Neural Inverse Rendering from Propagating Light

Anagh Malik[*1,2]   Benjamin Attal[*3]   Andrew Xie[1,2]   Matthew O'Toole[+3]   David B. Lindell[+1,2]

[1]University of Toronto [2]Vector Institute [3]Carnegie Mellon University

[*]joint first authors [+]equal contribution
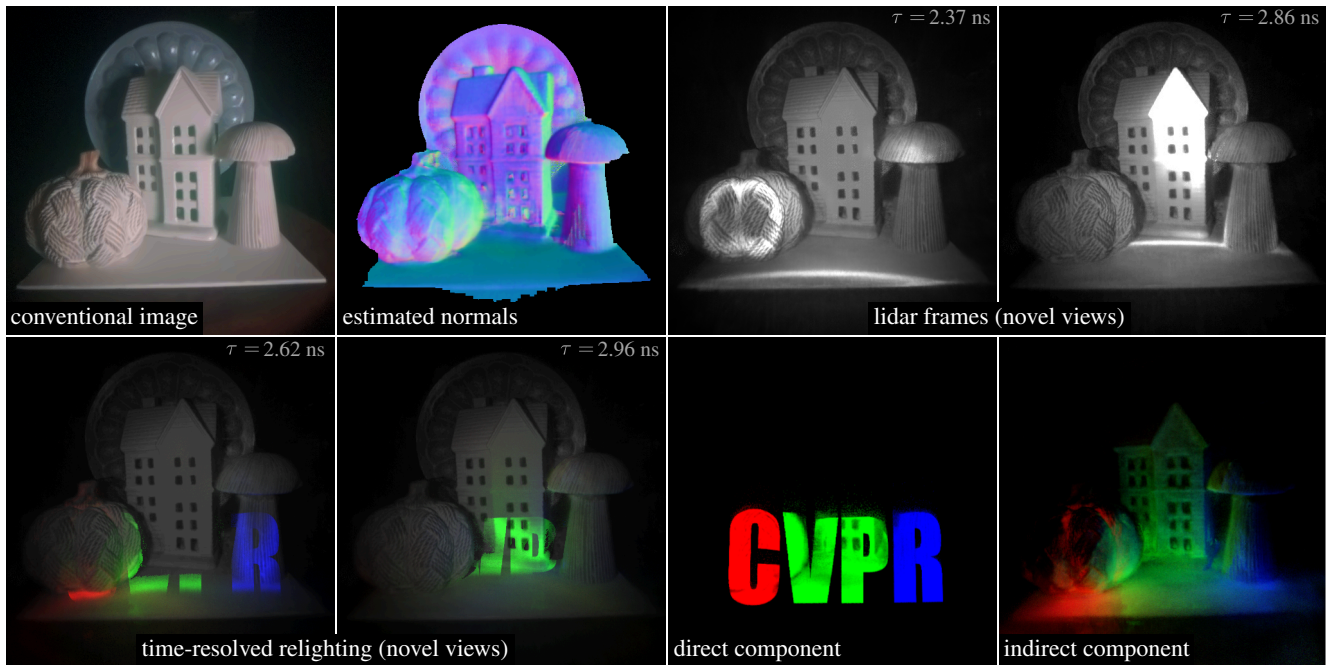
https://anaghmalik.com/InvProp

Figure 1. We introduce a method to model and invert multi-view, time-resolved measurements of propagating light from a flash lidar system. **(row 1)** Our method accurately recovers the geometry of this scene and enables rendering of time-resolved lidar measurements that reveal light propagation from novel views. **(row 2)** Physically-based modeling enables novel applications, such as time-resolved relighting and automatic decomposition of light transport into direct and indirect components.

Abstract

We present the first system for physically based, neural inverse rendering from multi-viewpoint videos of propagating light. Our approach relies on a time-resolved extension of neural radiance caching — a technique that accelerates inverse rendering by storing infinite-bounce radiance arriving at any point from any direction. The resulting model accurately accounts for direct and indirect light transport effects and, when applied to captured measurements from a flash lidar system, enables state-of-the-art 3D reconstruction in the presence of strong indirect light. Further, we demonstrate view synthesis of propagating light, automatic decomposition of captured measurements into direct and indirect components, as well as novel capabilities such as multi-view time-resolved relighting of captured scenes.

1. Introduction

Ultrafast imaging systems such as lidar illuminate a scene with a pulse of light and capture the backscattered "echoes" from the propagating wavefront [73]. Precisely measuring the speed-of-light time delay of backscattered light enables 3D reconstruction, and so lidar systems based on this principle are popular in applications from autonomous driving [36] to augmented reality [2] and remote sensing [13].

Lidar relies on time-resolved measurements of *direct* light transport, or light that reflects directly from a surface back to the sensor. Measurements of *indirect* light transport (i.e., light that scatters multiple times before reaching the sensor) are typically ignored or discarded because modeling indirect light requires computationally expensive (and often intractable) inverse rendering using procedures such as recursive path tracing [25, 64]. Still, measurements of

10534

indirect light are a rich source of information about material properties, appearance, and geometry [32, 55, 61, 86, 97]. Our work seeks to model and invert multi-viewpoint, time-resolved measurements of propagating light from a lidar system to recover scene geometry and to render videos of propagating light from novel viewpoints and under novel illumination conditions (see Figure 1).

Conventional lidar systems pre-process captured time-resolved measurements into a 3D point cloud, which represents an estimate of scene geometry based on direct light transport. While recent work leverages lidar measurements captured from multiple viewpoints to perform 3D reconstruction and novel view synthesis, existing methods use a point cloud representation [12, 21, 70, 94] or direct-only time-resolved measurements [45, 47, 53, 66], and thus ignore indirect light. Other methods for multi-viewpoint reconstruction with active imaging use time-of-flight sensors [3, 58], or structured light [23, 75], but similarly fail to explicitly model indirect light. Our work is close to that of Malik et al. [48], which uses lidar measurements and a representation based on neural radiance fields (NeRFs) [50] to render videos of propagating light from novel viewpoints. However, while their representation is effective for view synthesis, it does not use a physically-based model, and so cannot reconstruct accurate geometry or render the scene under novel illumination conditions.

Accurately modeling indirect light transport requires solving the rendering equation [29]. Many renderers use path tracing algorithms based on Monte Carlo sampling, and differentiable versions of these methods have shown promise for inverse rendering of conventional (i.e., steady state) images [35, 44, 57] and time-resolved measurements [89, 92], such as those from a lidar system. However, while these methods work in constrained problem settings, such as non-line-of-sight imaging [24, 80], they are difficult to deploy on captured multi-viewpoint experiments due to their computational expense and sensitivity to noise and local minima. To address this issue, recent methods approximate the rendering equation in a hybrid fashion using a volumetric representation of geometry and physics-based appearance models [77, 96]. This type of approach has proven especially effective for inverse rendering of appearance, geometry, and material parameters from conventional intensity images [4, 28, 42, 97]. Still, no previous technique performs inverse rendering of time-resolved direct and indirect light transport in the same fashion.

Here, we propose a method for inverse rendering from multi-viewpoint, time-resolved measurements of propagating light from a *flash lidar* system — a type of lidar that flood-illuminates the entire scene with a pulse of light. Our approach is based on a hybrid neural representation, where we model geometry using volume rendering, and appearance using a physically-based model that simulates global

illumination effects using a radiance cache [85]. Specifically, instead of integrating light paths using path tracing, our radiance cache stores a representation of time-resolved radiance arriving at any point in a volume from any direction. The representation is optimized in an amortized fashion, thereby removing the need to evaluate the rendering integral recursively; instead, we need only render direct lighting from the lidar and query the radiance cache to evaluate indirect light at any surface point. Our model builds on previous neural rendering frameworks for multi-viewpoint, time-resolved rendering of the direct [47] and indirect components [48] of propagating light, and grounds them in a framework for physically-based rendering. Overall, we make the following contributions.

- We propose a method for neural inverse rendering of propagating light using a physically-based model with a time-resolved radiance cache.
- We capture a new dataset of multi-viewpoint, time-resolved flash lidar measurements with calibrated light source and camera positions.
- We demonstrate our method in simulation and on captured data; we show state-of-the-art results in geometry reconstruction under strong indirect light transport, and we render videos of propagating light from novel viewpoints and under novel illumination conditions in scenes with varying reflectance properties and significant indirect light transport effects.

2. Related Work

Our approach brings together the areas of time-resolved imaging and rendering, as well as inverse rendering.

