Learned Feature Embeddings for Non-Line-of-Sight Imaging and Recognition

WENZHENG CHEN*, University of Toronto, Vector Institute
FANGYIN WEI*, Princeton University
KIRIAKOS N. KUTULAKOS, University of Toronto
SZYMON RUSINKIEWICZ, Princeton University
FELIX HEIDE, Princeton University

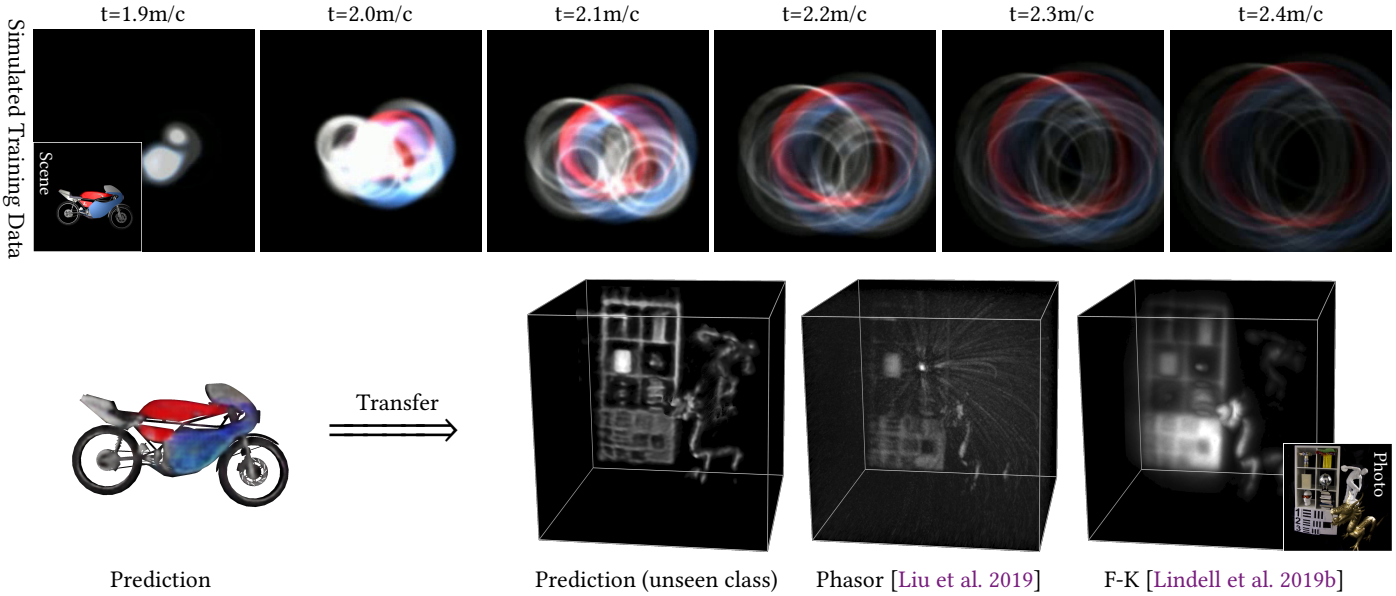

Fig. 1. We devise a method for learning feature embeddings tailored to non-line-of-sight reconstruction and object recognition. The proposed learned inverse method is supervised purely using synthetic transient image data (top row). Trained on a synthetic scenes containing only a single object type ("motorbike") from ShapeNet [2015], the trained model generalizes from synthetic data (bottom left) to unseen classes of measured experimental data (bottom right). Note that the proposed model recovers geometry not present in existing methods, such as the reflective styrofoam parts of the mannequin head.

Objects obscured by occluders are considered lost in the images acquired by conventional camera systems, prohibiting both visualization and understanding of such hidden objects. Non-line-of-sight methods (NLOS) aim at recovering information about hidden scenes, which could help make medical imaging less invasive, improve the safety of autonomous vehicles, and potentially enable capturing unprecedented high-definition RGB-D data sets that include geometry beyond the directly visible parts. Recent NLOS methods have demonstrated scene recovery from time-resolved pulse-illuminated measurements encoding occluded objects as faint indirect reflections. Unfortunately, these systems are fundamentally limited by the quartic intensity fall-off for diffuse scenes. With laser illumination limited by eye-safety limits, recovery algorithms must tackle this challenge by incorporating scene priors. However, existing NLOS reconstruction algorithms do not facilitate learning scene priors. Even if they did, datasets that allow for such supervision do not exist, and successful encoder-decoder networks and generative adversarial networks fail for real-world NLOS data. In this work, we close this gap by learning hidden scene feature representations tailored to both reconstruction and recognition tasks such as classification or object detection, while still relying on physical models at the feature level. We *overcome the lack of real training data* with a generalizable architecture that can be trained in simulation. We learn the differentiable scene representation jointly with the reconstruction task using a differentiable transient renderer in the objective, and demonstrate that it *generalizes to unseen classes and unseen real-world*

* indicates equal contribution.

Authors' addresses: Wenzheng Chen*, wenzheng@cs.toronto.edu, University of Toronto, Vector Institute; Fangyin Wei*, fwei@cs.princeton.edu, Princeton University; Kiriakos N. Kutulakos, kyros@cs.toronto.edu, University of Toronto; Szymon Rusinkiewicz, smr@princeton.edu, Princeton University; Felix Heide, fheide@cs.princeton.edu, Princeton University.

0730-0301/2020/12-ART230 $15.00
https://doi.org/10.1145/3414685.3417825

scenes, unlike existing encoder-decoder architectures and generative adversarial networks. The proposed method allows for *end-to-end training for different NLOS tasks*, such as image reconstruction, classification, and object detection, while being memory-efficient and running at real-time rates. We demonstrate *hidden view synthesis, RGB-D reconstruction, classification, and object detection* in the hidden scene in an end-to-end fashion.

CCS Concepts: • **Computing methodologies → Computational photography**.

ACM Reference Format:
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, Article 230 (December 2020), 18 pages. https://doi.org/10.1145/3414685.3417825

1 INTRODUCTION

Conventional sensor systems capture objects in their direct line of sight, limiting downstream display and scene understanding methods to the visible parts of the scene. Non-line-of-sight (NLOS) methods, in contrast, aim at recovering information about occluded objects by analyzing their indirect reflections or shadows on surfaces that *are* in the line of sight of the detector. Scene understanding of occluded objects may enable unprecedented applications across domains, including remote sensing, medical imaging, and industrial machine vision, and it could help to make autonomous driving safer by detecting *all* objects in the vicinity of a vehicle and not only the directly visible ones.

NLOS imaging and scene understanding is challenging because of two fundamental limitations of the measurement formation process. First, there is an inherent low-pass angular filter induced by imaging diffuse indirect reflections of diffuse scene surfaces. Second, the intensity of these indirect reflections decreases quartically with distance to the visible relay surface. To tackle the lack of angular resolution, a number of NLOS approaches have been described over the last few years that temporally probe the light-transport in the scene, thereby unmixing light path contributions by their optical path length [Abramson 1978; Kirmani et al. 2009; Naik et al. 2011; Pandharkar et al. 2011]. This provides a means for increasing angular resolution, at the expense of needing high effective temporal resolution (on the order of picoseconds). To acquire temporally resolved images of light transport, the most successful methods directly sample the temporal impulse response of the scene by recording the temporal echoes of laser pulses [Velten et al. 2012; Pandharkar et al. 2011; Gupta et al. 2012; Buttafava et al. 2015; Tsai et al. 2017; Arellano et al. 2017; O'Toole et al. 2018a]. However, while successfully recovering angular resolution, these methods unfortunately do not solve the second challenge of the *low signal* present in the indirect illumination. While some systems rely on engineered retro-reflective materials [O'Toole et al. 2018a; Chen et al. 2019; Lindell et al. 2019b], which are rare in realistic scenes, general-purpose methods often resort to increasing illumination power, exceeding the eye-safety limits for a Class 1 laser (e.g. Velodyne HDL-64E) by a factor of 1000 [Lindell et al. 2019b]. As a result, the *underlying inverse problem* is fundamentally limited by the low-signal component of the temporally resolved measurements.

NLOS reconstruction methods must cope with this ill-posedness and noise sensitivity by incorporating accurate forward models and image priors. While forward models have been proposed that can successfully handle different surface reflection types [Liu et al. 2019; Lindell et al. 2019b; O'Toole et al. 2018b] and occlusions in the hidden volume [Heide et al. 2019], existing methods incorporate only limited scene priors. Specifically, previous inverse filtering methods either support no scene priors [Liu et al. 2019; Lindell et al. 2019b], are limited to non-negativity or sparsity priors with iterative optimization *at the cost of more than 100 min of recovery time* [O'Toole et al. 2018b] (LCT+TV variant), or they *explicitly enforce scene priors* as surface representations [Pediredla et al. 2017; Tsai et al. 2019]. As such, *existing methods do not allow for learning rich scene priors from scene datasets*, and existing vanilla image-to-image mapping networks fail for non-local NLOS reconstruction problems as we show in this work. Moreover, even if existing methods facilitated learning priors, *large real or synthetic datasets that allow for supervised learning do not exist*. The lack of datasets and trainable NLOS reconstruction methods also makes it challenging to learn recognition tasks such as classification or detection of objects in the hidden scene components in an end-to-end fashion, limiting existing methods [Caramazza et al. 2018a] to captured data of a single class with baked-in setup geometry.

In this work, we close the gap between learned methods, which allow for rich priors, and physically motivated reconstruction methods. We propose to learn hidden scene feature representations tailored to both NLOS reconstruction and recognition directly from the raw transient images. Instead of aggregating transient intensities, and explicitly enforcing hidden albedo constraints, we base our system on learned deep *feature maps* that are extracted from the input transients, propagated to the hidden scene volume as a learnable low-resolution 3D feature map, and used directly by downstream rendering and recognition tasks. This strategy allows us to overcome many of the weaknesses of traditional methods. First, the mapping is trained to be insensitive to surface reflectance and occlusion. Second, the propagation from 2D to 3D can proceed via a learned module, or can exploit existing physical models (applied to feature maps instead of intensity) without inheriting their limitations. The 3D feature maps can be used to enforce multi-view and depth consistency, while their projections back into 2D result in high-quality images via a rendering network. Finally, the whole process can run at real-time rates, as intermediate 3D feature representation is more compressed than the input data.

We supervise this differentiable scene representation using simulated transient renderings. To generate a large training data corpus for training, we propose a novel highly-efficient transient rendering method relying on rasterization hardware.

Although trained in simulation, the proposed reconstruction method *generalizes well to real data, in contrast to existing encoder-decoder or generative adversarial networks*. It allows for high-resolution reconstructions from time-resolved transient measurements at real-time rates. We validate that the proposed method naturally allows us to learn diverse downstream NLOS tasks such as hidden view synthesis, RGB-D reconstruction, classification, and object detection in the hidden volume in an end-to-end fashion.

In particular, we make the following contributions:

- We introduce a method for learning feature embeddings tailored to non-line-of-sight reconstruction, as well as specific imaging and downstream classification and object detection tasks. We extract these sparse hidden features from simulated transient images, using *learned feature extraction blocks* and *feature propagation units that can leverage physical models*.
- The proposed learned feature representation *natively incorporates* 3D scene structure, such as occlusion and multi-view consistency. It learns to encode geometry and surface properties in a scene- and task-dependent manner, with priors learned in an end-to-end fashion.
- We *train and analyze* the proposed method in simulation and validate that the method outperforms state-of-the-art reconstruction methods by more than 5 dB in PSNR for RGB-D image recovery, evaluated on more than 600 scenes, while being memory-efficient and allowing for real-time reconstruction rates.
- We *assess* the proposed method on a dataset of experimental data, validating that the approach generalizes and outperforms recent volumetric reconstruction methods across a variety of scenes. All datasets, models, and code for rendering and training of the proposed models will be published.

1.1 Overview of Limitations

The proposed deep reconstruction method requires a large training corpus to represent objects with arbitrary shapes, orientations, locations and reflectance; at the same time, unfortunately, only a dozen real-world transient measurements are available. Although we tackle this issue by training in simulation, without sacrificing generalization, we rely on representative 3D scene datasets, such as ShapeNet [Chang et al. 2015], and inherit their limitations in diversity and realism, e.g., including various surface reflectance types. Also as a result of existing shape dataset limitations, semantic decomposition and analysis of hidden 3D scenes that are complex is out of the scope of this work, but we anticipate that this is a promising avenue for future work.

2 RELATED WORK

We review prior art most related to our contributions, below.

