2018. Differentiable monte carlo ray tracing through edge sampling. *ACM Transactions on Graphics (TOG)* 37, 6 (2018), 1–11.

Zhengpeng Liao, Deyang Jiang, Xiaochun Liu, Andreas Velten, Yajun Ha, and Xin Lou. 2021. FPGA Accelerator for Real-Time Non-Line-of-Sight Imaging. *IEEE Transactions on Circuits and Systems I: Regular Papers* (2021).

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), 1–13.

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 Fields Virtual Wave Optics. *Nature* (2019).

Julio Marco, Adrian Jarabo, Ji Hyun Nam, Xiaochun Liu, Miguel Ángel Cosculluela, Andreas Velten, and Diego Gutierrez. 2021. Virtual light transport matrices for non-line-of-sight imaging. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV*.

Fangzhou Mu, Sicheng Mo, Jiayong Peng, Xiaochun Liu, Ji Hyun Nam, Siddeshwar Raghavan, Andreas Velten, and Yin Li. 2022. Physics to the Rescue: Deep Non-line-of-sight Reconstruction for High-speed Imaging. In *IEEE Conference on Computational Photography (ICCP)*.

Ji Hyun Nam, Eric Brandt, Sebastian Bauer, Xiaochun Liu, Marco Renna, Alberto Tosi, Eftychios Sifakis, and Andreas Velten. 2021. Low-latency time-of-flight non-line-of-sight imaging at 5 frames per second. *Nature communications* 12, 1 (2021), 1–10.

Michael Niemeyer, Jonathan T. Barron, Ben Mildenhall, Mehdi S. M. Sajjadi, Andreas Geiger, and Noha Radwan. 2022. RegNeRF: Regularizing Neural Radiance Fields for View Synthesis from Sparse Inputs. In *CVPR*.

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.

Adithya Pediredla, Ashok Veeraraghavan, and Ioannis Gkioulekas. 2019. Ellipsoidal path connections for time-gated rendering. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–12.

Markus Plack, Clara Callenberg, Monika Schneider, and Matthias B Hullin. 2023. Fast Differentiable Transient Rendering for Non-Line-of-Sight Reconstruction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 3067–3076.

Syed Azer Reza, Marco La Manna, Sebastian Bauer, and Andreas Velten. 2019. Phasor field waves: experimental demonstrations of wave-like properties. *Opt. Express* 27, 22 (Oct. 2019), 32587. https://doi.org/10.1364/OE.27.032587

Diego Royo, Jorge García, Adolfo Muñoz, and Adrian Jarabo. 2022. Non-line-of-sight transient rendering. *Computers & Graphics* 107 (2022), 84–92. https://doi.org/10.1016/j.cag.2022.07.003

Guy Satat, Barmak Heshmat, Nikhil Naik, Albert Redo-Sanchez, and Ramesh Raskar. 2016. Advances in Ultrafast Optics and Imaging Applications. In *SPIE Defense+ Security*.

Siyuan Shen, Zi Wang, Ping Liu, Zhengqing Pan, Ruiqian Li, Tian Gao, Shiying Li, and Jingyi Yu. 2021. Non-line-of-sight imaging via neural transient fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43, 7 (2021), 2257–2268.

Peter Shirley and Kenneth Chiu. 1997. A low distortion map between disk and square. *Journal of graphics tools* 2, 3 (1997), 45–52.

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.

Andreas Velten, Thomas Willwacher, Otkrist Gupta, Ashok Veeraraghavan, Moungi G. Bawendi, and Ramesh Raskar. 2012. Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging. *Nature Communications* 3 (2012).

Lifan Wu, Guangyan Cai, Ravi Ramamoorthi, and Shuang Zhao. 2021. Differentiable time-gated rendering. *ACM Transactions on Graphics (TOG)* 40, 6 (2021), 1–16.

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 *IEEE Computer Vision and Pattern Recognition (CVPR)*. 6800–6809.

Shinyoung Yi, Donggun Kim, Kiseok Choi, Adrian Jarabo, Diego Gutierrez, and Min H. Kim. 2021. Differentiable Transient Rendering. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia 2021)* 40, 6 (2021).

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*.

Shuang Zhao, Wenzel Jakob, and Tzu-Mao Li. 2020. Physics-based differentiable rendering: from theory to implementation. In *ACM siggraph 2020 courses*. 1–30.