Time-resolved imaging. Time-of-flight systems such as lidar measure the time of flight by marking the arrival time of a backscattered pulse of light [33]. These systems usually combine nanosecond or picosecond pulsed lasers with fast photodiodes [22] or single-photon avalanche diodes (SPADs) [31, 69, 76] to measure ultrafast variations in incident light. The resulting time-resolved measurements capture direct and indirect light transport, and can be used to record videos of propagating light at ultrafast timescales [16, 38, 59, 82]. While continuous-wave time-of-flight systems [18, 34] can also be used to capture light propagation, their temporal resolution is more limited (e.g., nanoseconds rather than picoseconds) [20, 59].

Our approach uses a lidar system with a picosecond pulsed laser and a SPAD to capture multi-view videos of propagating light. Similar to previous work [10, 15, 47, 48, 61, 68, 90], we repeatedly illuminate the scene with pulses of light from the laser. The SPAD detects the arrival times of individual photons with picosecond-level accuracy and outputs a photon count histogram that approximates the time-resolved waveform of incident light [60]. Our approach is

the first to capture a multi-view dataset of photon count histograms where both the flash lidar light source and sensor vary in position. Moreover, we develop the first technique for physically-based time-resolved inverse rendering using multi-viewpoint videos of propagating light.

Time-resolved rendering. Time-resolved path-tracing renderers simulate wavefronts of propagating light [27] and account for effects such as birefringence [57], refraction [63], and volumetric scattering [26, 62]. Recently, differentiable versions of these time-resolved renderers [89, 92] have been developed; however, robust analysis-by-synthesis scene reconstruction using these methods is an open problem due to computational complexity and sensitivity to initialization and noise. Other renderers approximate the time-resolved light transport matrix [59, 67] for specific imaging problems such as non-line-of-sight imaging. In this area, techniques have been developed to model two-bounce [32, 79], three-bounce [39, 41, 68, 74, 90], or higher-order scattering events [37, 72] to recover occluded geometry. In the non-line-of-sight setting, renderers usually assume that light reflects off of planar surfaces with known (usually diffuse) reflectance properties, or else model specific scattering paths rather than arbitrary light transport [15]. Although several works consider indirect light transport in the context of continuous-wave time-of-flight sensors, they usually treat it as a residual to be removed when estimating depth [1, 18, 56]. We perform physically-based modeling of multiply scattered light without restrictive assumptions on scene geometry or material properties and integrate our approach into an inverse rendering framework for scene reconstruction from captured measurements.

Physically-based inverse rendering. Inverse rendering aims to recover scene attributes, like materials, lighting, and geometry, from a set of images [11, 65, 93]. Existing techniques use a physically-based model of light transport [29, 64, 81] and differentiable rendering with gradient-based optimization to decompose the scene's appearance into its constituent attributes [10, 25, 35]. Recent techniques for physically-based rendering using NeRFs [50] have made inverse rendering considerably more robust, but either only consider direct illumination [7, 8, 77, 84, 96], or require explicitly simulating multiple light bounces to model indirect light [46], which is computationally expensive. Another approach is to use radiance caches — data structures that store the hemisphere of incoming radiance at every point [85]. This hemisphere is then integrated against the local bidirectional reflectance distribution function (BRDF) [64] to yield outgoing radiance, which is optimized to match the observed image pixels. Combining the radiance cache with NeRFs leads to more efficient modeling of indirect light [4, 42, 87, 91, 97].

Finally, we note that the concurrent work of Wu et al. [88] makes use of radiance-caching-based inverse rendering with a collocated point light source and color camera. However, no previous technique performs physically-based inverse rendering from multi-viewpoint time-resolved measurements. To this end, we develop a new, time-resolved radiance cache, enabling neural inverse rendering from videos of propagating light.

3. Background: Radiance Caching with NeRFs

The rendering equation [29], models the outgoing radiance of light in direction $\boldsymbol{\omega}_{\mathrm{o}}$ at a point $\mathbf{x}$ along a ray $\mathbf{x}(t) = \mathbf{o} - t\boldsymbol{\omega}_{\mathrm{o}}$ with origin $\mathbf{o}$ and ray parameter t:

$$L_{\mathrm{o}}(\mathbf{x}(t), \boldsymbol{\omega}_{\mathrm{o}}) = \int_{\Omega} f(\mathbf{x}, \boldsymbol{\omega}_{\mathrm{i}}, \boldsymbol{\omega}_{\mathrm{o}}) L_{\mathrm{i}}(\mathbf{x}, \boldsymbol{\omega}_{\mathrm{i}})(\mathbf{n} \cdot \boldsymbol{\omega}_{\mathrm{i}}) \, d\boldsymbol{\omega}_{\mathrm{i}} . \quad (1)$$

The equation integrates the incident radiance L_{i} arriving to $\mathbf{x}$ from direction $\boldsymbol{\omega}_{\mathrm{i}}$ weighted by the BRDF f. The integral is over the positive hemisphere with respect to the normal $\mathbf{n}$: $\Omega = \{\boldsymbol{\omega}_{\mathrm{i}} : \mathbf{n} \cdot \boldsymbol{\omega}_{\mathrm{i}} > 0\}$. Naive evaluation of the rendering equation leads to an exponential increase in computation since the equation must be evaluated recursively to compute the incident radiance.

To avoid this computational penalty, we leverage radiance caching, which removes the problematic recursion by replacing incident radiance L_{i} in the rendering equation with a look-up into a cache L_{i}^{cache}:

$$L_{\mathrm{o}}(\mathbf{x}(t), \boldsymbol{\omega}_{\mathrm{o}}) = \int_{\Omega} f(\mathbf{x}, \boldsymbol{\omega}_{\mathrm{i}}, \boldsymbol{\omega}_{\mathrm{o}}) L_{\mathrm{i}}^{cache}(\mathbf{x}, \boldsymbol{\omega}_{\mathrm{i}})(\mathbf{n} \cdot \boldsymbol{\omega}_{\mathrm{i}}) \, d\boldsymbol{\omega}_{\mathrm{i}} . \quad (2)$$

The integral can be efficiently approximated, e.g., by sampling the cache and the BRDF using multiple importance sampling [64].

Recent work demonstrates that NeRFs provide accurate modeling of the radiance cache [4, 28, 40]. Specifically, we can compute $L_{\mathrm{i}}^{cache}(\mathbf{x}, \boldsymbol{\omega}_{\mathrm{i}})$ by volume rendering the NeRF along a secondary ray $\mathbf{x}'(t) = \mathbf{o}' - t\boldsymbol{\omega}_{\mathrm{o}}'$, where $\mathbf{o}' = \mathbf{x}$ and $\boldsymbol{\omega}_{\mathrm{o}}' = \boldsymbol{\omega}_{\mathrm{i}}$ [14, 43, 50]:

$$L_{\mathrm{i}}^{cache}(\mathbf{o}', \boldsymbol{\omega}_{\mathrm{i}}) = \sum_{k=1}^{N} w_k \, L_{\mathrm{o}}^{cache}(\mathbf{x}'(t_k), \boldsymbol{\omega}_{\mathrm{o}}') . \quad (3)$$

Here, L_{o}^{cache} is the outgoing radiance at each point along the secondary ray predicted by the NeRF. The values w_k are quadrature weights that account for the transmittance and absorption along the ray, calculated as a function of the density σ at each sample point $\mathbf{x}'(t_k)$ and the ray interval $(\Delta t)_k$ [49, 78]:

$$w_k = \left(1 - e^{-\sigma(\mathbf{x}'(t_k))(\Delta t)_k}\right) e^{\left(-\sum_{j=1}^{k-1} \sigma(\mathbf{x}'(t_j))(\Delta t)_j\right)} . \quad (4)$$

