Neural Fields for Structured Lighting

Aarrushi Shandilya Benjamin Attal Christian Richardt [∞] James Tompkin [†] Matthew O'Toole
Carnegie Mellon University [∞] Meta Reality Labs Research [†] Brown University

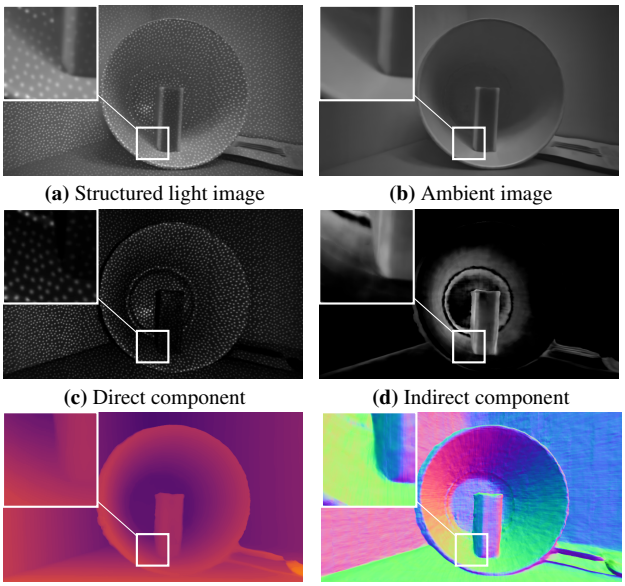
Abstract

*We present an image formation model and optimization procedure that combines the advantages of neural radiance fields and structured light imaging. Existing depth-supervised neural models rely on depth sensors to accurately capture the scene's geometry. However, the depth maps recovered by these sensors can be prone to error, or even fail outright. Instead of depending on the fidelity of processed depth maps from a structured light system, a more principled approach is to explicitly model the raw structured light images themselves. Our proposed approach enables the estimation of high-fidelity depth maps, including for objects with complex material properties (*e.g.*, partially-transparent surfaces). Besides computing depth, the raw structured light images also confer other useful radiometric cues, which enable predicting surface normals and decomposing scene appearance in terms of a direct, indirect, and ambient component. We evaluate our framework quantitatively and qualitatively on a range of real and synthetic scenes, and decompose scenes into their constituent components for novel views.*

1. Introduction

3D scene reconstruction lies at the center of fields like photogrammetry, robotics, and digital preservation. However, reconstructing scenes from 2D image supervision alone is under-constrained and classical approaches struggle in textureless regions [25], where finding correspondences between images is hard. Recent neural rendering techniques like NeRF [18] and other variants [29] are good at novel-view synthesis, but they, too, struggle to reconstruct geometry in scenes with low-texture regions or from few input views.

Many depth cameras alleviate these issues by introducing their own lighting into the scene [9, 24]. For example, active stereo systems (*e.g.*, Intel RealSense [14]) use a projector to illuminate the scene with an (often unknown) light pattern, which adds texture to help solve the stereo correspondence problem. Coded structured light systems use known light patterns to solve correspondences using as few as one camera viewpoint. Such active depth-sensing devices are found in many smartphones and tablets [1, 14, 39], and unlock new VR and AR applications. However, these sensors can also fail to reliably estimate depth, especially in cases where light misbehaves [10, 22], *e.g.*, due to light traveling many different paths before reaching a particular camera pixel.



(a) Structured light image **(b)** Ambient image
(c) Direct component **(d)** Indirect component
(e) Disparity map **(f)** Normal map

Figure 1. **Scene decomposition of a real scene from a novel viewpoint.** Our proposed framework uses the raw measurements from a single infrared camera on an Intel RealSense to generate a volumetric representation of the scene. The images **(a–f)** synthesized from a novel camera viewpoint show the different representations of shape and appearance recovered using our proposed framework.