Transient Imaging. Kirmani et al. [2009] first proposed the concept of recovering "hidden" objects outside a camera's direct line of sight, using temporally resolved light transport measurements in which short pulses of light are captured "in flight" before the global transport reaches steady state. These transient measurements are the temporal impulse responses of light transport in the scene. Abramson [1978] first demonstrated a holographic capture system for transient imaging, and Velten et al. [2013] showed the first experimental non-line-of-sight imaging results using a femto-second laser and streak camera system. The first successful reconstruction method is filtered backprojection which propagates and aggregates time-resolved intensity measurements back into in the occluded volume, followed by a Laplacian filter [Velten et al. 2012], an approach extended and made efficient in recent years [Laurenzis and Velten 2014; Arellano et al. 2017; Jarabo et al. 2017]. Since these seminal works, a growing body of work has been exploring transient imaging with a focus on enabling improved non-line-of-sight imaging [Pandharkar et al. 2011; Naik et al. 2011; Wu et al. 2012; Gupta et al. 2012; Heide et al. 2014, 2013; Buttafava et al. 2015].

Impulse Non-Line-of-Sight Sensing and Imaging. A growing body of work explores optical NLOS imaging techniques [Pandharkar et al. 2011; Velten et al. 2012; Gupta et al. 2012; Kadambi et al. 2016; O'Toole et al. 2018a; Tsai et al. 2017; Arellano et al. 2017; Pediredla et al. 2017; O'Toole et al. 2018b; Xu et al. 2018; Liu et al. 2019]. Following Kirmani et al. [2009], who first proposed the concept of recovering occluded objects from time-resolved light transport, these methods directly sample the temporal impulse response of a scene by sending out pulses of light and capturing their response using detectors with high temporal precision of < 10 ps, during which a pulse travels a distance of 3 mm. While early work relies on costly and complicated streak camera setups [Velten et al. 2012, 2013], a recent line of work uses single photon avalanche diodes (SPADs) [Buttafava et al. 2015; O'Toole et al. 2018b; Liu et al. 2019, 2020]. Although SPAD sensors offer comparable time resolution of under 10 ps [Nolet et al. 2018], existing detectors with large active area are challenging to realize as arrays [Parmesan et al. 2014], requiring point-by-point scanning [O'Toole et al. 2018b; Liu et al. 2019, 2020] similar to scanning LIDAR systems. These recent scanning-based systems achieve the highest resolution NLOS reconstructions and transient image resolutions. Parallel to our work, [Chopite et al. 2020] propose to directly train an encoder-decoder network to learn NLOS reconstruction from synthetic transients. Their results indicate that existing encoder-decoder networks generalize poorly to real data. In this work, we depart from such architectures and learn feature embeddings that allow us to close the domain gap.

Modulated and Coherent Non-Line-of-Sight-Imaging. Correlation-based time-of-flight sensors have been proposed as an alternative to impulse-based acquisition [Heide et al. 2013; Kadambi et al. 2013; Heide et al. 2014; Kadambi et al. 2016], encoding travel-time indirectly in phase measurements. A recent line of work [Marco et al. 2017; Su et al. 2018; Guo et al. 2018] relies on synthetic data for training depth estimation networks. Although these works aim to recover the direct reflection, while this work focuses on indirect bounces, they demonstrate the potential of learning inverse models for complex light transport in the scene.

Katz et al. [2012, 2014] demonstrate that correlations in the carrier wave itself can be used to realize fast single shot NLOS imaging that is, however, limited to scenes at microsopic scales [Katz et al. 2014]. Recently, [Metzler et al. 2020] demonstrate a correlography approach to NLOS imaging. While this approach achieves high spatial resolution of 300 µm it is also limited to a single sparse object and small standoff distances of 1 m. Unfortunately, recent acoustic methods [Lindell et al. 2019a] are currently limited to meter-sized lab scenes and minutes of acquisition time.

Non-Line-of-Sight Tracking and Classification. Several recent works use conventional intensity images for NLOS tracking and localization [Klein et al. 2016; Caramazza et al. 2018a; Chan et al. 2017;

Bouman et al. 2017; Chen et al. 2019]. The ill-posedness of the underlying inverse problem limits these methods to localization with highly reflective targets [Bouman et al. 2017; Chen et al. 2019], sparse dark background, scenes with additional occluders present [Bouman et al. 2017; Saunders et al. 2019], or a single object class [Caramazza et al. 2018a]. Using radar sensors, recently, [Scheiner et al. 2020] achieved NLOS detection and tracking of multiple object classes at large stand-off distances of more than 20 m in automotive outdoor scenarios.

Learning Multiview Image Synthesis. A growing body of work explores learning multiview image synthesis from sparsely sampled images of a given 3D scene. Such existing methods learn scene representations [Tatarchenko et al. 2015; Zhou et al. 2016; Sitzmann et al. 2019a; Olszewski et al. 2019; Lombardi et al. 2019; Sitzmann et al. 2019b; Mildenhall et al. 2020] from input data and generate new views penalized by re-rendering losses. While several earlier works focus on representing the scene in the latent space [Tatarchenko et al. 2015; Zhou et al. 2016], recently, researchers have become interested in explicitly encoding the scene as a 3D volumetric feature block [Sitzmann et al. 2019a; Olszewski et al. 2019]. Moreover, ray tracing and ray marching technique can also be added [Lombardi et al. 2019; Mildenhall et al. 2020] to learn how to deal with occlusion, which result in much more high resolution reconstruction. All of these techniques have in common that they naturally exploit multiview geometry and scene constraints. In this work, we also reason on volumetric feature spaces. However, instead of extracting such features directly from multiview scene photographs, we extract them from transient images. We spatially transform the transient features to the hidden scene volume, which we only then map to rendered images of the unknown scene. Moreover, instead of overfitting models to a single (or parameterized) scene for view interpolation [Mildenhall et al. 2020] (note that overfitting to the scene can be intended for multiview reconstruction methods), we introduce an inverse method that does not overfit and recovers occluded information for transient input data from unseen scenes – generalizing to real data although trained in simulation only.

3 OBSERVATION MODEL

NLOS methods recover information about occluded objects outside the direct line of sight from time-resolved global light transport measurements of third-order reflections. Specifically, a small patch of a diffuse relay wall in the direct line of sight of the detector is illuminated with a short laser pulse. The light scatters off this patch to the hidden object, which reflects some of it back to the visible wall, where it gets recorded after a third diffuse reflection to the detector, see Fig. 2. Without loss of generality, we assume a setup in which a single laser spot at the center of the relay wall, at coordinates $(0, 0)$, is illuminated and the indirect reflections are sampled at positions (x', y') on the visible relay wall. The derived image formation of this NLOS setup generalizes to both non-coaxial setups with multiple laser points as well as co-axial setups.

The time-resolved incident photon flux, including the indirect and direct global illumination (direct only for the center of relay wall), is recorded for every sample position as a transient observation τ, resulting in a 3D spatio-temporal measurement cube, i.e. a video

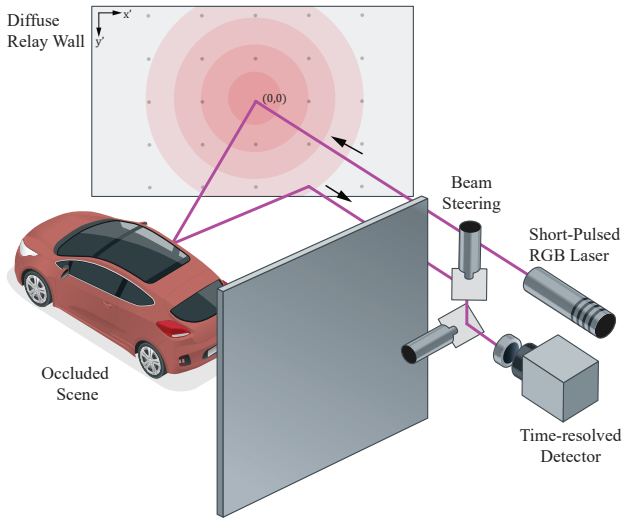

Fig. 2. **Temporally Resolved NLOS Acquisition.** A laser pulse is projected at the center of a diffuse wall, and the resulting time-resolved direct and indirect light transport is sampled at points (x', y').

of the pulse traveling through the scene at picosecond resolution. Fig. 1 shows an example of such a transient measurement cube.

Assuming the visible wall geometry to be known, e.g. from first-bounce direct time-of-flight measurements or (active) stereo methods, we can ignore the direct bounce, either by discarding samples around the center light position or by employing gating hardware [Liu et al. 2019; Lindell et al. 2019b], resulting in the following observation model

$$\tau\left(x', y', t\right) = \iiint \rho\left(x, y, z\right)\; f\left(\omega_{(0,0,0)\to(x,y,z)},\; \omega_{(x,y,z)\to(x',y',0)}\right) \gamma\left(0, 0, x, y, z\right)\; \gamma\left(x', y', x, y, z\right) \delta\left(\sqrt{x^2+y^2+z^2}+\sqrt{(x'-x)^2+(y'-y)^2+z^2}-tc\right)\, dx\, dy\, dz, \tag{1}$$

with the time dirac delta function $\delta\left(\cdot\right)$ converting time t to the travel distance $r = tc$, with c as the speed of light. Here, the geometry term $\gamma\left(\cdot\right)$ models mutual visibility, foreshorting due to surface orientation n of the hidden surface, and intensity falloff as

$$\gamma\left(x', y', x, y, z\right) = \frac{\left(\omega_{(x',y',0)\to(x,y,z)} \cdot n\left(x, y, z\right)\right) \cdot v_{(x',y',0)\to(x,y,z)}}{\sqrt{(x'-x)^2 + (y'-y)^2 + z^2}}, \tag{2}$$

where the orientation $\omega_{\mathbf{p}_i\to\mathbf{p}_o} = (\mathbf{p}_o - \mathbf{p}_i)/\|\mathbf{p}_o - \mathbf{p}_i\|$ is the normalized direction between the two input arguments. Adopting the same notation, the visibility term $v_{\mathbf{p}_i\to\mathbf{p}_o} \in [0, 1]$ models the visibility of a path between the two input patches at $\mathbf{p}_i$ and $\mathbf{p}_o$ in the hidden scene. For partial occlusions, we adopt the continuous notation from [Heide et al. 2019]. We model the bidirectional reflectance distribution function (BRDF) f from forward model Eq. (1) as

$$f\left(\omega_i, \omega_o\right) \;=\; f_d\left(\omega_i, \omega_o\right) \;+\; f_s\left(\omega_i, \omega_o\right) \;+\; f_r\left(\omega_i, \omega_o\right)\;. \tag{3}$$

Here, the diffuse component f_d models diffuse scattering, which are almost directionally constant. The specular component f_s represents specular highlights, i.e., mirror-like reflections with a specular lobe. Although these specular components can be used for large wall geometries [Chen et al. 2019], diffuse reflections typically dominate the transient image for small relay wall geometries and long stand-off distances. The retroreflective BRDF component f_r represents a sharp retroreflective lobe around $\omega_i = \omega_o = \omega$, which is present in a few engineered surface types [O'Toole et al. 2018b; Lindell et al. 2019b; Chen et al. 2019]. Note that we only measure this retroreflective component for the light source position $x' = 0, y' = 0$. The confocal scanning setup [O'Toole et al. 2018b; Lindell et al. 2019b] is a variation of the proposed model for this point, where the light source is now moved along with the sampling position x', y' to be able to sample this retroreflective BRDF component at every sampling position. We model the unknown, hidden scene albedo as a directionally constant but spatially-varying function $\rho(x, y, z)$.

Note that the only assumption that the forward model from Eq. (1) makes is that the indirectly reflected light from the occluded scene scatters only once in the occluded scene.

3.1 Detector Model

Although our method is not limited to a specific transient detector type, the results in this paper assume that transient images are captured using a single photon avalanche diode (SPAD). SPAD detectors offer high temporal resolution of under 10 ps [Nolet et al. 2018], and offer the promise of potential implementation as high-resolution sensor arrays in CMOS technology in the future [Burri 2016]. As such, a growing body of work relies on SPAD detectors for NLOS imaging [Buttafava et al. 2015; O'Toole et al. 2018b; Liu et al. 2019]. Unfortunately, SPADs are not without disadvantages: they suffer from a small active area, and their operating principle prohibits recording subsequent photons after a given photo-electron has generated an avalanche. While this behavior can lead to pile-up histogram skew [Coates 1972] for the direct peak (and hence does affect purely co-axial setups), the indirect reflections are in a low-flux regime, where the probability of observing multiple photons from a single pulse is small and, hence, pile-up can be ignored [Kirmani et al. 2014].