Our work adapts this observation to time-resolved rendering based on lidar measurements.
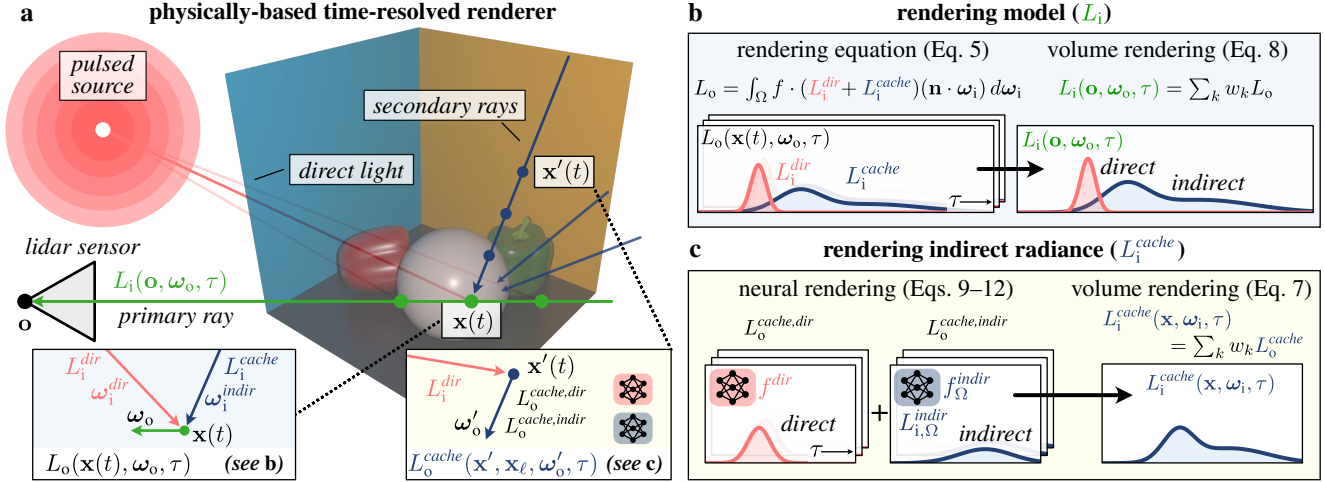
Figure 2. Method overview. **(a)** Our time-resolved renderer combines physically-based rendering for primary rays *(left inset)*, and neural rendering for an indirect radiance cache along secondary rays *(right inset)*. **(b)** The incident radiance L_i at a sensor pixel is a function of the outgoing radiance L_o from each point $\mathbf{x}(t)$ along a sensor ray, which integrates incident direct light L_i^{dir} and indirect light L_i^{cache} from a pulsed laser source. **(b,** *left)* The rendering equation computes the outgoing radiance as an integral over the positive hemisphere Ω (with respect to the normal $\mathbf{n}$) of the incident radiance. **(b,** *right)* Applying volume rendering to the outgoing radiance yields the incident radiance at a sensor pixel. **(c)** The radiance cache is used to evaluate indirect radiance L_i^{cache} by casting secondary rays and querying neural networks at points $\mathbf{x}'(t)$. **(c,** *left)* The networks output the reflectance f^{dir} of direct light L_i^{dir}, indirect reflectance f_Ω^{indir}, and indirect incident radiance $L_{i,\Omega}^{indir}$. These quantities are used to calculate the time-resolved direct ($L_o^{cache,dir}$) and indirect ($L_o^{cache,indir}$) outgoing radiance in the direction ω_o'. **(c,** *right)* Volume rendering the outgoing radiance along a secondary ray yields $L_i^{cache}(\mathbf{x}, \omega_i, \tau)$. We optimize the scene appearance and geometry to enforce consistency between rendered and captured measurements.

4. Method

We model and invert time-resolved light transport, including direct and indirect effects, to recover scene geometry and material properties. Our method uses a physically-based time-resolved renderer with a time-resolved radiance cache parameterized by neural networks, as shown in Figure 2. We perform inverse rendering by optimizing the representation using measurements of propagating light from a flash lidar system.

4.1. Physically-Based Time-Resolved Rendering

We model a lidar measurement by casting a primary ray $\mathbf{x}(t) = \mathbf{o} - t\omega_o$ into the scene. For each point along that ray, we render the outgoing radiance, L_o, in the direction of the sensor. Our time-resolved rendering equation is a modified version of Equation 1, where we add the time of flight τ as

$$L_o(\mathbf{x}(t), \omega_o, \tau)$$
$$= \int_\Omega f(\mathbf{x}(t), \omega_i, \omega_o) L_i(\mathbf{x}(t), \omega_i, \tau)(\mathbf{n} \cdot \omega_i)\, d\omega_i. \quad (5)$$

The reflectance f, is modeled using the Disney–GGX BRDF [9], which depends on the material properties of the scene as described in the supplement.

We further decompose the incident radiance into two components $L_i = L_i^{dir} + L_i^{cache}$: a direct component L_i^{dir}, and an indirect component L_i^{cache} evaluated using the radiance cache. The direct component is given as

$$L_i^{dir}(\mathbf{x}(t), \omega_i, \tau) = \frac{\delta(\omega_\ell - \omega_i) L_i^\ell\left(\omega_\ell, \tau - \frac{||\mathbf{x}(t) - \mathbf{x}_\ell||}{c}\right)}{||\mathbf{x}(t) - \mathbf{x}_\ell||^2}, \quad (6)$$

where $\mathbf{x}_\ell$ is the position of the light source, ω_ℓ is the direction from the light source, L_i^ℓ is the light source intensity, and $\delta(\omega_\ell - \omega_i)$ is a Dirac delta function. We model the inverse-square law intensity falloff and the time delay to position $\mathbf{x}(t)$ based on the speed of light c.

Similar to the steady-state case (Equation 3), the time-resolved radiance cache $L_i^{cache}(\mathbf{x}, \omega_i, \tau)$ is evaluated using secondary rays $\mathbf{x}'(t) = \mathbf{o}' - t\omega_o'$ cast from points on the primary ray $\mathbf{o}' = \mathbf{x}$ with $\omega_o' = \omega_i$. We use time-resolved volume rendering [3, 17, 47] to render the radiance cache as

$$L_i^{cache}(\mathbf{o}', \omega_i, \tau) = \sum_{k=1}^N w_k\, L_o^{cache}\left(\mathbf{x}'(t_k), \mathbf{x}_\ell, \omega_o', \tau - \frac{t_k}{c}\right), \quad (7)$$

where L_o^{cache} is the outgoing radiance predicted by a neural representation. The above states that the light incident at $\mathbf{o}'$ in direction ω_i at time τ is the sum of delayed copies of the light leaving each point $\mathbf{x}'$ along ω_o', and the delay depends on the distance to $\mathbf{o}'$ (given by the ray parameter t_k).

After evaluating the time-resolved rendering equation (Equation 5) for each point on the primary ray, the lidar measurement is computed by volume rendering the outgoing radiance in the same way as for the cache:

$$L_i(\mathbf{o}, \omega_o, \tau) = \sum_{k=1}^N w_k\, L_o\left(\mathbf{x}(t_k), \omega_o, \tau - \frac{t_k}{c}\right). \quad (8)$$

4.2. Time-Resolved Radiance Cache

Representation. The cache is parameterized using a multi-resolution hash encoding $\mathcal{H}^{app}$ [5, 54] to learn a position-dependent appearance feature $\mathbf{f}^{app}$. Similarly, a hash encoding-based neural network $\mathcal{N}^{geom}$ represents scene geometry through density and normals. That is,

$$\mathbf{f}^{app} = \mathcal{H}^{app}(\mathbf{x}), \quad \sigma, \mathbf{n} = \mathcal{N}^{geom}(\mathbf{x}). \quad (9)$$

The density values used for volume rendering are shared across both the physically-based model and the radiance cache (i.e., Equations 7 and 8). The appearance features are used to compute the radiance cache, which we decompose into direct and indirect components as

$$L_{\mathrm{o}}^{cache} = L_{\mathrm{o}}^{cache,dir} + L_{\mathrm{o}}^{cache,indir}. \quad (10)$$

We describe each of these components as follows.

Direct light. The direct component is due to light that is emitted from the lidar source at $\mathbf{x}_\ell$, propagates to a point $\mathbf{x}'$ in the scene, and scatters directly to the ray origin $\mathbf{o}'$:

$$L_{\mathrm{o}}^{cache,dir}(\mathbf{x}'(t), \mathbf{x}_\ell, \boldsymbol{\omega}'_{\mathrm{o}}, \tau)$$
$$= f^{dir}(\mathbf{f}^{app}, \mathbf{n}, \boldsymbol{\omega}_\ell, \boldsymbol{\omega}'_{\mathrm{o}}) \, L_{\mathrm{i}}^{dir}(\mathbf{x}'(t), \boldsymbol{\omega}_\ell, \tau)(\mathbf{n} \cdot \boldsymbol{\omega}_\ell). \quad (11)$$

Here, f^{dir} is a neural network that learns the BRDF (see the supplement for a detailed description). In practice, we discretize the equation and compute a vector that represents radiance at each time interval.