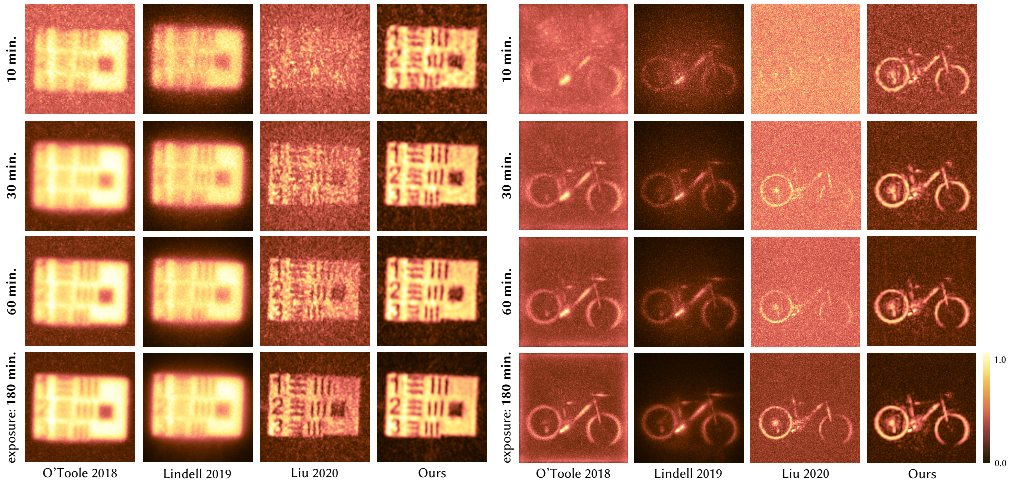

Figure 7: Reconstructed volumetric intensity comparison using the Resolution (left) and Bike (right) real scenes captured under increasing exposure times of 10, 30, 60, and 180 minutes. Existing methods [Lindell et al. 2019; Liu et al. 2020; O'Toole et al. 2018] (first to third columns) fail to reproduce details on the resolution chart across all exposure times, and yield high-frequency noise in the reconstructions due to low SNR in the Bike datasets. Our method (last column) converges to imaging parameters that produce the sharpest results robustly under different exposure times, without requiring manual parameter tuning.

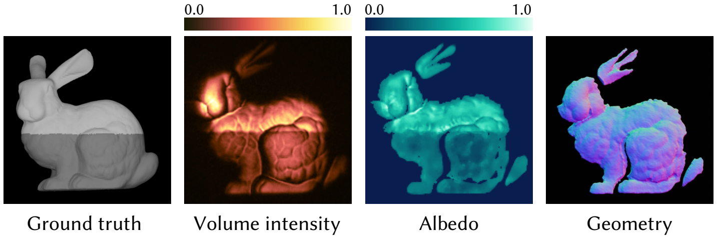

Figure 8: Our optimization scheme estimates spatially-varying albedo in a consistent manner, without affecting the surface and normal estimation. From left to right: Synthetic Bunny scene with two different albedos (0.3 and 1.0), our converged volumetric intensity, the optimized albedo, and the estimated geometry.

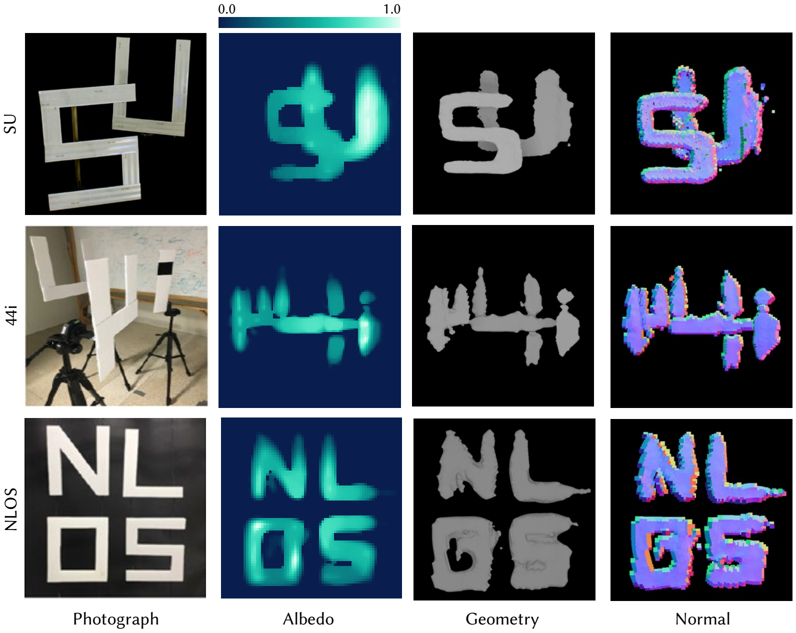

Figure 9: Our inverse rendering results with confocal real scene SU and non-confocal scenes 44i and NLOS. Our approach uses transient measurements to reconstruct surface albedo (second column), geometry (third column), and normals (fourth column), estimating them correctly on multiple isolated objects at different distances.