We propose a volumetric image formation model and corresponding optimization procedure designed to synthesize structured light images under a known projection pattern. Given a set of raw structured light and ambient-only images captured from different viewpoints, our proposed framework reconstructs scenes through a neural volume rendering procedure [18], recovering a representation of a scene's shape and appearance from only a few input views. Beyond recovering the geometry of challenging scenes (*e.g.*, scenes containing translucent objects), our image formation model takes advantage of additional radiometric cues present in the raw structured light images, to solve for normals and

separate images into direct, indirect, and ambient components; see Figure 1. Through a wide range of experiments on real and synthetic datasets, we explore the advantages of our proposed framework, and provide comparisons to both NeRF [18] and depth-supervised NeRF baselines [8].

In summary, we provide the following contributions:

- a physically-based neural volume rendering model for multi-view structured light imaging, incorporating shading cues that inform normals and the separation of direct and indirect components;
- an implementation on a widely-available commercial system, an Intel RealSense camera [14], leading to reliable depth reconstruction performance when compared to baseline approaches and the original RealSense depth;
- a demonstration that our model allows us to tackle new problems with structured light cameras, such as recovering geometry through partially transparent surfaces and through fine meshes.

2. Related Work

Differentiable volume rendering is a reliable approach to reconstructing a digital copy of a scene, enabling the synthesis of images from novel viewpoints and recovering its 3D shape [29]. The approach involves representing geometry and appearance of a scene at every point in space, from which one can render views of the scene by performing numerical integration to approximate a volume rendering integral [18]. Given enough images of a scene, one can optimize a volume representation that takes advantage of the compressive priors induced by a neural network [28, 33]. One current disadvantage is that many different viewpoints are required to accurately reconstruct a scene. This limitation can be addressed using depth consistency priors [21], semantic priors [11], or depth supervision from traditional multi-view stereo algorithms [8, 23, 32]—although in this case, the depth inherits the limitations of traditional passive stereo approaches.

Past works also combine depth supervision from active illumination sensors with neural volume rendering to achieve reconstruction with few images [3, 7, 8, 27, 41]. However, these use limited or simplified image formation models for depth sensing, which do not exploit the potential advantages of neural volume rendering. In contrast, we model the physical image formation process of the raw images from a structured light sensor, and can therefore better take advantage of the benefits of both structured light and volumetric reconstruction. Our approach is similar in spirit to prior works that use flood illumination [4] or time-of-flight sensors [2], though ours focuses on using structured light systems.

In addition to improving reconstruction quality, modeling illumination within a volume rendering image formation model can offer several additional benefits. For example, Bi et al. [4] and Zhang et al. [35] combine neural volume rendering with a flash light source collocated with the camera to recover depth, normals, and scene reflectance. Various works make use of slightly more complicated illumination conditions in the form of point light sources at several different positions [13, 26, 37, 40]. Since these use illumination with limited spatial variation, they rely on large number of captures or light configurations. Other works leverage environment map lighting that is optimized alongside the neural volume [6, 12, 15, 17, 34, 38]. While some works account for global illumination [26, 40], they either limit themselves to two-bounce global illumination or else do not demonstrate direct-global separation in real-world settings [40]. In this work, we show that we can achieve accurate scene reconstruction and intrinsic decomposition (including direct-global separation) on real-world scenes.

Table 1. **Mathematical symbol legend.**

Symbol	Units	Description
$\mathbf{x}, \mathbf{x}_c, \mathbf{x}_p$		A point $\in \mathbb{R}^3$, camera center, projector center.
$\boldsymbol{\omega}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o$		A unit vector $\in \mathbb{R}^3$, incoming direction, outgoing direction.
$\mathbf{n}(\mathbf{x})$		A unit normal $\in \mathbb{R}^3$ perpendicular to a surface at point $\mathbf{x}$.
$L(\mathbf{x}, \boldsymbol{\omega})$	$\mathrm{W}\cdot\mathrm{sr}^{-1}\cdot\mathrm{m}^{-2}$	Radiance measured at point $\mathbf{x}$ in direction $\boldsymbol{\omega}$.
$L_i(\mathbf{x}, \boldsymbol{\omega}_i)$	$\mathrm{W}\cdot\mathrm{sr}^{-1}\cdot\mathrm{m}^{-2}$	Incident radiance to a point $\mathbf{x}$ from a direction $\boldsymbol{\omega}_i$.
$L_o(\mathbf{x}, \boldsymbol{\omega}_o)$	$\mathrm{W}\cdot\mathrm{sr}^{-1}\cdot\mathrm{m}^{-2}$	Outgoing radiance from a point $\mathbf{x}$ in a direction $\boldsymbol{\omega}_o$.
$\sigma(\mathbf{x})$	m^{-1}	Density function at a point.
$T(\mathbf{x}, \mathbf{x}')$	*unitless*	Transmittance function, *i.e.*, accumulated density.
$f(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o)$	sr^{-1}	Bidirectional reflectance distribution function (BRDF).
$f_r(\mathbf{x}, \boldsymbol{\omega}_o)$	sr^{-1}	(Retro-)reflectance function given by $f(\mathbf{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_o)$.