We follow the forward model approach proposed by Hernandez et al. [2017]. While the authors propose an extensive detector model that also comes at high computational cost, we adopt the core noise components from their method and model the raw transient measurements accumulated with N pulses as

$$\tilde{\tau}(x', y', \tilde{t}) \sim \text{Poisson}\left(N\mu\left(\tau \otimes g + s\right)(x', y', t^\dagger) + N\,d\right) \quad \text{with}$$
$$t^\dagger \sim \text{Jitter}\left(\tilde{t}, \sigma_{\text{jitter}}\right), \tag{4}$$

where $\mu > 0$ is the quantum efficiency, d is the dark count rate per time bin $\tilde{t}$, and s is the ambient light per time bin. The continuous transient image τ is convolved with a function g the laser impulse response. We model detector jitter as a sampling process where the temporal acquisition bin $t^\dagger$ is sampled from a Jitter distribution Jitter, which we model as a Gaussian distribution with mean $\tilde{t}$ and standard deviation σ_{jitter}. Here we simplify the time jitter model in [Hernandez et al. 2017] and ignore the exponential tail for efficiency in training. We found that explicitly modeling detector jitter instead of absorbing it in the temporal PSF, e.g., in contrast to [Heide et al. 2018], is critical for synthetic data that generalizes. Combining all the detected photon arrival events into a single histogram results in a discrete Poisson-distributed random variable for each temporal bin $\tilde{t}$ of the resulting transient measurement $\tilde{\tau}$.

We note that [Hernandez et al. 2017] also model crosstalk and afterpulsing which we ignore. As the samples in confocal measurement setups are captured individually, we do not observe cross-talk in our experimental measurements. As the detectors used for validation in this work have an afterpulsing probability between 0.1% and 3%, we ignore afterpulsing to make our forward model more efficient for training with large datests. We show simulated transient frames rendered with the proposed noise model in Fig. 3 and compare them to an experimental measurement.

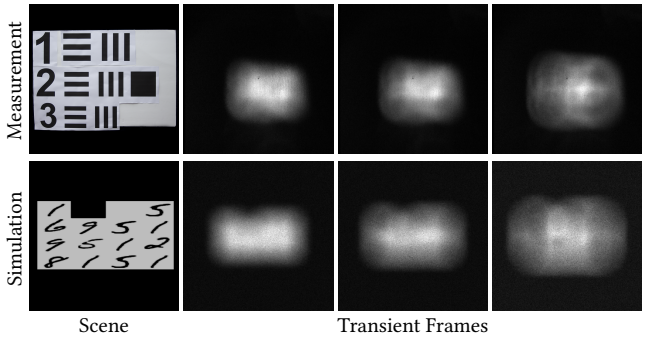

Fig. 3. **Synthesis of Realistic Training Data.** Top: Measurement of the resolution chart scene from [Lindell et al. 2019b] and three transient frames. Bottom: synthetic 3D model with digit texture placed in a virtual hidden volume for a comparable setup (exact position of the chart was not provided) and simulated transient frames. The proposed rasterization-based renderer and detector noise model synthesize realistic simulated training data.

3.2 Transient Rasterization

In this work, we propose a deep neural network to learn occluded 3D scene recovery. Training this network requires a large corpus of transient training data, which does not exist. Instead of capturing such a dataset with existing lab setups, which would mandate tens of minutes of capture time per scene [Liu et al. 2019; O'Toole et al. 2018a], we train purely on simulated transient image data. Although rendering approaches for steady-state indirect measurements have been proposed [Chen et al. 2019; Tancik et al. 2018], these methods unfortunately do not extend to time-resolved rendering, and hence cannot be applied in our setting. Recent ray tracing methods such as those by Jarabo et al. [2014], Jarabo and Arellano [2018], and Pediredla et al. [2019] are also impractical, as they would require rendering times of multiple weeks for the training data corpus used in this work.

To tackle this challenge, we propose a highly efficient transient renderer using rasterization hardware acceleration, extending [Chen et al. 2019] to render transient data for arbitrary setup geometries. As shown in Fig. 4, each camera pixel (x', y') on the wall receives

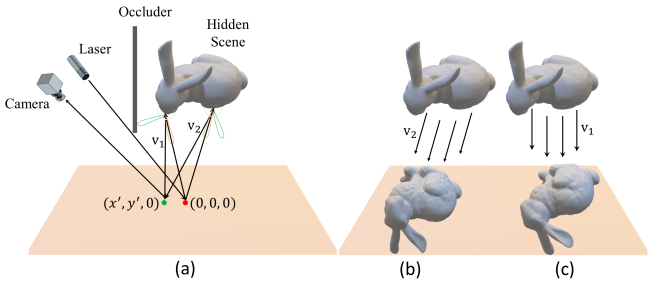

Fig. 4. **Fast Transient Rendering using Rasterization. (a)** We decompose the transient image formation process into the integral over incoming indirect illumination. The camera pixel (x', y') receives lights from directions v_1 and v_2. Indirect reflections may be composed of diffuse components (gray light lobe), specular component (green light lobe) and retro-reflective component (orange light lobe). **(b) & (c)** Different light directions result in different projections on the relay wall. We render intensity map and distance map (from light source to object then back to the wall), and accumulate them in a histogram to render a full transient image volume.

photons from directions, such as $\mathbf{v}_1$ and $\mathbf{v}_2$, over the incident hemisphere centered at (x', y'), resulting from indirect reflections of hidden objects illuminated by the light source at $(0, 0)$. This means that we can rewrite Eq. (1) from an integral over the scene into an integral over directions $\mathbf{v}$ on the incident hemisphere Ω

$$\begin{aligned}\tau\left(x', y', t\right) = \int_{\mathbf{v}\in\Omega} & \rho\left(x'_{\mathbf{v}}, y'_{\mathbf{v}}, z'_{\mathbf{v}}\right)\ f\left(\omega_{(0,0,0)\to(x'_{\mathbf{v}}, y'_{\mathbf{v}}, z'_{\mathbf{v}})},\ -\mathbf{v}\right) \\ & \gamma\left(0, 0, x'_{\mathbf{v}}, y'_{\mathbf{v}}, z'_{\mathbf{v}}\right)\ \gamma\left(x', y', x'_{\mathbf{v}}, y'_{\mathbf{v}}, z'_{\mathbf{v}}\right) \\ & \delta\left(\sqrt{x'^2_{\mathbf{v}} + y'^2_{\mathbf{v}} + z'^2_{\mathbf{v}}}\ +\ s(x', y', \mathbf{v})\ - tc\right)\ \ d\mathbf{v},\end{aligned} \tag{5}$$

where the scalar function $s(x', y', \mathbf{v})$ expresses the distance to first intersection along the ray starting at $(x', y', 0)$ in the direction $\mathbf{v}$, and $(x'_{\mathbf{v}}, y'_{\mathbf{v}}, z'_{\mathbf{v}}) = (x', y', 0) + s(x', y', \mathbf{v}) \cdot \mathbf{v}$ is that intersection point.

We evaluate this integral by sampling directions $\mathbf{v}$, each of which corresponds to a single rasterization pass. As shown in Fig. 4, although a standard orthogonal view is used for direction v_1 in (c), v_2 in (b) requires a sheared parallel projection, which nonetheless is accommodated by rasterization hardware. We use OpenGL rasterization and use both vertex shader and fragment (pixel) shaders. Using vertex shaders, we not only obtain an RGB intensity map for a point light at $(0, 0, 0)$, but are also able to use the alpha channel to store the path length from the light source to the vertex position and back to the pixel position. This approach also generalizes to confocal captures [O'Toole et al. 2018a], where each pixel $\tau\left(x', y', t\right)$ is illuminated by a source shifted to position $(x', y', 0)$. We implement this setup geometry directly in the vertex shader as directional illumination from the wall in direction $\mathbf{v}$. As each wall patch maps to a sensor measurement location, the proposed rasterization-based method naturally scales to different uniform sampling resolutions. Each relay wall (sensor) pixel only receives light from its "own" source, and, as there is no cross talk, our renderer can be used for confocal or non-confocal setups, see Fig. 5.

The final third-bounce transient image is rendered by accumulating 10000 cosine-weighted hemisphere samples, i.e., Lambertian importance sampling, with each intensity pixel accumulated in its arrival time bin. We implement this process in GPU memory using CUDA programming, allowing us to render transient images with spatial resolution of 256×256 and 600 time bins in 117 ms for the mesh shown in Fig. 5 with 52081 vertices and 200018 faces. In Table. 1, we also compare the rendering time of the proposed renderer to the multi-path ray tracing renderers from [Jarabo et al. 2014; Jarabo and Arellano 2018], [Pediredla et al. 2019], and the three-bounce renderer from [Iseringhausen and Hullin 2020]. The proposed rasterization-based method *outperforms existing methods by an order of magnitude*, though, similar to [Tsai et al. 2019; Iseringhausen and Hullin 2020] it does also fail to account for higher-order light bounces and it is not unbiased, please refer to the Supplemental Material for details. Relying on our renderer to generate a large training dataset, we validate the resulting model on real transient measurements that include higher-order bounces. Extending the proposed method to additional bounces is out of scope for this work but may be facilitated by relying on Hemi-cube [1985] rendering in the future.

Scene Complexity	128×128 quads	16×16 quads	4×4 quads
Jarabo et al.	21.64s	19.91s	18.94s
Pediredla et al.	26.2s	25.8s	25.5s
Iseringhausen and Hullin	1032.2ms	19.02ms	5.44ms
Proposed	26.90ms	24.06ms	23.19ms

Table 1. **Transient Rendering Time Comparisons.** The multi-path ray tracing transient renderers from [Jarabo et al. 2014; Jarabo and Arellano 2018] and [Pediredla et al. 2019] require around 20 sec to render a transient image. The three-bounce renderer from [Iseringhausen and Hullin 2020] requires one second to render scenes that have more than a few hundreds of primitives. The proposed rasterization-based rendering method renders both simple and complex scenes at real-time rates that are 30× faster than [Iseringhausen and Hullin 2020].

4 LEARNED NLOS SCENE REPRESENTATIONS

We propose an end-to-end approach to learn 3D representations from transient images. An overview of the proposed method is shown in Fig. 6. Given a transient image τ, we learn a 3D feature embedding C, which allows for diverse tasks including imaging, depth reconstruction, classification, and object detection — all learned in an end-to-end fashion with real-time inference.

At the core of the algorithm lies a learned volumetric feature representation of the 3D object. Instead of directly estimating volumetric albedo and density as in existing NLOS reconstruction methods, we learn at each voxel a latent vector that encodes shape and color information of the hidden volume. In other words, our representation differs from voxel-albedo (or voxel-color) because the latent volumetric feature vectors encode shape, occlusion, normal, semantics, etc. and not only albedo. Thinking of the learned features as generalizations of phasors from radiance transients provides an intuition. This representation is essential in making the proposed method generalize to unseen scenes and allows for real-time runtimes with low memory consumption. We obtain this embedding in two steps. We first extract 2D spatio-temporal features using a convolutional network. This step is motivated by the fact that transient images, as shown in Figure 5, are sparse with large areas of low entropy.

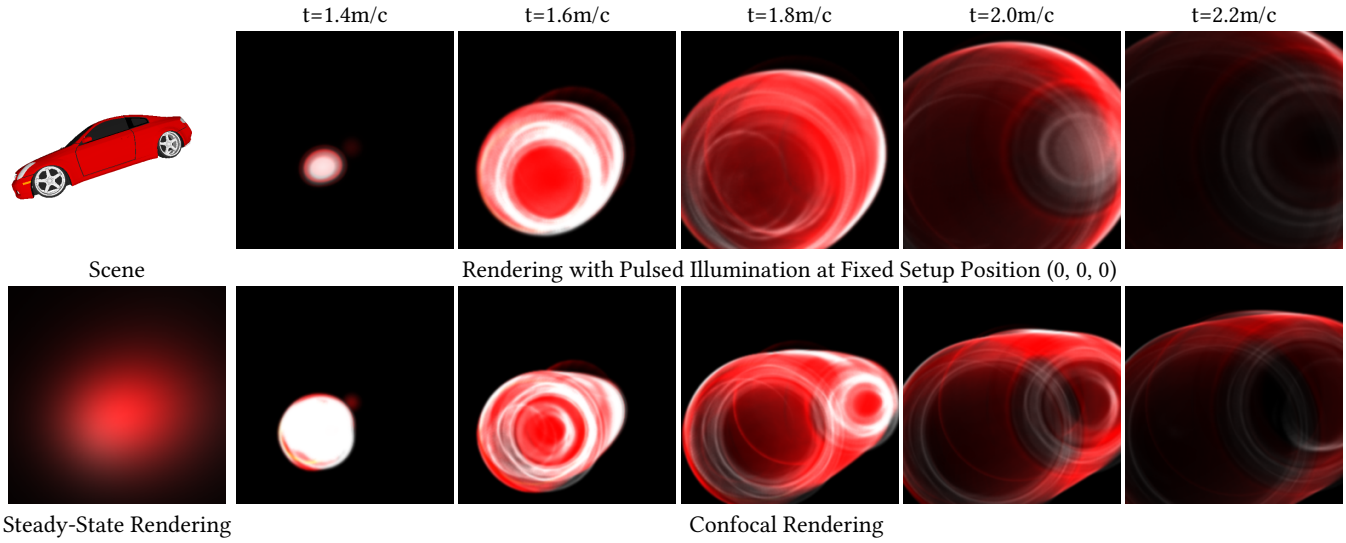

Fig. 5. **Transient Rasterization.** Our rendering pipeline renders transient images using hardware-accelerated rasterization, supports confocal and non-confocal setups and allows to render steady-state indirect reflections. To illustrate quality of the rendered transients we show images without the proposed detector noise model. Our approach renders a car model with 52081 vertices and 200018 faces to *full transient measurement cubes with* $256 \times 256 \times 600$ *spatio-temporal resolution at interactive rates* on consumer GPU hardware within 117ms. Given scene (top left), we show the synthesized transient video frames at different travel times (right 5 columns). Top: transient images illuminated in a setup with a single pulsed light source at (0, 0, 0). Bottom: transient images in a confocal setup. The steady-state rendering without time resolution is shown in the bottom left.