Indirect light. We leverage a split-sum approximation [30] to efficiently cache outgoing indirect light:

$$L_{\mathrm{o}}^{cache,indir}(\mathbf{x}'(t), \mathbf{x}_\ell, \boldsymbol{\omega}'_{\mathrm{o}}, \tau)$$
$$\approx \int_\Omega f(\mathbf{x}', \boldsymbol{\omega}'_{\mathrm{i}}, \boldsymbol{\omega}'_{\mathrm{o}})(\mathbf{n} \cdot \boldsymbol{\omega}'_{\mathrm{i}}) \, d\boldsymbol{\omega}'_{\mathrm{i}} \cdot \int_\Omega L_{\mathrm{i}}(\mathbf{x}', \mathbf{x}_\ell, \boldsymbol{\omega}'_{\mathrm{i}}, \tau) d\boldsymbol{\omega}'_{\mathrm{i}}$$
$$= f_\Omega^{indir}(\mathbf{f}^{app}, \mathbf{n}, \boldsymbol{\omega}'_{\mathrm{o}}) \cdot L_{\mathrm{i},\Omega}^{indir}(\mathbf{f}^{app}, \mathbf{x}_\ell, \mathbf{n}, \boldsymbol{\omega}'_{\mathrm{o}}), \quad (12)$$

where f_Ω^{indir} and $L_{\mathrm{i},\Omega}^{indir}$ are neural networks that predict the integrated BRDF and integrated incident radiance, respectively. We include conditioning on $\mathbf{x}_\ell$, as the indirect light depends on light source position. Following Malik et al. [48], $L_{\mathrm{i},\Omega}^{indir}$ predicts a vector that represents radiance over discretized time intervals.

4.3. Inverse Rendering from Propagating Light

The lidar system captures the time-resolved incident radiance at the sensor L_{i}^{meas}. We optimize the representation by minimizing the difference between the lidar measurements and the output L_{i} of the physically-based renderer (we omit their dependence on $\mathbf{o}$, $\boldsymbol{\omega}_{\mathrm{o}}$, and τ for brevity):

$$\mathcal{L}_{data} = \sum_{\mathbf{o}, \boldsymbol{\omega}_{\mathrm{o}}, \tau} \alpha(L_{\mathrm{i}}^{cache}) \|L_{\mathrm{i}} - L_{\mathrm{i}}^{meas}\|^2. \quad (13)$$

As the cache can also be used to render the time-resolved incident radiance at the sensor, we supervise it in the same fashion by minimizing $\mathcal{L}_{cache}$, which replaces L_{i} with L_{i}^{cache} in the above. The function α is chosen to more strongly penalize errors in darker regions, which improves perceptual quality similar to applying a tonemapping curve [51]:

$$\alpha(L) = \left| \sum_\tau L \right|^{-\beta}, \quad (14)$$

where β is a hyperparameter. Note that in Equation 13, we compute this weight using the incoming radiance from the cache (L_{i}^{cache} instead of L_{i}), as it is not affected by Monte Carlo render noise.

Following Hadadan et al. [19], we leverage a radiometric prior that constrains the direct and indirect light rendered using the cache to be consistent with the full physically-based model. Specifically, we render the direct and indirect components of radiance at sample points $\mathbf{x}$ along the primary ray, using the cache ($L_{\mathrm{o}}^{cache,dir/indir}$) and the physically-based model (i.e., $L_{\mathrm{o}}^{dir/indir}$, given by evaluating Equation 5 using only L_{i}^{dir} or L_{i}^{cache}, respectively). We constrain the cache using the loss:

$$\mathcal{L}_{dir/indir} = \sum_{\mathbf{x}, \boldsymbol{\omega}_{\mathrm{o}}, \tau} \alpha(L_{\mathrm{o}}^{cache,dir/indir}) \|L_{\mathrm{o}}^{cache,dir/indir} - L_{\mathrm{o}}^{dir/indir}\|^2. \quad (15)$$

Finally, the complete photometric loss function is

$$\mathcal{L}_{data} + \lambda_{cache}\mathcal{L}_{cache} + \lambda_{dir}\mathcal{L}_{dir} + \lambda_{indir}\mathcal{L}_{indir}, \quad (16)$$

where λ_{cache}, λ_{dir}, and λ_{indir} are hyperparameters that weigh each loss component. By minimizing this loss function, the method recovers a material model for the scene (parameterized using the Disney–GGX model) and the scene geometry, normals, and appearance parameters. In addition to the above photometric loss, we include a regularizer $\mathcal{L}_{normals}$ that ties predicted normals to analytic normals from the density field [83]; a smoothness penalty $\mathcal{L}_{geom}$ on the analytic normals; a smoothness penalty $\mathcal{L}_{mat}$ on the predicted BRDF parameters; proposal resampling and distortion losses $\mathcal{L}_{interlevel}$ and $\mathcal{L}_{distortion}$ as in Zip-NeRF [5]; and a mask loss $\mathcal{L}_{mask}$.

We use multiple importance sampling for secondary rays based on the BRDF and a learnable importance sampler for incident illumination as in Attal et al. [4]. The learnable importance sampler is supervised with a loss $\mathcal{L}_{vMF}$. We represent the time-resolved direct outgoing light as a one-hot vector, where each bin corresponds to a discrete time interval, and indirect light as a dense vector of the same size (where the size depends on the dataset). A complete description is provided in the supplement.
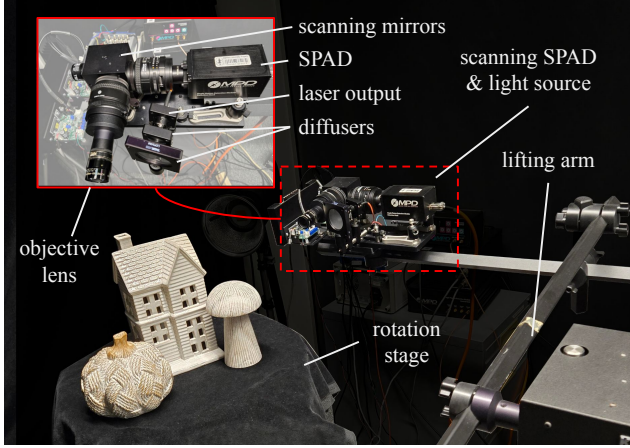
Figure 3. Multi-view capture setup. An elevation arm controls the elevation angle and a rotation stage controls the azimuth angle of the scanning SPAD. Laser light is out-coupled from an optical fiber through a collimating lens and diffusers onto the scene.

5. Results

We evaluate our system on three tasks: (1) view synthesis of time-resolved lidar measurements, (2) view synthesis of integrated (steady-state) lidar images, and (3) geometry reconstruction.

Evaluation metrics. To assess the rendered integrated lidar images, we use PSNR, SSIM, and LPIPS [95]. To evaluate the accuracy of the recovered time-resolved measurements, we use the transient intersection-over-union (transient IOU) introduced by Malik et al. [48]. We use mean absolute error (MAE) and L1 error to measure the accuracy of the recovered normals and depth, respectively. We report quantitative results for both simulated and captured scenes in Table 1.

Baselines. We compare our method to the following state-of-the-art time-resolved neural rendering techniques: T-NeRF [47], which uses a neural representation and a rendering model for time-resolved lidar measurements (but which only accounts for the direct component of light), and Flying with Photons (FWP) [48], which predicts time-resolved radiance at every point in space. Since the original version of FWP does not model light sources or movement of light sources, we use a modified version (FWP++) equivalent to our full, time-resolved radiance cache. We implement both baselines using the same hash-encoding-based neural representation [54], and we use the same regularizers and hyperparameters for fairness.

5.1. Simulated Results

Dataset. We use a modification [71] of the Mitsuba 2 renderer [57] to render multi-view transient data for three small-scale, object-centric synthetic scenes, *Cornell box*, *pots*, *peppers*, as well as one room scale scene, *kitchen* [6].

Table 1. Evaluation of lidar rendering from novel viewpoints and geometry recovery. Each dataset (*sim* and *real*) contains 4 scenes.

	method	PSNR (dB)↑	LPIPS↓	SSIM↑	MAE↓	L1 depth↓	T-IOU↑
sim	T-NeRF [47]	22.44	0.40	0.71	28.00	0.59	0.58
	FWP++ [48]	29.00	0.30	0.87	22.80	0.47	0.73
	ours	30.99	0.31	0.89	8.45	0.21	0.76
cap	T-NeRF [47]	14.67	0.53	0.35	—	—	0.23
	FWP++ [48]	28.45	0.32	0.81	—	—	0.55
	ours	27.39	0.33	0.80	—	—	0.54

Comparison. In Figure 4, we show integrated lidar scans from novel viewpoints, recovered normals, and individual lidar frames for the *peppers* and *pots* scenes. Since T-NeRF [47] only models direct light, it fails to recover accurate geometry under strong indirect light from specular reflections and diffuse inter-reflections. In particular, it introduces floating artifacts to explain these effects and hence fails to predict novel views accurately.