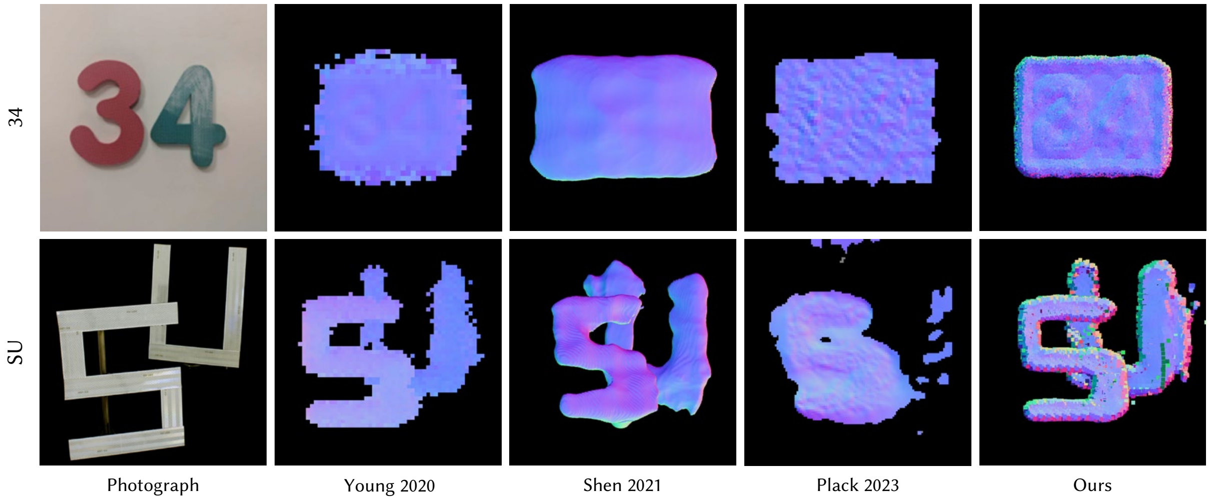

Figure 10: Comparison of the geometry estimation in two real scenes 34 and SU. From left to right: D-LCT [Young et al. 2020], NeTF [Shen et al. 2021], the differential renderer [Plack et al. 2023], and our result. Our method can reconstruct more accurately detailed features such as the depth changes of the numbers, or reproduce narrow segments of the letters, while other methods yield coarse reconstructions or even fail to reproduce partially occluded objects.

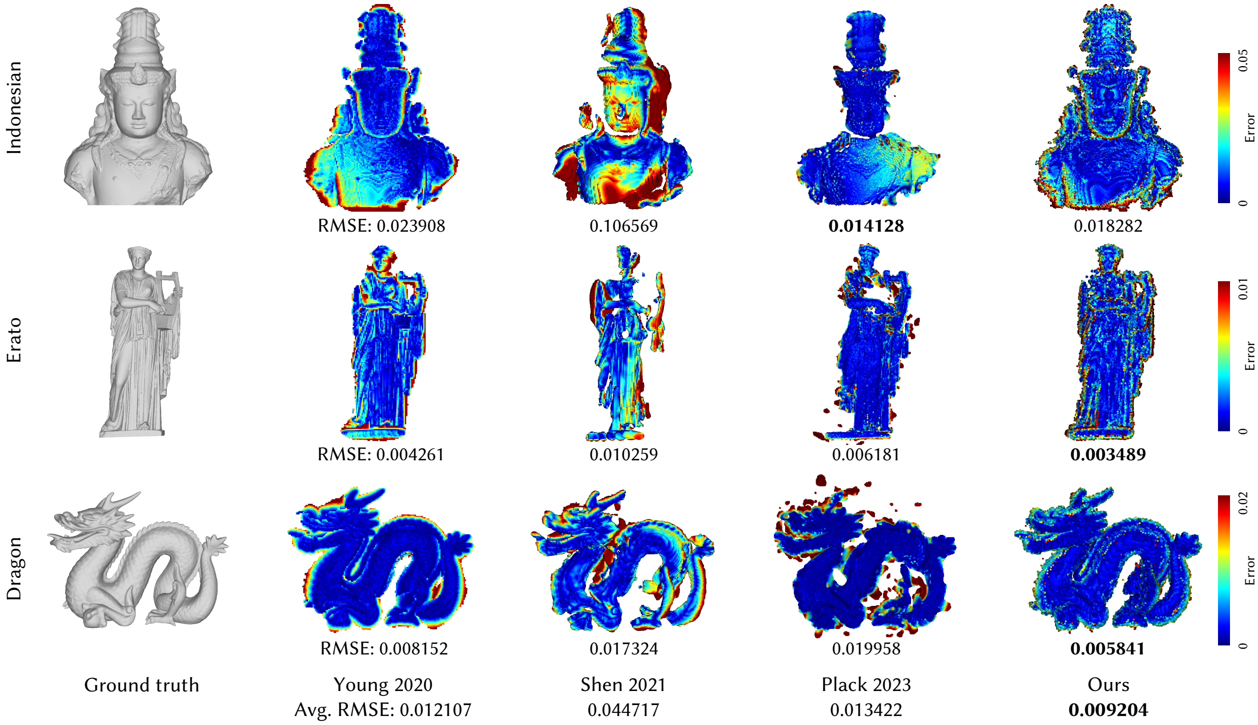

Figure 11: We perform a quantitative comparison of our surface reconstruction with Young et al. [2020], Shen et al. [2021] and Plack et al. [2023] using synthetic transient data with ground truth geometries Dragon, Erato, and Indonesian. We quantify the introduced errors using the Hausdorff distance between the ground truth geometry and the estimated geometries. Our method yields the smallest RMSE in Erato and Dragon, noticeable in highly-detailed areas. Note that while Plack et al. [2023] has smaller RMSE in Indonesian, the reconstructed surface is missing significant regions of the ground truth geometry, which are not quantified by the RMSE.