Instead of propagating *all intensity values* to a hidden volume, e.g., as in backprojection methods [Velten et al. 2012], we reason only on features that are critical for reconstruction, e.g., spherical wavefront shapes of scene objects instead of measurement noise or ambient background. These extracted features occupy a significantly smaller latent space than the original intensity values. In the second step, we propagate these features into the spatial target volume. This feature propagation can be either learned, or based on existing physical propagation models, such as backprojection [Velten et al. 2012; O'Toole et al. 2018b; Liu et al. 2019]. We formulate the individual steps in the latent feature extraction as

$$C_t = \mathcal{F}_e(\tau) \qquad \text{Feature Extraction} \quad (6)$$

$$C_s = \mathcal{F}_{t \to s}(C_t), \qquad \text{Feature Propagation} \quad (7)$$

where $\mathcal{F}_e$ and $\mathcal{F}_{t \to s}$ denote the feature extraction and propagation unit, respectively, and C_s and C_t are the extracted spatio-temporal feature and the 3D spatial feature, respectively. The learned embedding C_s is 3D-aware and can be used to reconstruct the hidden object and perform different semantic understanding tasks, which we discuss in Sec. 5.

4.1 Spatio-Temporal Feature Extraction

For an RGB transient image with 512 time bins and a spatial resolution of 256×256, the feature extraction network takes as input a tensor of size $(512, 256, 256, 3)$ and immediately applies a convolutional downsampling block to reduce the amount of data. The downsampling block is composed of two branches. The first branch contains one convolutional layer, and the second branch includes another convolutional layer followed by one ResNet block [He et al. 2016] to refine the downsampled features. Each ResNet block contains two convolutional layers, interlaced with one LeakyReLU layer. All convolutional layers have kernel size 3, stride 1, and three output channels (limited by our training hardware memory), except for the first convolutional layer of both branches. These first layers have stride 2 spatially and temporally to immediately compress features in the spatio-temporal domain. The outputs of the two branches are concatenated along the channels, resulting in a final extracted feature of size $(256, 128, 128, 6)$, i.e. $\approx 4\times$ smaller in size than the input raw data, see the Supplemental Material. While we assume a spatial feature resolution of 128 throughout this work, the spatial feature resolution and the number of channels is a free architecture choice, and we analyze different resolutions in Sec. 6.

4.2 Latent Feature Propagation

Learning hidden 3D representations from transient images requires transforming spatio-temporal information into a representation in the hidden spatial domain. To tackle this challenge, a large body of work [Pandharkar et al. 2011; Velten et al. 2012; Gupta et al. 2012; Kadambi et al. 2016; O'Toole et al. 2018a; Tsai et al. 2017; Arellano et al. 2017; Pediredla et al. 2017; O'Toole et al. 2018b; Xu et al. 2018; Heide et al. 2019; Liu et al. 2019] has explored inverse filtering and optimization methods that rely on approximate physical forward models. While convolutional deep learning has been shown to be effective for 3D reconstruction tasks using convolutional features for local feature extraction [Çiçek et al. 2016; Wu et al. 2015; Choy et al. 2016; Richter and Roth 2018], learning non-local representations that require spatial transformations is still an open problem [Wang et al. 2018; Jaderberg et al. 2015]. Indeed, common operations in deep

models have been shown to be excellent at extracting translation-invariant local details.

We propose to incorporate physical models to tackle this challenge. Given extracted features, the feature propagation network globally reasons about the shape over time and converts the information to the spatial domain instance as $\mathcal{F}_{t\to s}$ in Eq (7), which propagates spatio-temporal features $C_t \in \mathbb{R}^{c\times t\times h\times w}$ to 3D spatial features $C_s \in \mathbb{R}^{c\times d\times h\times w}$. This idea of feature propagation for time-to-space transformation is, in fact, compatible with a variety of existing methods. For example, one can replace $\mathcal{F}_{t\to s}$ with different physical model-based approaches such as the Back Projection (BP), Light Cone Transformation (LCT) [O'Toole et al. 2018a], Fast F-K Migration (F-K) [Lindell et al. 2019b], or a learnable algorithm such as a U-Net [Ronneberger et al. 2015] – all operating on feature vectors instead of intensity measurements. As a result, the proposed feature propagation network allows us to encode and propagate higher-level information beyond intensity. Moreover, the input to the feature propagation network decreases cubically compared to raw data, allowing for reduced runtime and memory footprint while enabling efficient high-resolution reconstructions. For example, given transient volumes of size $L\times L\times L$ and a downsampling factor of D, compared to methods with runtime complexity $O(L^3\log(L))$ and memory requirement $O(L^3)$, such as [O'Toole et al. 2018a; Lindell et al. 2019b], this results in a super-cubic speedup of $D^3\cdot\log(L)/\log(L/D)$ and cubic memory reduction of factor D^3, compared to existing methods.

4.3 Feature Abstraction

To further abstract and complete hidden scene information, we process the propagated feature from last step with an additional volumetric embedding block. In particular, after feature propagation, the volumetric representation is passed through a 3D convolutional layer with kernel size 3, stride 1, and output channel number as 6 without bias parameter. Instead of opting for a larger ResNet block, we initialize the weights such that its output is identical to its input when the training starts. This feature abstraction block aims at further abstracting and filling holes of the encoded representation before using it for reconstruction or recognition tasks. The output of this block is the final learned volumetric feature representation.

5 END-TO-END NLOS NETWORKS

The learned feature representation allows us to train learned methods for different NLOS tasks, in an end-to-end fashion, jointly with the feature extraction, propagation, and abstraction units described in the previous section. For the recovery of NLOS 2D images, we first estimate a 3D visibility map using a 3D convolutional network. We then collapse the 3D volume to a 2D output feature map, by accumulating the 3D feature map scaled by the visibility map for all voxels along the ray corresponding to each 2D pixel. Finally, we process the 2D feature map with a rendering network that includes upsampling layers, resulting in a high-resolution RGB image. A similar approach can be used to produce the corresponding NLOS 2D depth map in the hidden volume. In contrast, we train recognition tasks, such as classification and object detection, directly from the intermediate feature map. In the following, we discuss the differentiable modules that are used for the diverse NLOS tasks we address with the proposed method.

5.1 2D Rendering

Illustrated in Figure 6, the proposed rendering network consists of four modules tailored to the specific NLOS task the method tackles: (1) a view transformer that spatially transforms a 3D feature map based on camera positions, (2) a visibility network that predicts visibility over the volumetric embedding, (3) a differentiable renderer that renders an RGB image given a collapsed 2D feature of the hidden object, and (4) a depth estimator that reconstructs a depth map given the 3D representation and the corresponding visibility map.

View Transformer. We start with the 3D feature volume C_s of size (c, d, h, w), where d, h, w are the depth, height, and width of the volume and c is the encoded feature length at each location. Suppose we wish to render an orthogonal view defined by the virtual camera with look-at point at the center of the hidden volume. We define this camera by a rotation matrix $R \in \mathbb{R}^{3\times3}$ representing camera rotation around the hidden volume center. To render an intensity or depth image captured by the virtual camera, C_s is first spatially transformed by

$$C'_s = \mathcal{S}(C_s \cdot R^T), \tag{8}$$

where S is an operator that computes the values in the final discretized image space by sampling from the rotated feature with bilinear interpolation. For a hidden scene reconstruction from the "canonical" orthogonal view, corresponding to the direction towards the relay wall, R is an identity matrix.

Visibility Network. For hidden rendering tasks, hidden scene features embedded on surfaces visible to the virtual camera. To this end, we model visibility with a visibility map over the volumetric embeddings to predict features on the hidden surface. The visibility map is set to one for the voxel that has the maximum activation along all depth levels, and zero otherwise. With this visibility map, the feature embeddings C_s are collapsed onto a planar representation p of size (c, h, w), that is

$$p_{i,j,k} = \sum_{u=1}^{d} C'_{s_{i,u,j,k}} v_{u,j,k}. \tag{9}$$

After collapsing the features, $p_{\cdot,j,k}$ encodes only the features visible to the virtual camera across all depth planes at image position (j, k).

Image Rendering Network. To produce a 2D image from a collapsed feature map, we implement the rendering process with a convolutional network to decode the embedded information to intensity color channels. The network upsamples the feature to a higher resolution and outputs an RGB image

$$I = \mathcal{F}_{render}(p). \tag{10}$$

Depth Rendering Network. Depth estimation requires an input feature that encodes surface location along the viewing ray. The visibility map by definition provides such information, but its discretization is tailored to the feature locations that live in coarse 3D grids. To refine depth embedded in the visibility map, we rely on the collapsed planar feature p. Specifically, we concatenate the visibility

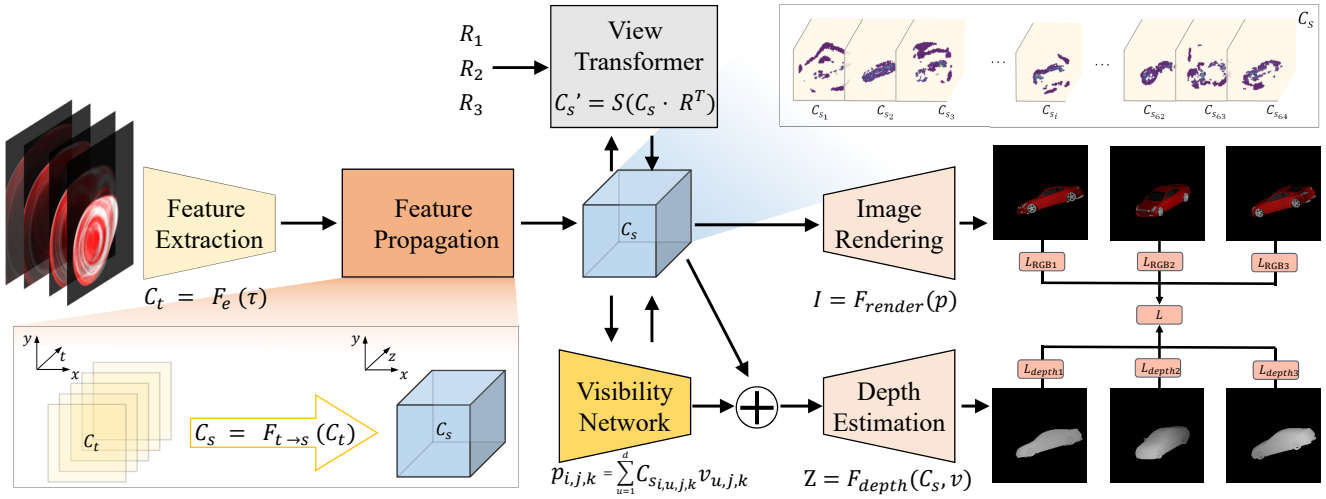

Fig. 6. **Overview of the proposed Feature Reconstruction Network.** At the core of the algorithm lies a learned feature embedding that lives in the hidden reconstruction volume. We first extract features from a transient input image, which are propagated to the hidden volume. Once the features are embedded in hidden volume, a visibility network is applied on the learned 3D feature to estimate a visibility map from the orthogonal view and flattens the 3D feature into 2D. In the final step, an image rendering network takes the flattened feature map as input and predicts the RGB image, while a depth estimator uses the concatenated visibility map and flattened feature map to predict a depth map.

map v and the collapsed feature p as input to our depth estimation network, which outputs a depth map

$$Z = \mathcal{F}_{depth}(v, p). \tag{11}$$

5.2 RGB-D Reconstruction

With the building blocks from the previous paragraphs in hand, we next describe how we train an end-to-end network to perform RGB-D reconstruction. The transient image first goes through feature extraction, propagation, and abstraction to be embedded into the proposed volumetric representation. The visibility network is then applied to estimate a visibility map from the canonical view, which is flattened to produce a 2D feature map. In the last step, the image rendering and depth rendering networks produce RGB and depth maps. These are penalized by the overall loss

$$L = \alpha L_{RGB} + \beta L_{depth} = \alpha \sum_{i=1}^{h \cdot w} (I_i^{pr} - I_i^{gt})^2 + \beta \sum_{i=1}^{h \cdot w} (Z_i^{pr} - Z_i^{gt})^2, \tag{12}$$

where pr and gt denote prediction and ground truth, respectively. The loss weights α and β control the loss contribution of each term, and we set $\alpha = \beta = 1$ for all experiments. Note that all stages along the way, as well as the final loss, are differentiable, allowing us to use backpropagation to train weights. This includes both the image and depth rendering networks from the previous paragraphs, which differentiably render scenes represented as latent embeddings learned from transient images. The loss in Eq. 12 hence penalizes these networks to learn rendering in the training, i.e., minimization of the loss.