On the other hand, FWP++ [48] models both direct and indirect radiance and recovers accurate integrated novel views and lidar frames. However, because it does not use a physically-accurate rendering model, it overfits to the data. Specifically, it is free to use a mirror copy of the scene to explain specular reflections (e.g., on the floor of the *peppers* scene or the partially specular blue walls in the *pots* scene). Further, it uses incorrect depths to explain diffuse inter-reflections (e.g., at the wall corners in the *peppers* scene or along the fluting of the column in the *pots* scene).

5.2. Captured Results

Dataset. We use a hardware setup (Figure 3) similar to that of Malik et al. [48] to capture a multi-viewpoint flash lidar dataset. Unlike Malik et al. [48], our light source position moves with the camera's viewpoint rather than being stationary with respect to the scene. We capture three scenes: *globe*, *house*, and *spheres*, and we use the *statue* scene from Malik et al. [48], which shows that our method can handle stationary light sources. We provide more details about capture and calibration in the supplement.

Comparison. Visually, the captured results follow a similar trend to the simulated ones. Figure 5 shows that we recover more accurate geometry than FWP++, especially for areas where indirect light is present, such as the corners of the walls in the *globe* scene or the bottom of the candles in the *statue* scene.

Table 1 provides quantitative results for novel view synthesis of lidar measurements. While we find that FWP++ shows slight improvements over our approach for view synthesis, we hypothesize that this is because it uses a far less constrained model—our approach is physically grounded and could thus be more sensitive to the calibration of the physical system and model mismatch. Note that we omit depth and normal metrics for the captured experiments, as there is no ground truth reference.
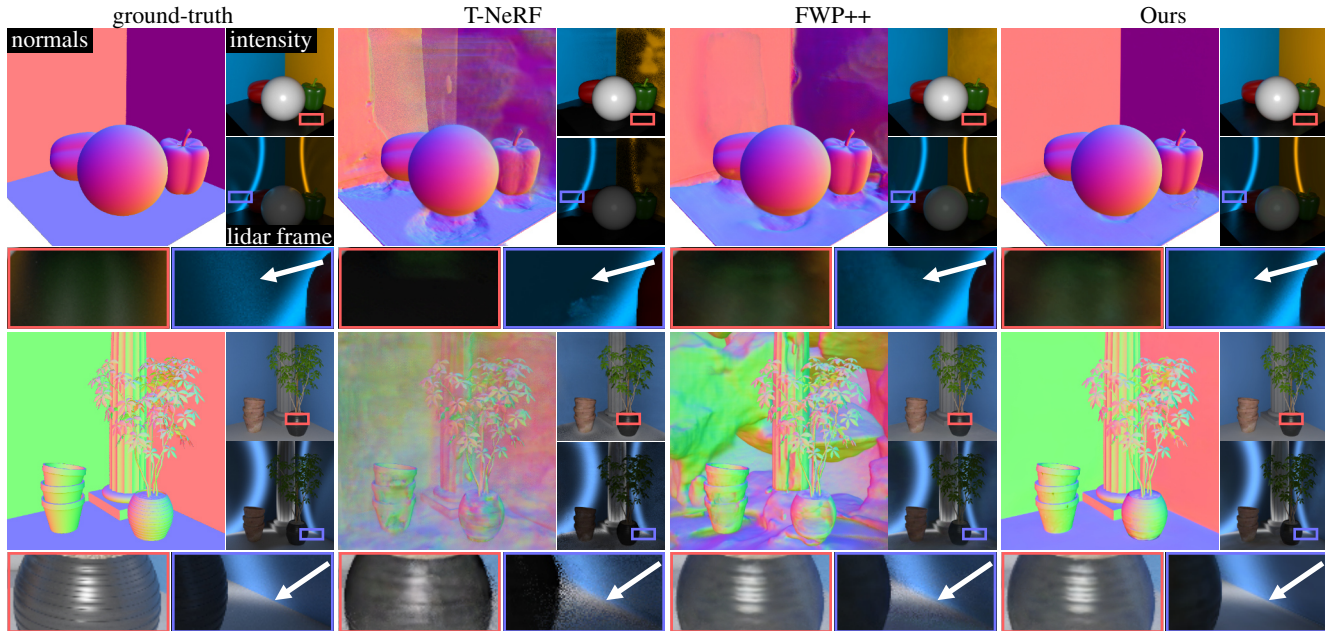
Figure 4. Simulated results. Compared to the baselines our method recovers more accurate normals and similar or improved intensity images due to physically-based modeling of time-resolved indirect light transport (see arrows in the lidar frame insets).
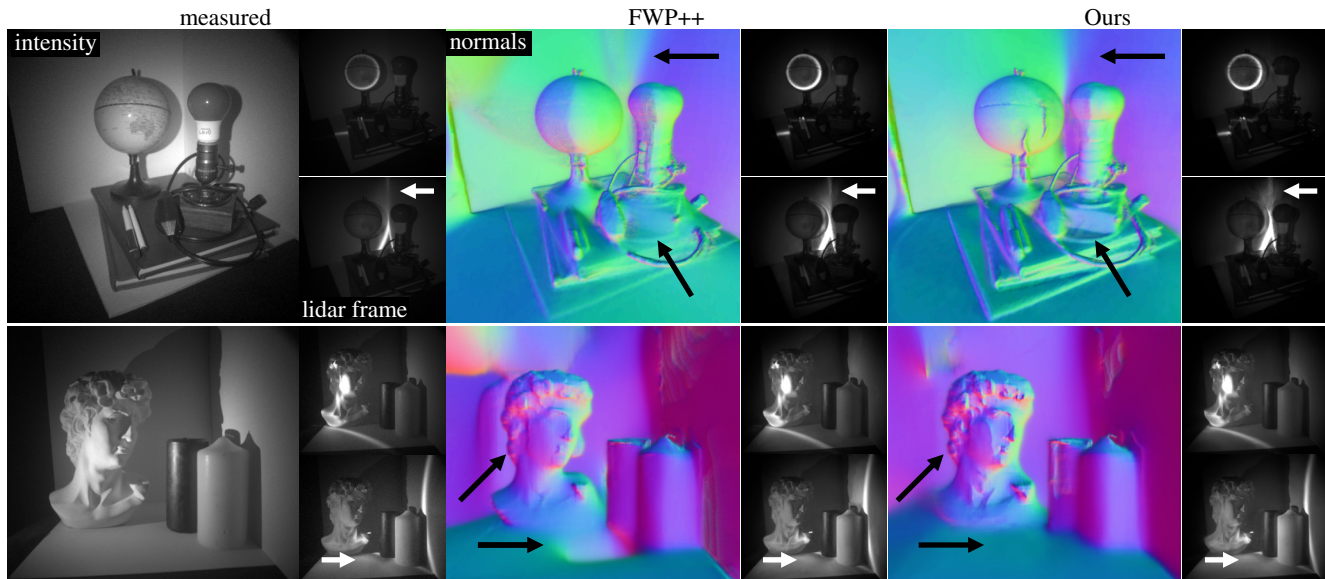

Figure 5. Results on the captured dataset. Our method recovers more accurate normals compared to FWP++ (cols. 3, 5) due to its physically-based modeling of indirect light transport effects (visible in the individual lidar frames; cols. 4, 6). Areas where FWP++ predicts incorrect normals usually correspond to regions with indirect light (arrows).

5.3. Additional Results

Relighting. Our approach enables time-resolved view synthesis and relighting, which, to our knowledge, has not been demonstrated before from captured multi-view, time-resolved measurements. Figure 1 shows relighting results on a novel view from the captured *house* scene with a simulated pulsed source that projects a pattern with the letters "C-V-P-R". In Figure 6, we show two additional examples

of time-resolved relighting: (1) the same *house* scene with three different light sources that converge on the house, and (2) the simulated *kitchen* scene, relit with three point sources that emit pulse trains.

Note that the indirect component of the radiance cache is predicted using a neural network conditioned on the light source position (Equation 12). So, if the intensity profile of the light source used for relighting differs from the training
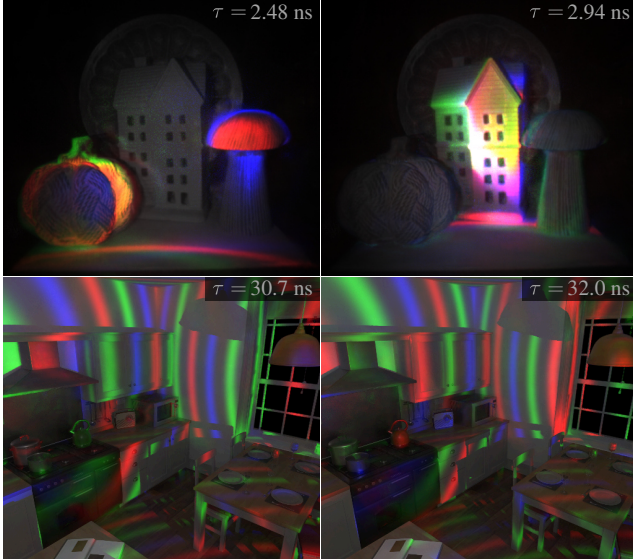
Figure 6. Relighting results using three different light sources (color-coded in the RGB channels) for the captured *house* scene (top) and the simulated *kitchen* scene (bottom).
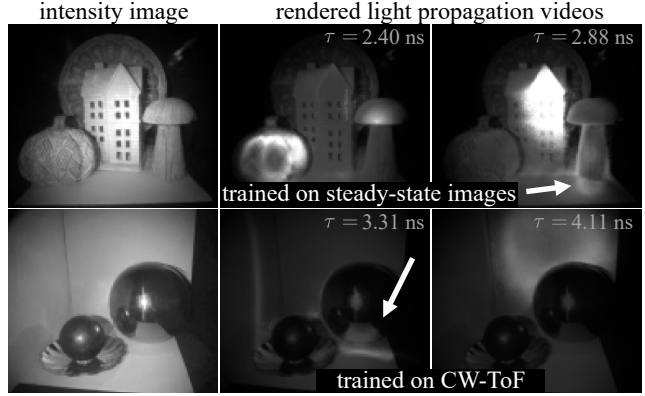

Figure 7. Our approach recovers time-resolved videos of propagating light after training on continuous-wave time-of-flight (CW-ToF) measurements or intensity images. The video frames show direct and indirect light transport effects (arrows).

6. Discussion

data (e.g., a projector instead of a uniform point source as in Figure 1), we use fine-tuning with the radiometric loss $\mathcal{L}_{dir/indir}$ of Equation 15 and the desired light source profile (additional details are provided in the supplement).

Time-resolved imaging without lidar. Although our system recovers time-resolved light transport, it does not necessarily require supervision with lidar measurements. Notably, we can train our model using continuous-wave time-of-flight (CW-ToF) measurements or even intensity images, as both can be derived from time-resolved data. We demonstrate this capability by generating time-resolved videos of propagating light based on each input type. Figure 7 shows the rendered time-resolved measurements after training with either CW-ToF measurements (emulated by convolving the lidar measurements with the CW-ToF illumination waveform) or intensity images (emulated by integrating the lidar measurements over time). We also demonstrate recovery of direct and indirect light transport effects from the ToF measurements and the steady-state images. For example, in *house*, we recover diffuse inter-reflections under the mushroom, and in *spheres*, we recover reflections from the ground to the specular sphere (arrows in Figure 7).

For this application, we only modify the supervision of the model. Specifically, we convert the time-resolved predictions to emulate the CW-ToF or steady-state measurements before calculating the loss. After optimization, we directly render the time-resolved video frames.

Material decomposition. Our method also recovers the material parameters for the Disney–GGX BRDF (albedo, roughness, and metalness). We visualize these parameters for synthetic and captured scenes in the supplement.

Limitations. As noted in the results section, our method relies on a more constrained physical model than other approaches, including the FWP++ baseline. As such, we notice some performance degradation compared to baselines on captured data, where model mismatch is a potential issue. This problem could perhaps be mitigated through better calibration of the physical setup. Additionally, our method requires more than one day of optimization on a single GPU due to the time-consuming physical light transport simulation, I/O penalties from loading large time-resolved measurement vectors, and GPU memory bandwidth requirements. To address this, it may be possible to use a different neural representation that does not predict the entire time-resolved vector, but instead predicts and supervises the signal at a single time instant. Using faster neural representations, like 3D Gaussian Ray Tracing [52] or EVER [46], is also an interesting direction.

Impact and future applications. Although we do not tackle the problem of non-line-of-sight imaging in this work, it is in principle possible to extend our framework for this application in unconstrained conditions, such as non-line-of-sight imaging with non-planar relay surfaces [39]. Due to the widespread use of lidar technologies, we believe our work has potential for impact in areas such as autonomous navigation or remote sensing — especially in scenarios with strong indirect lighting effects.

References

[1] Amit Adam, Christoph Dann, Omer Yair, Shai Mazor, and Sebastian Nowozin. Bayesian time-of-flight for realtime shape, illumination and albedo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(5):851–864, 2016. 3

[2] Apple Inc. Apple vision pro, 2024. Accessed: 2024-11-04. 1

[3] Benjamin Attal, Eliot Laidlaw, Aaron Gokaslan, Changil Kim, Christian Richardt, James Tompkin, and Matthew O'Toole. TöRF: Time-of-flight radiance fields for dynamic scene view synthesis. *Proc. NeurIPS*, 2021. 2, 4

[4] Benjamin Attal, Dor Verbin, Ben Mildenhall, Peter Hedman, Jonathan T Barron, Matthew O'Toole, and Pratul P Srinivasan. Flash cache: Reducing bias in radiance cache based inverse rendering. In *Proc. ECCV*, 2024. 2, 3, 5

[5] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Zip-NeRF: Anti-aliased grid-based neural radiance fields. In *Proc. ICCV*, 2023. 5

[6] Benedikt Bitterli. Rendering resources, 2016. https://benedikt-bitterli.me/resources/. 6

[7] Mark Boss, Varun Jampani, Raphael Braun, Ce Liu, Jonathan T. Barron, and Hendrik P.A. Lensch. Neural-pil: Neural pre-integrated lighting for reflectance decomposition. In *Proc. NeurIPS*, 2021. 3

[8] Mark Boss, Andreas Engelhardt, Abhishek Kar, Yuanzhen Li, Deqing Sun, Jonathan T. Barron, Hendrik P.A. Lensch, and Varun Jampani. SAMURAI: Shape And Material from Unconstrained Real-world Arbitrary Image collections. In *Proc. NeurIPS*, 2022. 3

[9] Brent Burley and Walt Disney Animation Studios. Physically-based shading at Disney. In *ACM SIGGRAPH Courses*, 2012. 4

[10] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proc. CVPR*, 2019. 2, 3

[11] Paul Debevec. Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proc. SIGGRAPH*, 1998. 3

[12] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised NeRF: Fewer views and faster training for free. In *Proc. CVPR*, 2022. 2

[13] Pinliang Dong and Qi Chen. *LiDAR remote sensing and applications*. CRC Press, 2017. 1

[14] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. *ACM SIGGRAPH*, 22(4):65–74, 1988. 3

[15] Daniele Faccio, Andreas Velten, and Gordon Wetzstein. Non-line-of-sight imaging. *Nat. Rev. Phys.*, 2(6):318–327, 2020. 2, 3

[16] Genevieve Gariepy, Nikola Krstajić, Robert Henderson, Chunyong Li, Robert R Thomson, Gerald S Buller, Barmak Heshmat, Ramesh Raskar, Jonathan Leach, and Daniele Faccio. Single-photon sensitive light-in-fight imaging. *Nat. Commun.*, 6(1):6021, 2015. 2

[17] Ioannis Gkioulekas, Anat Levin, and Todd Zickler. An evaluation of computational imaging techniques for heterogeneous inverse scattering. In *Proc. ECCV*, 2016. 4

[18] Mohit Gupta, Shree K Nayar, Matthias B Hullin, and Jaime Martin. Phasor imaging: A generalization of correlation-based time-of-flight imaging. *ACM Trans. Graph.*, 34(5): 1–18, 2015. 2, 3

[19] Saeed Hadadan, Geng Lin, Jan Novák, Fabrice Rousselle, and Matthias Zwicker. Inverse global illumination using a neural radiometric prior. In *Proc. SIGGRAPH*, pages 1–11, 2023. 5