Multi-View Supervision. To aid the representation learning, we add multi-view supervision. Specifically, during training, the hidden feature volume is simultaneously rendered from multiple random camera views. To render non-canonical views, the learned volumetric representation is first reprojected by the view transformer based on a new camera position $R' \in G$, where G is the set of sample views. The reprojected volume is then passed to the subsequent image renderer and depth estimator. Similar to single-view RGB-D reconstruction, multi-view supervision also applies to both RGB and depth images, resulting in the following multi-view loss

$$L = \alpha \sum_{j=1}^{m} L_{RGB_j} + \beta \sum_{j=1}^{m} L_{depth_j}, \tag{13}$$

where m is the total number of supervised views. We have found that incorporating multi-view consistency from random views helps to training learn more generalizable embeddings.

5.3 Classification

Aside from geometry and reflectance, the learned representation C_s also efficiently encodes semantic information. We rely on the proposed encoding to perform end-to-end recognition such as classification and hidden object detection as follows. For the task of r-class classification, we feed C_s into a convolutional network to predict the input class labels. The classification network is composed of five 3D convolutional layers. C_s is first downsampled by four convolutional layers with kernel size 3 and stride 2, then is convolved with a fifth layer with kernel size 4, becoming a vector of length r for class prediction. We use a softmax loss for training, that is

$$L = \sum_{i=1}^{r} -\log\left(\frac{exp(p_i^{gt})}{\sum_{i=1}^{r} exp(p_j^{pr})}\right), \tag{14}$$

where p_i^{gt} and p_j^{pr} are the ground-truth class label for class i and prediction for class j, respectively.

5.4 Object Detection

We formulate hidden object detection as predicting a bounding box $(x_{\min}, x_{\max}, y_{\min}, y_{\max}, z_{\min})$. For depth, we predict only the near boundary $z_{\min}$ and not the far boundary, because the input measurement only partially captures the front surface of the hidden object. To predict the bounding box, we use a convolutional network that takes the collapsed 2D feature map p as input and outputs five values for regression during training. The network consists of four convolutional layers with stride two and kernel size three, followed by an average pooling layer that extracts a one-dimensional feature of length 512 and a fully-connected layer that predicts five values. The loss function is sum of squared differences of the regressed box coordinates, that is

$$L = \sum_{\substack{u \in \{x_{\min}, x_{\max}, \\ y_{\min}, y_{\max}, z_{\min}\}}} (u^{gt} - u^{pr})^2. \tag{15}$$

6 ANALYSIS AND SYNTHETIC VALIDATION

In this section, we validate the the proposed method on synthetic transient image data.

6.1 Synthetic Dataset

For training and validation in simulation, we create several synthetic transient image datasets rendered from ShapeNet [Chang et al. 2015]. The transient images are rendered in non-confocal and confocal setups as histograms with 33 ps bin resolution (corresponding to a travel time of 1 cm), histogram length of 512, and 256×256 spatial scanning resolution. To evaluate the generalization of the proposed method in an extreme setting, from a single class during training to multiple unseen classes or measured data, we render a motorbike dataset and a car dataset. The car dataset consists of 2244 transient data cubes rendered from 2244 different cars, with each car rendered with one random transformation. The motorbike dataset consists of 6925 transient images rendered from 277 different motorbikes, with each motorbike rendered from 25 random transformations. To sample random model transformations, we first rotate the object with a rotation uniformly sampled from the range yaw $\in [-180°, 180°]$, roll $\in [-20°, 20°]$, pitch $\in [-20°, 20°]$, and then shift the object by an offset uniformly sampled from $[-0.3, 0.3]$ for all coordinate axes. Moreover, we also evaluate the proposed method when trained on multi-class data. To this end, we render a dataset consisting of the top 13 classes with the most number of examples in ShapeNet [Chang et al. 2015], where for each class 446 to 500 transient images are rendered for different object instances. For all the datasets, the training, validation, and testing split is 8:1:1, and views of the testing objects are unseen during training. We refer to the Supplemental Material for additional training details.

We apply noise calibrated for $N = 20k$ pulses with $s = 0.02$ for our measurement model from Sec 3.1. We normalize the transient measurements by their 99th percentile to range $[0, 1]$. For multi-view supervision, two views are used for each object during training. The two views include one fixed orthogonal view and a random non-orthogonal view. The non-orthogonal view is rendered by uniform random rotation of the camera around the center of the hidden volume with fixed distance to the hidden volume center. As transient information is biased to surfaces facing the relay wall, we limit the maximum deviation to 25 degrees in order to prevent the model from hallucinating occluded parts.

Test Score	FBP [2012]	LCT [2018b]	F-K [2019b]	Proposed
PSNR [dB]	19.72	19.06	23.74	29.29
SSIM	0.25	0.51	0.80	0.92

Table 2. **NLOS Reconstruction Evaluation.** PSNR and SSIM comparison between the proposed RGB-D model trained on multi-class with multi-view supervision and state-of-the-art methods after maximum intensity-projection along the z-axis. All methods are evaluated on a held-out testing set composed of 643 multi-class examples, with transient histograms of 512 time bins and a spatial resolution of 256×256. We note that the transients are not cropped in the temporal domain. The proposed model outperforms existing methods by more than 5 dB in PSNR.

6.2 2D Image Reconstruction

We first evaluate NLOS 2D image reconstruction on the multi-class data set. Tab. 2 lists quantitative evaluations. We highlight that the proposed model outperforms existing methods by a large margin of more than 5 dB in PSNR. In Fig. 7, we visualize qualitative results. Compared to F-K [Lindell et al. 2019b], LCT [O'Toole et al. 2018a], and filtered back-projection (FBP) [Velten et al. 2012] (all evaluated with unmodified code from [Lindell et al. 2019b]), we observe that the our learned method is able to reconstruct 2D images with clearer boundaries while achieving more accurate color rendering. The first column shows an example where our model is able to reconstruct details on the front surface while F-K fails to recover fine details and LCT outputs only rough blurred shapes. In the 7th column, the proposed approach reconstructs the rear light in contrast to existing methods.

While the existing methods rely on physical models and do not facilitate learning rich scene priors, the proposed model, however, relies on deep convolutional networks that can overfit when trained on small datasets or data that is not representative of real measurements. To validate our model, we also assess the generalization ability of the proposed approach. We train a model on the *single-class* motorbike dataset, and evaluate it on both unseen object of the same class motorbike and *unseen classes*; see Fig. 8. The proposed method not only faithfully reconstructs orthogonal view NLOS images for unseen objects of the same class, see Supplementary Material, but also generalizes well to diverse unseen classes. Trained only on the motorbike class, the proposed model is able to reconstruct other fine structures and pattern that do not exist in the training data set, for example, the thin structures on lamps, ships and chair backs.

6.3 Depth and Multi-View Image Reconstruction

As described in Section 5.1, by adding a depth rendering network and multi-view supervision, the proposed method supports joint image and depth reconstruction from multiple viewpoints in an end-to-end fashion. To assess multi-view RGB-D recovery, we train an RGB-D model with depth supervision on the multi-class dataset. For brevity, we refer to the Supplementary Material for qualitative multi-view reconstruction results. Fig. 9 shows depth reconstruction comparisons. For the methods compared, we apply maximum

Fig. 7. **Qualitative Evaluation for NLOS 2D Imaging.** Compared to F-K [Lindell et al. 2019b], LCT [O'Toole et al. 2018a], and filtered back-projection (FBP) [Velten et al. 2012] (unmodified code from [Lindell et al. 2019b] for all comparisons), we observe that the proposed method is able to reconstruct 2D images with clearer boundaries while achieving more accurate color rendering. For example in the first column, the proposed model is able to reconstruct details on the front surface while F-K fails to recover fine details and LCT outputs a much blurrier estimates.

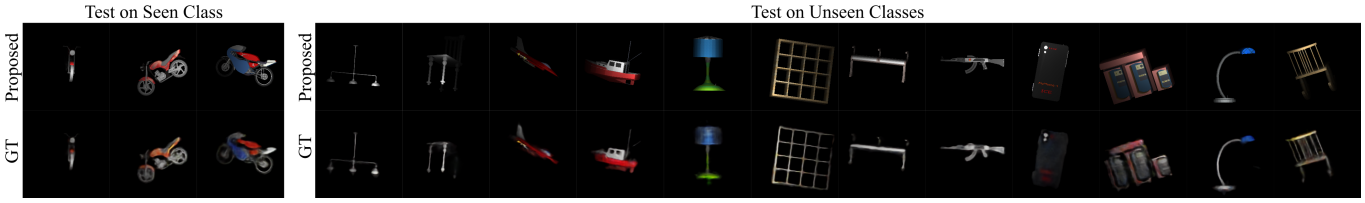

Fig. 8. **Generalization to Unseen Classes.** We note that the proposed model, trained only on motorbikes, not only faithfully reconstructs unseen motorbikes (left), but also generalizes well on other unseen classes (right). We observe that even thin structures can be recovered well by the proposed method, e.g., the first lamp and the antenna on the fourth watercraft.

intensity projection along z axis. Specifically, for each pixel on $x-y$ plane, we find the voxel with the maximum intensity along z axis, and use this voxel's z position as predicted depth at location (x, y). Despite the complex geometry of the compared scene, the proposed approach recovers fine structures with a smaller error compared to existing F-K [Lindell et al. 2019b] and LCT [O'Toole et al. 2018b] methods. As also evident from the individual depth map reconstructions, especially in model parts distanced further from the relay wall, the proposed method excels. Please see the Supplemental Material for additional RGB, depth, and multi-view reconstruction evaluations.

6.4 Ablation Study and Analysis

Next, we analyze the influence of different network architecture components for the key modules in the proposed method. First, we show how adding depth and multi-view prediction impacts NLOS image reconstruction, and then we compare the performance of using different methods in our feature propagation network. Moreover, we also analyze the resolutions of the feature map in the proposed feature propagation module. For each comparison, we use the same baseline model. This model is trained on the car data set for single-view 2D NLOS image reconstruction. The model uses $(d, h, w) = (32, 32, 32)$ as the feature map resolution for the feature propagation network.

Depth and Multi-view Prediction. In the left of Tab. 3, we analyze the influence of depth and multi-view supervision on NLOS RGB reconstruction. Adding depth does not significantly influence 2D NLOS image reconstruction, while multi-view supervision helps improve single-view recovery.

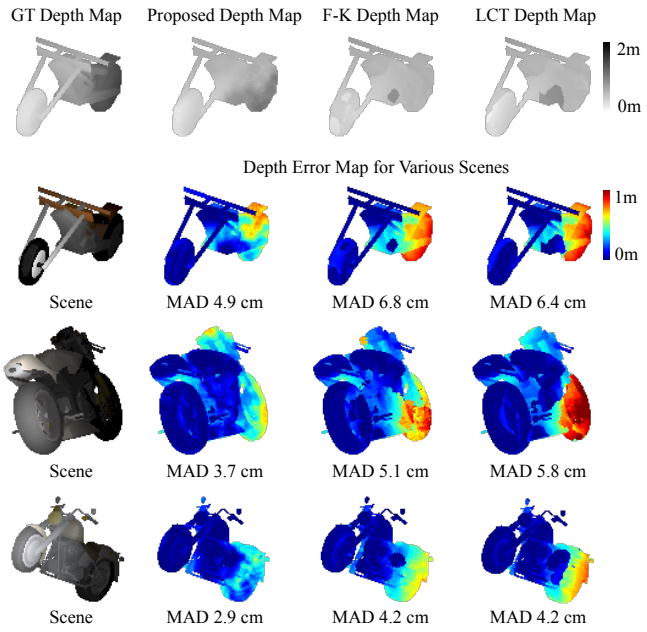

Fig. 9. **Depth Prediction.** We visualize the error in depth prediction of a synthetic motorbike and compute Mean Absolute Distance (MAD) for each method. Our method predicts more accurate depth compared to the F-K and LCT methods, especially in challenging model parts that are distanced further from the relay wall.