[20] Felix Heide, Matthias B Hullin, James Gregson, and Wolfgang Heidrich. Low-budget transient imaging using photonic mixer devices. *ACM Trans. Graph.*, 32(4):1–10, 2013. 2

[21] Shengyu Huang, Zan Gojcic, Zian Wang, Francis Williams, Yoni Kasten, Sanja Fidler, Konrad Schindler, and Or Litany. Neural lidar fields for novel view synthesis. In *Proc. ICCV*, 2023. 2

[22] Andrew S Huntington. *InGaAs avalanche photodiodes for ranging and Lidar*. Woodhead Publishing, 2020. 2

[23] Kazuto Ichimaru, Diego Thomas, Takafumi Iwaguchi, and Hiroshi Kawasaki. Neural active structure-from-motion in dark and textureless environment. *Proc. ACCV*, 2024. 2

[24] Julian Iseringhausen and Matthias B Hullin. Non-line-of-sight reconstruction using efficient transient rendering. *ACM Trans. Graph.*, 39(1):1–14, 2020. 2

[25] Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, and Delio Vicini. Dr. jit: A just-in-time compiler for differentiable rendering. *ACM Trans. Graph.*, 41(4):1–19, 2022. 1, 3

[26] Adrian Jarabo and Victor Arellano. Bidirectional rendering of vector light transport. In *Computer Graphics Forum*, pages 96–105. Wiley Online Library, 2018. 3

[27] Adrian Jarabo, Julio Marco, Adolfo Munoz, Raul Buisan, Wojciech Jarosz, and Diego Gutierrez. A framework for transient rendering. *ACM Trans. Graph.*, 33(6):1–10, 2014. 3

[28] Haian Jin, Isabella Liu, Peijia Xu, Xiaoshuai Zhang, Songfang Han, Sai Bi, Xiaowei Zhou, Zexiang Xu, and Hao Su. TensoIR: Tensorial inverse rendering. In *Proc. CVPR*, 2023. 2, 3

[29] James T Kajiya. The rendering equation. In *Proc. SIGGRAPH*, 1986. 2, 3

[30] Brian Karis. Real shading in Unreal Engine 4. *ACM SIGGRAPH Courses*, 2013. 5

[31] Ahmed Kirmani, Dheera Venkatraman, Dongeek Shin, Andrea Colaço, Franco NC Wong, Jeffrey H Shapiro, and Vivek K Goyal. First-photon imaging. *Science*, 343(6166): 58–61, 2014. 2

[32] Tzofi Klinghoffer, Xiaoyu Xiang, Siddharth Somasundaram, Yuchen Fan, Christian Richardt, Ramesh Raskar, and Rakesh Ranjan. PlatoNeRF: 3D reconstruction in Plato's cave via single-view two-bounce lidar. In *Proc. CVPR*, 2024. 2, 3

[33] Walter Koechner. Optical ranging system employing a high power injection laser diode. *IEEE Trans. Aerosp. Electron. Syst.*, AES-4(1):81–91, 1968. 2

[34] Larry Li et al. Time-of-flight camera—an introduction. *Technical white paper*, 2014. 2

[35] Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Trans. Graph.*, 37(6):1–11, 2018. 2, 3

[36] You Li and Javier Ibanez-Guzman. Lidar for autonomous driving: The principles, challenges, and trends for automotive lidar and perception systems. *IEEE Signal Process. Mag.*, 37(4):50–61, 2020. 1

[37] David B Lindell and Gordon Wetzstein. Three-dimensional imaging through scattering media based on confocal diffuse tomography. *Nat. Commun.*, 11(1):4517, 2020. 3

[38] David B Lindell, Matthew O'Toole, and Gordon Wetzstein. Towards transient imaging at interactive rates with single-photon detectors. In *Proc. ICCP*, 2018. 2

[39] David B Lindell, Gordon Wetzstein, and Matthew O'Toole. Wave-based non-line-of-sight imaging using fast fk migration. *ACM Trans. Graph.*, 38(4):1–13, 2019. 3, 8

[40] Jingwang Ling, Ruihan Yu, Feng Xu, Chun Du, and Shuang Zhao. NeRF as a non-distant environment emitter in physics-based inverse rendering. In *Proc. SIGGRAPH*, 2024. 3

[41] Xiaochun Liu, Ibón Guillén, Marco La Manna, Ji Hyun Nam, Syed Azer Reza, Toan Huu Le, Adrian Jarabo, Diego Gutierrez, and Andreas Velten. Non-line-of-sight imaging using phasor-field virtual wave optics. *Nature*, 572(7771):620–623, 2019. 3

[42] Yuan Liu, Peng Wang, Cheng Lin, Xiaoxiao Long, Jiepeng Wang, Lingjie Liu, Taku Komura, and Wenping Wang. NERO: Neural geometry and BRDF reconstruction of reflective objects from multiview images. *ACM Trans. Graph.*, 42(4):1–22, 2023. 2, 3

[43] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: learning dynamic renderable volumes from images. *ACM Trans. Graph.*, 38(4):1–14, 2019. 3

[44] Guillaume Loubet, Nicolas Holzschuch, and Wenzel Jakob. Reparameterizing discontinuous integrands for differentiable rendering. *ACM Trans. Graph.*, 38(6):1–14, 2019. 2

[45] Weihan Luo, Anagh Malik, and David B Lindell. Transient-tangelo: Few-viewpoint surface reconstruction using single-photon lidar. *Proc. WACV*, 2025. 2

[46] Alexander Mai, Dor Verbin, Falko Kuester, and Sara Fridovich-Keil. Neural microfacet fields for inverse rendering. In *Proc. ICCV*, 2023. 3, 8

[47] Anagh Malik, Parsa Mirdehghan, Sotiris Nousias, Kiriakos N Kutulakos, and David B Lindell. Transient neural radiance fields for lidar view synthesis and 3D reconstruction. In *Proc. NeurIPS*, 2023. 2, 4, 6

[48] Anagh Malik, Noah Juravsky, Ryan Po, Gordon Wetzstein, Kiriakos N. Kutulakos, and David B. Lindell. Flying with photons: Rendering novel views of propagating light. In *Proc. ECCV*, 2024. 2, 5, 6

[49] Nelson Max. Optical models for direct volume rendering. *IEEE Trans. Vis. Comput. Graph.*, 1(2):99–108, 1995. 3

[50] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM*, 65(1):99–106, 2021. 2, 3

[51] Ben Mildenhall, Peter Hedman, Ricardo Martin-Brualla, Pratul P. Srinivasan, and Jonathan T. Barron. NeRF in the dark: High dynamic range view synthesis from noisy raw images. In *Proc. CVPR*, 2022. 5

[52] Nicolas Moenne-Loccoz, Ashkan Mirzaei, Or Perel, Riccardo de Lutio, Janick Martinez Esturo, Gavriel State, Sanja Fidler, Nicholas Sharp, and Zan Gojcic. 3D Gaussian ray tracing: Fast tracing of particle scenes. *Proc. SIGGRAPH Asia*, 2024. 8

[53] Fangzhou Mu, Carter Sifferman, Sacha Jungerman, Yiquan Li, Mark Han, Michael Gleicher, Mohit Gupta, and Yin Li. Towards 3D vision with low-cost single-photon cameras. In *Proc. CVPR*, 2024. 2

[54] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph. (SIGGRAPH)*, 41(4):1–15, 2022. 5, 6

[55] Nikhil Naik, Shuang Zhao, Andreas Velten, Ramesh Raskar, and Kavita Bala. Single view reflectance capture using multiplexed scattering and time-of-flight imaging. *ACM Trans. Graph.*, 30(6):1–10, 2011. 2

[56] Nikhil Naik, Achuta Kadambi, Christoph Rhemann, Shahram Izadi, Ramesh Raskar, and Sing Bing Kang. A light transport model for mitigating multipath interference in time-of-flight sensors. In *Proc. CVPR*, pages 73–81, 2015. 3

[57] Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. Mitsuba 2: A retargetable forward and inverse renderer. *ACM Trans. Graph.*, 38(6):1–17, 2019. 2, 3, 6

[58] Mikhail Okunev, Marc Mapeke, Benjamin Attal, Christian Richardt, Matthew O'Toole, and James Tompkin. Flowed time of flight radiance fields. In *Proc. ECCV*, 2024. 2