Feature Propagation Units. Next, we analyze the impact of different feature propagation units $\mathcal{F}_{t\to s}$ on the reconstruction quality. In the middle table in Tab. 3, we compare models with different propagation approaches. The first model uses a learned 3D convolutional U-Net architecture and the last three methods use physical propagation methods as feature propagation units. The U-Net has four downsampling and four upsampling steps. At each downsampling step a $3 \times 3 \times 3$ convolution with stride 2 and output channel number doubled is applied and followed by an instance normalization layer and LeakyReLU. Each upsampling step consists of a $3 \times 3 \times 3$ up-convolution with stride 2 that halves the number of input feature channels, an instance normalization layer, a ReLU, and a concatenation with a feature map from its corresponding downsampling stage. We note that the learned U-Net has the weakest reconstruction performance and struggles to learn the global spatial transformation of the NLOS image formation. Moreover, with the U-Net as learned propagation block, the resulting architecture also has a large number of learnable parameters (slightly over 24 million for eight 3D convolutional layers) in feature propagation network, which makes this model not only harder to optimize but also prone to overfitting. Comparing existing physically-based reconstruction methods, we find that, perhaps interestingly, the performance of filtered backprojection is very comparable with that of the LCT-based propagation block [O'Toole et al. 2018b]. Fig. 10 documents qualitative comparisons. We see that all methods are able to predict the rough shape and color of the hidden object. While the LCT and FBP-based propagation units perform on-par, Stolt's F-K migration [Stolt 1978] as a propagation block performs the best among all of them. For example, even the small feature volume of size $16 \times 32 \times 32$ is able to recover the red light at the top-right corner of the first car, the front glass of the third car in the right shape and color, and the rear light and window frame of the last car.

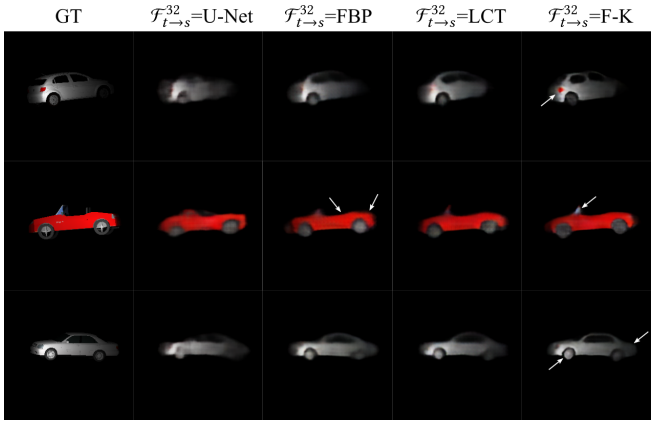

Fig. 10. **Analysis of the Feature Propagation Module.** We illustrate the impact of different feature propagation methods in our network architecture. While all variants are able to predict the rough shape and color of the hidden object, FBP and LCT perform on-par, and an F-K feature propagation aids the reconstruction of our network. See the red rear light of the vehicle in the first row, and the windshield in the third row.

Method	MSE [$\times 10^{-3}$]	PSNR [dB]
RGB	6.34	22.87
RGB-D	6.45	22.75
RGB Multi-view	6.20	23.08
RGB-D Multi-view	6.21	23.05

$\mathcal{F}_{t\to s}$	MSE [$\times 10^{-3}$]	PSNR [dB]
U-Net	8.15	21.96
FBP	6.40	22.81
LCT	6.34	22.87
F-K	5.72	23.28

Resolution	MSE [$\times 10^{-3}$]	PSNR [dB]
32	6.34	22.87
64	5.10	23.61

Table 3. **Ablations and Analysis of NLOS Image and Depth Reconstruction Networks.** The left table shows that multi-view prediction improves performance and that RGB model and RGB-D model perform similarly. The center table compares reconstruction performance using different feature propagation methods. The right table illustrates how resolution of the learned volumetric representation influences the performance. Experiments for left two tables use a feature resolution of 32.

Feature Embedding Resolution. Finally, we also analyze the effect of the resolution of the latent feature embedding. Tab. 3 lists how the feature map resolution affects the performance. Increasing the resolution from $16 \times 32 \times 32$ to $32 \times 64 \times 64$ (both with 32 feature channels) results in a large performance gain. We also observe that with resolution of $32 \times 64 \times 64$, our model is able to preserve more details than the smaller model. We did not observe further gains at higher resolutions for the given setup configuration.

6.5 Object Recognition

In contrast to existing optimization and filtering-based methods, the proposed approach facilitates learning NLOS reconstruction jointly with a downstream recognition or imaging task, in an end-to-end fashion. Next, we demonstrate how the proposed model allows for end-to-end classification and detection of hidden objects.

Classifier	Airplane	Lamp	Firearm	Chair	Watercraft	Car	Motorbike	Overall
$Classifier_{GT\text{-}image}$	70.0%	65.3%	54.1%	90.0%	62.0%	73.4%	76.7%	70.2%
$Classifier_{sequential\text{-}F\text{-}K}$	68.0%	48.9%	27.0%	64.0%	38.0%	44.8%	64.2%	50.7%
$Classifier_{sequential\text{-}ours}$	64.0%	51.0%	33.3%	66.0%	30.0%	46.9%	58.9%	50.0%
$Classifier_{end\text{-}to\text{-}end}$	68.0%	67.3%	56.3%	82.0%	52.0%	81.6%	57.1%	66.3%
Prediction on Bike Scene	0.1%	1.8%	11.5%	7.1%	10.7%	3.0%	66.1%	

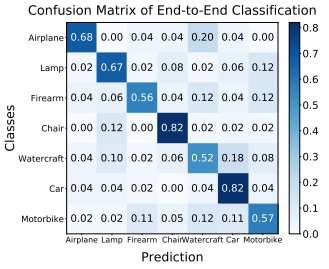

Table 4. **End-to-end NLOS Classification on Synthetic and Real Data** We compare the classification accuracy of the proposed method, learned to classifying hidden scenes with a monolithic end-to-end network ($Classifier_{end\text{-}to\text{-}end}$), and sequential NLOS image classification baselines. For these baselines, we train and evaluate a 2D classifier on the ground truth albedo maps ($Classifier_{GT\text{-}image}$), FK [Lindell et al. 2019b] albedo prediction using maximum-intensity projection ($Classifier_{sequential\text{-}F\text{-}K}$), and intermediary albedo map produced by the proposed method ($Classifier_{sequential\text{-}ours}$). The first four rows of the table report accuracy on the synthetic test set, see text, where the proposed end-to-end classifier $Classifier_{end\text{-}to\text{-}end}$ (66.3%) trained in feature space outperforms the sequential method $Classifier_{sequential\text{-}F\text{-}K}$ (50.7%) and $Classifier_{sequential\text{-}ours}$ (50.0%) trained on intermediate NLOS images. We report the confusion matrix for the proposed end-to-end classifier on the right. The last row in the table reports the confidence scores for the experimental bike measurement. We note that the proposed model recognizes it as a motorbike with more than 66% probability.

While [Caramazza et al. 2018b] show that neural networks can be used to identify hidden pedestrians, assuming only a single pedestrian in the scene, we tackle multi-class classification and detection, discriminating objects with various shapes and categories. For training, we use a subset of class from the confocal multi-class data set described in Sec. 6.1, including seven classes (plane, car, chair, lamp, motorbike, firearm, and watercraft). Each class has 500 examples split into training, validation and testing sets with 8:1:1.

For classification, we replace the rendering network with a classification network $Classifier_{end\text{-}to\text{-}end}$; see Sec. 5.4. This network ingests the feature map p as input and uses four downsampling convolutional layers to predict the probability of each category directly from the learned feature encoding. It is trained in an end-to-end fashion only supervised by the classification loss. We compare the proposed approach to sequential image reconstruction, producing an intermediary image, followed by conventional classification. To this end, we train a 2D image classifier $Classifier_{sequential\text{-}F\text{-}K}$ on intermediate reconstructions from the F-K and our learned reconstruction method. For the sequential F-K method, we use maximum intensity projections as intermediate image reconstruction while for our method, we use our prediction as the intermediate image. We then train classifiers with matching network capacity on the images. These sequential 2D classifiers take the projected 2D image of size $3 \times 256 \times 256$ size as input to predict seven classification scores. Tab. 4 (left) validates that the proposed end-to-end classification approach *outperforms existing sequential methods by more than 15% in accuracy*, including ones using 2D images produced by our reconstruction network.

For object detection, we initialize the detection head with random weights as described in Sec. 5.4 and use a pre-trained image reconstruction model to initialize the parameters of the feature extraction and abstraction network. Then the entire model is trained with 2.5D bounding box regression, on a data set containing 14 classes. We report 2D IoU (Intersection over Union) between ground truth and predicted bounding box projected onto $x - y$ plane. In Figure 11 , we report our end-to-end detection results and compare with two

Detector	Avg. IoU	Cabinet	Chair	Display	Firearm	Table	Watercraft	Car	Bike
$Detector_{end\text{-}to\text{-}end}$	0.73	0.80	0.75	0.72	0.73	0.73	0.81	0.74	0.74
$Detector_{sequential\text{-}ours}$	0.69	0.79	0.75	0.72	0.64	0.70	0.79	0.73	0.68
$Detector_{sequential\text{-}F\text{-}K}$	0.67	0.78	0.70	0.70	0.65	0.68	0.75	0.71	0.69

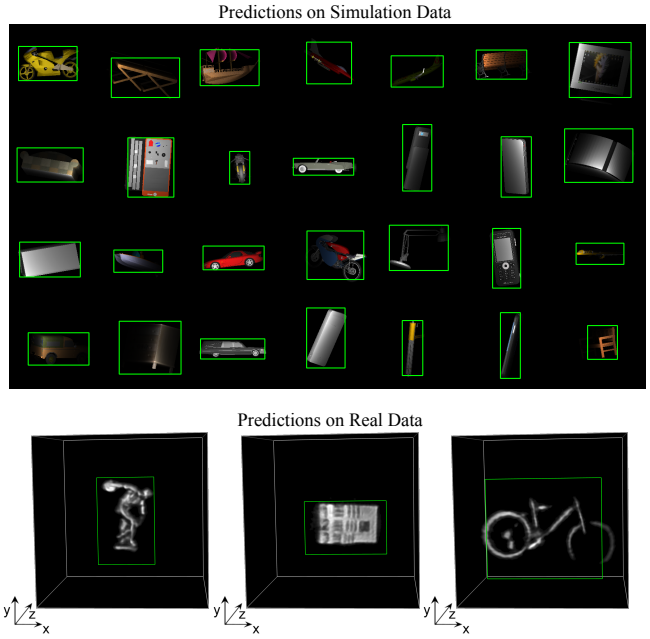

Fig. 11. **2.5D Object Detection.** We compare the proposed end-to-end detector (End-to-end) to two sequential detectors trained on images and depth as intermediary representation. Specifically, we train a detector on RGB-D images from our reconstruction method (Sequential-Ours) and a detection model with input image and depth from F-K (Sequential-FK); see text. Top: IoU evaluation on the synthetic test set, validating that proposed end-to-end model outperforms sequential detection. Center: End-to-end detection result on simulation data. Bottom: Evaluation of the synthetic model on real data validating its generalization capability.

baselines whose inputs are rendered image and depth: a detection model with input rendered from the proposed RGB-D method and a detection model with input rendered from F-K. In the first column, we show average IoU across all classes and IoU for seven individual classes with top per-class IoU. We note that the proposed model performs the best in this task, indicating the advantage of end-to-end detection with learned features over sequential detection – even when compared to the proposed method producing the intermediary image. Fig. 11 shows qualitative end-to-end detection results on both simulation and real data. Please see Supplemental Material for a full IoU table for all 14 classes and additional qualitative detection results.

7 EXPERIMENTAL VALIDATION

In this section, we assess the proposed method on experimentally acquired measurements.

7.1 RGB-D Imaging

To validate that the proposed method generalizes to unseen experimental data, we test the proposed model trained on the synthetic motorbike-only dataset on the experimental dataset from [Lindell et al. 2019b], which contains diverse unseen scenes. Specifically, pulse-scanned confocal measurements are acquired for a dragon, a bicycle, a statue, a resolution table, a disco ball and an indoor scene. As all the real captures are single wavelength captures, we train our model in grayscale as an approximation to a single wavelength model. The proposed method is an RGB-D NLOS reconstruction model using an F-K $\mathcal{F}_{t\to s}$ feature propagation network with resolution 128. The full network recovery performs at real-time rates of at 20 reconstructions per second, on the full transient image without temporal cropping. We refer to the Supplemental Material for additional training details. Fig. 12 shows reconstruction results compared to state-of-the-art physically-based reconstruction methods, including F-K migration [Lindell et al. 2019b], LCT [O'Toole et al. 2018a], Phasor NLOS [Liu et al. 2019] and filtered back-projection (FBP) [Velten et al. 2012], which we discuss in the following. We note that all compared method take as input the full time-resolved transient sequences *without any temporal cropping*.

Generalization. The first column in Fig. 12 illustrates that the proposed model predicts plausible results for all the real captures with diverse shapes (dragon, statue, bike and resolution table) and even complex geometry arrangement (indoor scene), which all have not been seen during training on the completely synthetic motorbike data set. These results validate the generalization ability of our model in two aspects. First, the proposed model is able to generalize from synthetic to real data despite the domain gap. Second, note that none of the tested classes appears in the training set, validating the cross-class generalization on real data that was observed in simulation in Sec. 6. We validate the generalization capability of the proposed architecture by comparing to a vanilla U-Net model, which is trained with simulation data to directly predict RGB-D from input transient images. We refer to the Supplemental Material for network and training details. The fifth column in Fig. 12 confirms that such existing encoder-decoder methods do not generalize to measured data. In the Supplemental Material, we further validate that U-Net models with additional adversarial losses also fail to generalize to real data.

Qualitative Assessment. As all compared methods directly generate a 3D albedo volume, we similarly present the results of our method in a 3D volume by combining the predicted intensity map and depth map. The proposed method produces sharper object boundaries, reveals fine detail missing in other methods, while removing clutter and producing a clean background. For the dragon, statue and bike, existing methods, such as F-K, LCT and FBP suffer from artifacts, including fuzzy outlines and blurred out geometry. While the Phasor NLOS method is able to reconstruct planar hidden scenes relatively sharply, estimates contain a background noise floor resulting from the low signal to noise ratio in the measurements.

The proposed model generates faithful hidden textures and geometry, which can be observed, for example, in the dragon tail, bike axle, and the shadow region in the statue in Fig. 12. The method recovers small geometry and albedo variations such as the statues legs, arms, body and statue base, and it successfully recovers complex scenes as the room scene. Fine detail in the shelf, such as the books, mannequin head, and feature and T-shaped reflective object in the top left are recovered by the proposed method, while they appear blurred in existing methods. Moreover, the statue in the background is recovered at higher contrast compared to previous methods. The proposed network architecture also handles highly specular scenes, such as the discoball scene in row four of Fig. 12 without background artifacts as in the compared methods.

We attribute the improved hidden image recovery without recovery noise to the rich scene priors the network has learned from observing diverse synthetic data. The proposed method learns this prior by working on feature vectors instead of intensity, which makes it possible to embed useful information while suppressing the noise in the feature space. The effectiveness of the learned prior is also confirmed by additional experiments that perform denoising on intermediate outputs from the F-K and Phasor methods. In the Supplemental Material, we validate that learned denoising methods trained on intermediate outputs from existing methods do not offer an alternative to the proposed approach.

7.2 Object Recognition

As reported in Table 4, the proposed end-to-end classification model trained on synthetic data does not only substantially outperform sequential classifiers (even if trained on top of intermediary output images produced by the proposed method) but it also generalizes to measured data. Specifically, we train a model in simulation on grayscale input data, as in Sec. 6.5. Tested on the real bike measurement, the model predicts the correct class as listed in the bottom row of Tab. 6.5, illustrating the effectiveness of our transfer learning method.

Fig. 11 shows that the proposed end-to-end detection model is also able to correctly predict the bounding boxes for different classes with various color, shape, and pose. Both quantitative and qualitative evaluations indicate that our model is able to predict 3D bounding box with a high precision. The 3D predictions on the bottom of Fig. 11 validate, given the small experimental data available, that the proposed end-to-end detector model generalizes to real data.

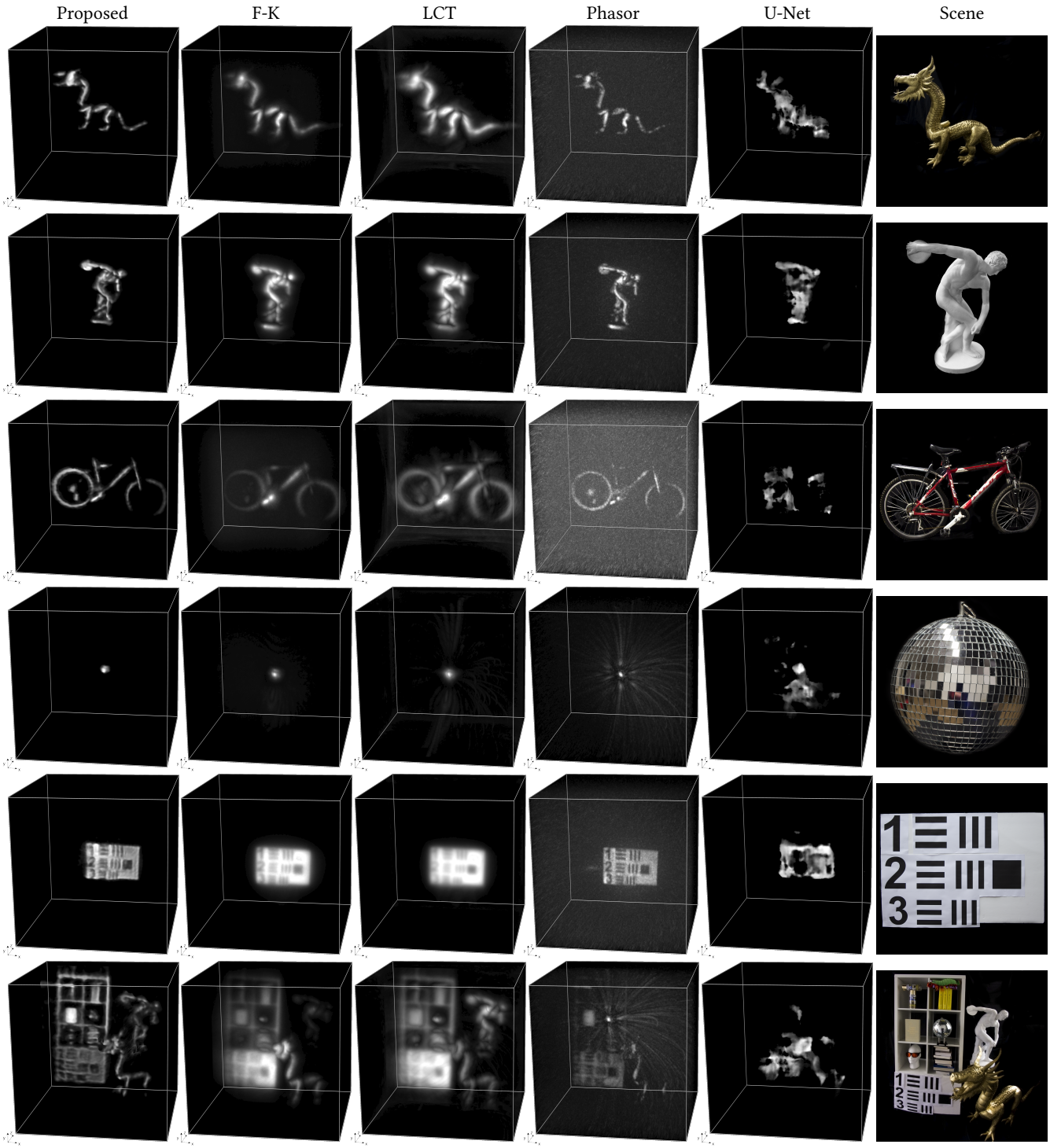

Fig. 12. **Reconstructions from Pulsed Single-Photon Measurements** The proposed learned reconstruction method, trained only on synthetic motorbike dataset, generalizes to transient measurements acquired with the setup described in [Lindell et al. 2019b]. The network handles challenging scenes with complex geometries, occlusions, and varying reflectance. Validating the synthetic assessment, the proposed learned method recovers fine hidden detail, especially with low reflectance, without amplifying reconstruction noise, outperforming existing methods qualitatively and quantitatively.

Method	FBP [2012]	LCT [2018b]	F-K [2019b]	Phasor [2019]	LCT + TV [2018b]	Proposed
Runtime (CPU)	13.22 s	13.29 s	18.44 s	17.64 s	100 min	N/A
Runtime (GPU)	0.28 s	0.24 s	0.43 s	0.48 s	N/A	0.045 s
Memory	15.6 GB	17.7 GB	21.0 GB	10.8 GB	17.7 GB	512 MB

Table 5. **Runtime and Memory Comparisons.** The proposed method is five times faster and consumes less memory than existing methods that do not allow to incorporate priors. It is multiple orders of magnitude faster than existing methods that can incorporate total variation priors. We note that the CPU runtimes and memory estimates are generated with author-provided unmodified MATLAB code, while the GPU implementations of the existing methods are our PyTorch GPU implementations.

We envision our end-to-end detector and classification methods as basic building blocks for future NLOS scene understanding that could analyze complex environments just by observing their indirect reflections.

7.3 Memory and Runtime

To produce a full hidden RGB-D reconstruction from an experimental transient measurement with histogram length of 512, and 256×256 spatial scanning resolution, the proposed method runs at real-time rates of 45 ms on an NVIDIA GeForce RTX 2080 GPU. The peak memory consumption of our method is 512 MB including learnable parameters, calculated by analyzing the data flow in the inference forward pass and with sequential layer execution. As discussed in Sec. 4.1, the proposed method theoretically allows for cubic memory reduction compared to existing methods. We first run existing methods with MATLAB author-provided code on a general-purpose CPU. As shown in the first row of Tab. 5, the existing FBP, LCT, F-K, and Phasor methods require more than 10 seconds per transient measurement, on the CPU, and consume an order of magnitude more memory exceeding 10 GB.

While efficient GPU implementations may allow these methods to achieve real-time runtimes, they do not facilitate the use of priors, and incorporating even traditional gradient priors requires hundreds of iterations of alternating optimization methods, as for the total-variation-regularized LCT variant from [O'Toole et al. 2018b]. Specifically, LCT with total variation penalty requires around 100 minutes (100 iterations at 60 seconds per-iteration) using the original code [O'Toole et al. 2018b] on the CPU. To assess the memory and runtime profiles of existing methods on the GPU in the same inference framework, we also reimplemented existing methods in PyTorch and report the runtime and memory consumption on the GPU in Tab. 5. In this setting, the proposed method is around five times faster and consumes an order of magnitude less memory than existing methods that do not allow to incorporate priors. As the first efficient method that allows to incorporate complex learned scene priors, we envision researchers to build on top of our code and models which we will release.

8 DISCUSSION AND CONCLUSION

We propose to learn feature embeddings tailored to non-line-of-sight imaging and non-line-of-sight recognition tasks, such as classification and hidden object detection. Instead of relying on intensity values to recover and analyze occluded scenes, we propagate and reason in feature space about the hidden scene information, such as shape, reflectance and object type. As such, the proposed method makes a first step towards combining recent deep network architectures, that excel at extracting such features of interest, with physical image formation models, while being trainable in an end-to-end fashion. This allows us to learn rich scene priors which aid NLOS reconstruction and analysis. We show that recovering images or object class from space-time transformed features allows the proposed method to generalize far better than existing encoder-decoder architectures that do not follow this structure. Leveraging physical models for this spatio-temporal transform allows us to learn scene representation which, in contrast to existing albedo representations, natively encode 3D scene structure, reflectance, multi-view consistency, and hidden scene semantics in a compressed form. We train and validate the proposed method on a large simulated transient image dataset, enabled by a novel state-of-the-art transient renderer. The proposed method outperforms the state-of-the-art by more than 5 dB in NLOS image recovery. Although trained on simulated data only, we validate that the method generalizes to experimental data, where it outperforms recent inverse filtering and optimization methods across a variety of scenes, while allowing for real-time reconstruction at low memory consumption. As such, the proposed method is the first efficient method that allows to incorporate learned image priors and end-to-end training into pulsed NLOS recovery and scene understanding.

We foresee this work becoming a building block in more rich end-to-end reconstruction and scene understanding pipelines, making a step towards conceptually turning every scene surface into a sensor. The approach may potentially also motivate similar feature-based reconstruction methods for other challenging inverse problem domains, e.g., fluid reconstruction, x-ray diffraction imaging, or computer generated holography.

ACKNOWLEDGMENTS

We thank our reviewers for their invaluable comments. WC and KK thank the support of NSERC under the RGPIN and COHESA programs, and DARPA under the REVEAL program. FW and SR thank the U. S. National Science Foundation for support under grant IIS-1815070.

REFERENCES

Nils Abramson. 1978. Light-in-flight recording by holography. *Optics Letters* 3, 4 (1978), 121–123.

Victor Arellano, Diego Gutierrez, and Adrian Jarabo. 2017. Fast back-projection for non-line of sight reconstruction. *Optics Express* 25, 10 (2017), 11574–11583.