[59] Matthew O'Toole, Felix Heide, Lei Xiao, Matthias B Hullin, Wolfgang Heidrich, and Kiriakos N Kutulakos. Temporal frequency probing for 5D transient analysis of global light transport. *ACM Trans. Graph.*, 33(4):1–11, 2014. 2, 3

[60] Matthew O'Toole, Felix Heide, David B Lindell, Kai Zang, Steven Diamond, and Gordon Wetzstein. Reconstructing transient images from single-photon sensors. In *Proc. CVPR*, 2017. 2

[61] Matthew O'Toole, David B Lindell, and Gordon Wetzstein. Confocal non-line-of-sight imaging based on the light-cone transform. *Nature*, 555(7696):338–341, 2018. 2

[62] Adithya Pediredla, Ashok Veeraraghavan, and Ioannis Gkioulekas. Ellipsoidal path connections for time-gated rendering. *ACM Trans. Graph.*, 38(4):1–12, 2019. 3

[63] Adithya Pediredla, Yasin Karimi Chalmiani, Matteo Giuseppe Scopelliti, Maysamreza Chamanzar, Srinivasa Narasimhan, and Ioannis Gkioulekas. Path tracing estimators for refractive radiative transfer. *ACM Trans. Graph.*, 39(6):1–15, 2020. 3

[64] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. MIT Press, 2023. 1, 3

[65] Ravi Ramamoorthi and Pat Hanrahan. A signal-processing framework for inverse rendering. In *Proc. SIGGRAPH*, 2001. 3

[66] Andrea Ramazzina, Stefanie Walz, Pragyan Dahal, Mario Bijelic, and Felix Heide. Gated fields: Learning scene reconstruction from gated videos. In *Proc. CVPR*, 2024. 2

[67] Raskar Ramesh and James Davis. 5D time-light transport matrix: What can we reason about scene properties? Technical report, MIT, 2008. 3

[68] Joshua Rapp, Charles Saunders, Julián Tachella, John Murray-Bruce, Yoann Altmann, Jean-Yves Tourneret, Stephen McLaughlin, Robin MA Dawson, Franco NC Wong, and Vivek K Goyal. Seeing around corners with edge-resolved transient imaging. *Nat. Commun.*, 11(1):5929, 2020. 2, 3

[69] Joshua Rapp, Julian Tachella, Yoann Altmann, Stephen McLaughlin, and Vivek K Goyal. Advances in single-photon lidar for autonomous vehicles: Working principles, challenges, and recent advances. *IEEE Signal Process. Mag.*, 37(4):62–71, 2020. 2

[70] Konstantinos Rematas, Andrew Liu, Pratul P Srinivasan, Jonathan T Barron, Andrea Tagliasacchi, Thomas Funkhouser, and Vittorio Ferrari. Urban radiance fields. In *Proc. CVPR*, 2022. 2

[71] Diego Royo, Jorge García, Adolfo Muñoz, and Adrian Jarabo. Non-line-of-sight transient rendering. *Comput. Graph.*, 107:84–92, 2022. 6

[72] Diego Royo, Talha Sultan, Adolfo Muñoz, Khadijeh Masumnia-Bisheh, Eric Brandt, Diego Gutierrez, Andreas Velten, and Julio Marco. Virtual mirrors: Non-line-of-sight imaging beyond the third bounce. *ACM Trans. Graph.*, 42 (4):1–15, 2023. 3

[73] Brent Schwarz. Mapping the world in 3D. *Nat. Photonics*, 4 (7):429–430, 2010. 1

[74] Sheila Seidel, Hoover Rueda-Chacón, Iris Cusini, Federica Villa, Franco Zappa, Christopher Yu, and Vivek K Goyal. Non-line-of-sight snapshots and background mapping with an active corner camera. *Nat. Commun.*, 14(1):3677, 2023. 3

[75] Aarrushi Shandilya, Benjamin Attal, Christian Richardt, James Tompkin, and Matthew O'Toole. Neural fields for structured lighting. In *Proc. ICCV*, 2023. 2

[76] Dongeek Shin, Ahmed Kirmani, Vivek K Goyal, and Jeffrey H Shapiro. Photon-efficient computational 3-d and reflectivity imaging with single-photon detectors. *IEEE Trans. Comput. Imaging*, 1(2):112–125, 2015. 2

[77] Pratul P Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T Barron. NeRV: Neural reflectance and visibility fields for relighting and view synthesis. In *Proc. CVPR*, 2021. 2, 3

[78] Andrea Tagliasacchi and Ben Mildenhall. Volume rendering digest (for NeRF). *arXiv preprint*, 2022. 3

[79] Chia-Yin Tsai, Ashok Veeraraghavan, and Aswin C Sankaranarayanan. Shape and reflectance from two-bounce light transients. In *Proc. ICCP*, pages 1–10. IEEE, 2016. 3

[80] Chia-Yin Tsai, Aswin C Sankaranarayanan, and Ioannis Gkioulekas. Beyond volumetric albedo–a surface optimization framework for non-line-of-sight imaging. In *Proc. CVPR*, 2019. 2

[81] Eric Veach. *Robust Monte Carlo methods for light transport simulation*. Stanford University, 1998. 3

[82] Andreas Velten, Di Wu, Adrian Jarabo, Belen Masia, Christopher Barsi, Chinmaya Joshi, Everett Lawson, Moungi Bawendi, Diego Gutierrez, and Ramesh Raskar. Femto-photography: capturing and visualizing the propagation of light. *ACM Trans. Graph.*, 32(4):1–8, 2013. 2

[83] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *Proc. CVPR*, 2022. 5

[84] Dor Verbin, Ben Mildenhall, Peter Hedman, Jonathan T Barron, Todd Zickler, and Pratul P Srinivasan. Eclipse: Disambiguating illumination and materials using unintended shadows. In *Proc. CVPR*, 2024. 3

[85] Gregory J Ward, Francis M Rubinstein, and Robert D Clear. A ray tracing solution for diffuse interreflection. In *Proc. SIGGRAPH*, 1988. 2, 3

[86] Di Wu, Andreas Velten, Matthew O'Toole, Belen Masia, Amit Agrawal, Qionghai Dai, and Ramesh Raskar. Decomposing global light transport using time of flight imaging. *Int. J. Comput. Vis.*, 107:123–138, 2014. 2

[87] Haoqian Wu, Zhipeng Hu, Lincheng Li, Yongqiang Zhang, Changjie Fan, and Xin Yu. Nefii: Inverse rendering for reflectance decomposition with near-field indirect illumination. In *Proc. CVPR*, pages 4295–4304, 2023. 3

[88] Jiaye Wu, Saeed Hadadan, Geng Lin, Matthias Zwicker, David Jacobs, and Roni Sengupta. GaNI: Global and near field illumination aware neural inverse rendering. *arXiv*, 2024. 3

[89] Lifan Wu, Guangyan Cai, Ravi Ramamoorthi, and Shuang Zhao. Differentiable time-gated rendering. *ACM Trans. Graph.*, 40(6):1–16, 2021. 2, 3

[90] Shumian Xin, Sotiris Nousias, Kiriakos N Kutulakos, Aswin C Sankaranarayanan, Srinivasa G Narasimhan, and Ioannis Gkioulekas. A theory of Fermat paths for non-line-of-sight shape reconstruction. In *Proc. CVPR*, 2019. 2, 3

[91] Yao Yao, Jingyang Zhang, Jingbo Liu, Yihang Qu, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. NEILF: Neural incident light field for physically-based material estimation. In *Proc. ECCV*, 2022. 3

[92] Shinyoung Yi, Donggun Kim, Kiseok Choi, Adrian Jarabo, Diego Gutierrez, and Min H. Kim. Differentiable transient rendering. *ACM Trans. Graph.*, 40(6), 2021. 2, 3

[93] Yizhou Yu, Paul Debevec, Jitendra Malik, and Tim Hawkins. Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *Proc. SIGGRAPH*, 1999. 3

[94] Junge Zhang, Feihu Zhang, Shaochen Kuang, and Li Zhang. NeRF-lidar: Generating realistic lidar point clouds with neural radiance fields. In *Proc. AAAI*, 2024. 2

[95] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proc. CVPR*, 2018. 6

[96] Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Trans. Graph.*, 40(6):1–18, 2021. 2, 3

[97] Yuanqing Zhang, Jiaming Sun, Xingyi He, Huan Fu, Rongfei Jia, and Xiaowei Zhou. Modeling indirect illumination for inverse rendering. In *Proc. CVPR*, 2022. 2, 3