Katherine L Bouman, Vickie Ye, Adam B Yedidia, Frédo Durand, Gregory W Wornell, Antonio Torralba, and William T Freeman. 2017. Turning corners into cameras: Principles and methods. In *IEEE International Conference on Computer Vision (ICCV)*. 2289–2297.

Samuel Burri. 2016. *Challenges and Solutions to Next-Generation Single-Photon Imagers*. Technical Report. EPFL.

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. *Optics express* 23, 16 (2015), 20997–21011.

Piergiorgio Caramazza, Alessandro Boccolini, Daniel Buschek, Matthias Hullin, Catherine F Higham, Robert Henderson, Roderick Murray-Smith, and Daniele Faccio. 2018a. Neural network identification of people hidden from view with a single-pixel, single-photon detector. *Scientific reports* 8, 1 (2018), 11945.

Piergiorgio Caramazza, Alessandro Boccolini, Daniel Buschek, Matthias Hullin, Catherine F Higham, Robert Henderson, Roderick Murray-Smith, and Daniele Faccio.

2018b. Neural network identification of people hidden from view with a single-pixel, single-photon detector. *Scientific Reports* 8, 1 (2018), 11945.

Susan Chan, Ryan E Warburton, Genevieve Gariepy, Jonathan Leach, and Daniele Faccio. 2017. Non-line-of-sight tracking of people at long range. *Optics express* 25, 9 (2017), 10109–10117.

Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. 2015. *ShapeNet: An Information-Rich 3D Model Repository.* Technical Report arXiv:1512.03012 [cs.GR]. Stanford University — Princeton University — Toyota Technological Institute at Chicago.

Wenzheng Chen, Simon Daneau, Fahim Mannan, and Felix Heide. 2019. Steady-state non-line-of-sight imaging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6790–6799.

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

Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 2016. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European conference on computer vision*. Springer, 628–644.

Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 2016. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*. Springer, 424–432.

PB Coates. 1972. Pile-up corrections in the measurement of lifetimes. *Journal of Physics E: Scientific Instruments* 5, 2 (1972), 148.

Michael F Cohen and Donald P Greenberg. 1985. The hemi-cube: A radiosity solution for complex environments. *ACM Siggraph Computer Graphics* 19, 3 (1985), 31–40.

Qi Guo, Iuri Frosio, Orazio Gallo, Todd Zickler, and Jan Kautz. 2018. Tackling 3d tof artifacts through learning and the flat dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 368–383.

Otkrist Gupta, Thomas Willwacher, Andreas Velten, Ashok Veeraraghavan, and Ramesh Raskar. 2012. Reconstruction of hidden 3D shapes using diffuse reflections. *Opt. Express* 20, 17 (Aug 2012), 19096–19108.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

Felix Heide, Steven Diamond, David B Lindell, and Gordon Wetzstein. 2018. Sub-picosecond photon-efficient 3D imaging using single-photon sensors. *Scientific reports* 8, 1 (2018), 17726.

Felix Heide, Matthias B Hullin, James Gregson, and Wolfgang Heidrich. 2013. Low-budget transient imaging using photonic mixer devices. *ACM Transactions on Graphics (ToG)* 32, 4 (2013), 1–10.

Felix Heide, Matthew O'Toole, Kai Zang, David B Lindell, Steven Diamond, and Gordon Wetzstein. 2019. Non-line-of-sight imaging with partial occluders and surface normals. *ACM Transactions on Graphics (ToG)* 38, 3 (2019), 22.

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

Quercus Hernandez, Diego Gutierrez, and Adrian Jarabo. 2017. A Computational Model of a Single-Photon Avalanche Diode Sensor for Transient Imaging. arXiv:physics.ins-det/1703.02635

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

Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. 2015. Spatial transformer networks. In *Advances in neural information processing systems*. 2017–2025.

Adrian Jarabo and Victor Arellano. 2018. Bidirectional rendering of vector light transport. In *Computer Graphics Forum*, Vol. 37. Wiley Online Library, 96–105.

Adrian Jarabo, Julio Marco, Adolfo Munoz, Raul Buisan, Wojciech Jarosz, and Diego Gutierrez. 2014. A Framework for Transient Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 33, 6 (nov 2014). https://doi.org/10.1145/2661229.2661251

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.

Achuta Kadambi, Refael Whyte, Ayush Bhandari, Lee Streeter, Christopher Barsi, Adrian Dorrington, and Ramesh Raskar. 2013. Coded time of flight cameras: sparse deconvolution to address multipath interference and recover time profiles. *ACM Transactions on Graphics (ToG)* 32, 6 (2013), 167.

Achuta Kadambi, Hang Zhao, Boxin Shi, and Ramesh Raskar. 2016. Occluded imaging with time-of-flight sensors. *ACM Transactions on Graphics (ToG)* 35, 2 (2016), 15.

Ori Katz, Pierre Heidmann, Mathias Fink, and Sylvain Gigan. 2014. Non-invasive single-shot imaging through scattering layers and around corners via speckle correlations. *Nature photonics* 8, 10 (2014), 784.

Ori Katz, Eran Small, and Yaron Silberberg. 2012. Looking around corners and through thin turbid layers in real time with scattered incoherent light. *Nature photonics* 6, 8 (2012), 549–553.

A. Kirmani, T. Hutchison, J. Davis, and R. Raskar. 2009. Looking around the corner using transient imaging. In *IEEE International Conference on Computer Vision (ICCV)*. 159–166.

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

Jonathan Klein, Christoph Peters, Jaime Martín, Martin Laurenzis, and Matthias B Hullin. 2016. Tracking objects outside the line of sight using 2D intensity images. *Scientific reports* 6 (2016), 32491.

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.

David B Lindell, Gordon Wetzstein, and Vladlen Koltun. 2019a. Acoustic non-line-of-sight imaging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6780–6789.

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

Xiaochun Liu, Sebastian Bauer, and Andreas Velten. 2020. Phasor field diffraction based reconstruction for fast non-line-of-sight imaging systems. *Nature Communications* 11 (2020). https://doi.org/10.1038/s41467-020-15157-4

Xiaochun Liu, Ibón Guillén, Marco La Manna, Ji Hyun Nam, Syed Azer Reza, Toan Huu Le, Adrian Jarabo, Diego Gutierrez, and Andreas Velten. 2019. Non-line-of-sight imaging using phasor-field virtual wave optics. *Nature* (2019), 1–4.

Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. 2019. Neural Volumes: Learning Dynamic Renderable Volumes from Images. *ACM Trans. Graph.* 38, 4, Article 65 (July 2019), 14 pages.

Julio Marco, Quercus Hernandez, Adolfo Muñoz, Yue Dong, Adrian Jarabo, Min H Kim, Xin Tong, and Diego Gutierrez. 2017. DeepToF: off-the-shelf real-time correction of multipath interference in time-of-flight imaging. *ACM Transactions on Graphics (ToG)* 36, 6 (2017), 1–12.

Christopher A. Metzler, Felix Heide, Prasana Rangarajan, Muralidhar Madabhushi Balaji, Aparna Viswanath, Ashok Veeraraghavan, and Richard G. Baraniuk. 2020. Deep-inverse correlography: towards real-time high-resolution non-line-of-sight imaging. *Optica* 7, 1 (Jan 2020), 63–71. https://doi.org/10.1364/OPTICA.374026

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. arXiv:cs.CV/2003.08934

N. Naik, S. Zhao, A. Velten, R. Raskar, and K. Bala. 2011. Single view reflectance capture using multiplexed scattering and time-of-flight imaging. *ACM Trans. Graph.* 30, 6 (2011), 171.

Frédéric Nolet, Samuel Parent, Nicolas Roy, Marc-Olivier Mercier, Serge Charlebois, Réjean Fontaine, and Jean-Francois Pratte. 2018. Quenching Circuit and SPAD Integrated in CMOS 65 nm with 7.8 ps FWHM Single Photon Timing Resolution. *Instruments* 2, 4 (2018), 19.

Kyle Olszewski, Sergey Tulyakov, Oliver Woodford, Hao Li, and Linjie Luo. 2019. Transformable Bottleneck Networks. *The IEEE International Conference on Computer Vision (ICCV)* (Nov 2019).

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

Matthew O'Toole, David B. Lindell, and Gordon Wetzstein. 2018b. Confocal Non-line-of-sight imaging based on the light cone transform. *Nature* (2018), 338–341. Issue 555.

R. Pandharkar, A. Velten, A. Bardagjy, E. Lawson, M. Bawendi, and R. Raskar. 2011. Estimating motion and size of moving non-line-of-sight objects in cluttered environments. In *Proc. CVPR*. 265–272.

Luca Parmesan, Neale AW Dutton, Neil J Calder, Andrew J Holmes, Lindsay A Grant, and Robert K Henderson. 2014. A 9.8 μm sample and hold time to amplitude converter CMOS SPAD pixel. In *Solid State Device Research Conference (ESSDERC), 2014 44th European*. IEEE, 290–293.

Adithya Pediredla, Ashok Veeraraghavan, and Ioannis Gkioulekas. 2019. Ellipsoidal Path Connections for Time-gated Rendering. *ACM Trans. Graph. (SIGGRAPH)* (2019).

Adithya Kumar Pediredla, Mauro Buttafava, Alberto Tosi, Oliver Cossairt, and Ashok Veeraraghavan. 2017. Reconstructing rooms using photon echoes: A plane based model and reconstruction algorithm for looking around the corner. In *IEEE International Conference on Computational Photography (ICCP)*. IEEE.

Stephan R Richter and Stefan Roth. 2018. Matryoshka networks: Predicting 3d geometry via nested shape layers. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1936–1944.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 234–241.

Charles Saunders, John Murray-Bruce, and Vivek K Goyal. 2019. Computational periscopy with an ordinary digital camera. *Nature* 565, 7740 (2019), 472.

Nicolas Scheiner, Florian Kraus, Fangyin Wei, Buu Phan, Fahim Mannan, Nils Appenrodt, Werner Ritter, Jurgen Dickmann, Klaus Dietmayer, Bernhard Sick, et al. 2020. Seeing Around Street Corners: Non-Line-of-Sight Detection and Tracking In-the-Wild Using Doppler Radar. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2068–2077.

Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Niessner, Gordon Wetzstein, and Michael Zollhöfer. 2019a. DeepVoxels: Learning Persistent 3D Feature Embeddings. In *Proc. CVPR*.

Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. 2019b. Scene Representation Networks: Continuous 3D-Structure-Aware Neural Scene Representations. In *Advances in Neural Information Processing Systems*.

Robert H Stolt. 1978. Migration by Fourier transform. *Geophysics* 43, 1 (1978), 23–48.

Shuochen Su, Felix Heide, Gordon Wetzstein, and Wolfgang Heidrich. 2018. Deep end-to-end time-of-flight imaging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6383–6392.

Matthew Tancik, Guy Satat, and Ramesh Raskar. 2018. Flash Photography for Data-Driven Hidden Scene Recovery. *CoRR* abs/1810.11710 (2018). arXiv:1810.11710 http://arxiv.org/abs/1810.11710

Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. 2015. Single-view to Multi-view: Reconstructing Unseen Views with a Convolutional Network. *CoRR* abs/1511.06702 (2015). arXiv:1511.06702 http://arxiv.org/abs/1511.06702

Chia-Yin Tsai, Kiriakos N Kutulakos, Srinivasa G Narasimhan, and Aswin C Sankaranarayanan. 2017. The geometry of first-returning photons for non-line-of-sight imaging. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*.

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 Conference on Computer Vision and Pattern Recognition*. 1545–1555.

A. Velten, T. Willwacher, O. Gupta, A. Veeraraghavan, M.G. Bawendi, and R. Raskar. 2012. Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging. *Nature Communications* 3 (2012), 745.

A. Velten, D. Wu, A. Jarabo, B. Masia, C. Barsi, C. Joshi, E. Lawson, M. Bawendi, D. Gutierrez, and R. Raskar. 2013. Femto-Photography: Capturing and Visualizing the Propagation of Light. *ACM Trans. Graph.* 32 (2013).

Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. 2018. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7794–7803.

D. Wu, M. O'Toole, A. Velten, A. Agrawal, and R. Raskar. 2012. Decomposing global light transport using time of flight imaging. In *Proc. CVPR*. 366–373.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1912–1920.

Feihu Xu, Gal Shulkind, Christos Thrampoulidis, Jeffrey H. Shapiro, Antonio Torralba, Franco N. C. Wong, and Gregory W. Wornell. 2018. Revealing hidden scenes by photon-efficient occlusion-based opportunistic active imaging. *OSA Opt. Express* 26, 8 (2018), 9945–9962.

Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A. Efros. 2016. View Synthesis by Appearance Flow. *CoRR* abs/1605.03557 (2016). arXiv:1605.03557 http://arxiv.org/abs/1605.03557