3. Neural Volume Rendering for Structured Light Imaging

Consider using a projector-camera system (*e.g.*, the Intel RealSense [14]) to capture measurements of a scene from multiple viewpoints, where the projection pattern is known. In this scenario, the projector produces stroboscopic illumination, *i.e.*, it is turned *"on"* for even frames and *"off"* for odd frames. When the projector is on, it actively illuminates a scene with the known fixed pattern, and the camera measures the scene's radiometric response to both the projector's illumination and all other light sources in the environment. When the projector is off, the camera only measures the ambient light. Given these measurements, our proposed volume rendering framework reconstructs the depths and normals corresponding to scene geometry, as well as the direct, indirect, and ambient light transport components that contribute to scene appearance (see overview in Figure 2).

To understand how, first consider modeling the light incident at a surface point $\mathbf{x}$ with a function $L_i^{\alpha}(\mathbf{x}, \omega_i)$:

$$L_i^{\alpha}(\mathbf{x}, \omega_i) = \underbrace{L_i^{\text{ambient}}}_{\text{passive}} + \alpha \underbrace{(L_i^{\text{direct}} + L_i^{\text{indirect}})}_{\text{active}}, \qquad (1)$$

where $\alpha \in \{0, 1\}$ accounts for whether the projector is *off* (L_i^0) or *on* (L_i^1). The light from the projector can either take a direct path to an object's surface (L_i^{direct}) or reach the

surface indirectly, by reflecting off of other scene points first (L_i^{indirect}). The ambient term (L_i^{ambient}) represents both direct and indirect light from all sources other than the projector.

Given these incident light sources, the outgoing radiance at the point can be calculated using the rendering equation:

$$L_o^{\alpha}(\mathbf{x}, \boldsymbol{\omega}_o) = \int_{\Omega(\mathbf{x})} f(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) L_i^{\alpha}(\mathbf{x}, \boldsymbol{\omega}_i)(\mathbf{n}(\mathbf{x}) \cdot \boldsymbol{\omega}_i) \, \mathrm{d}\boldsymbol{\omega}_i, \tag{2}$$

where the domain $\Omega(\mathbf{x})$ represents the hemisphere of incident light directions at point $\mathbf{x}$. Here, the bidirectional reflectance distribution function (BRDF), $f(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o)$, defines the proportion of incoming light scattered in the outgoing direction. When combined with Equation 1, we can further decompose the rendering equation as follows:

$$L_o^{\alpha}(\mathbf{x}, \boldsymbol{\omega}_o) = L_o^{\text{ambient}} + \alpha(L_o^{\text{direct}} + L_o^{\text{indirect}}), \tag{3}$$

where each term represents the result of evaluating the integral with respect to the corresponding incident light component. Note that both active components, L_o^{direct} and L_o^{indirect}, depend on the pose of the projector.

Neural Volume Rendering Framework. To model the environment, a neural network $F_{\boldsymbol{\theta}}$ takes as input a 3D position $\mathbf{x}$ and viewing direction $\boldsymbol{\omega}_o$, and outputs terms $(\sigma(\mathbf{x}), \mathbf{n}(\mathbf{x}), f(\mathbf{x}, \boldsymbol{\omega}_o), L_o^{\text{ambient}}(\mathbf{x}, \boldsymbol{\omega}_o), L_o^{\text{indirect}}(\mathbf{x}, \boldsymbol{\omega}_o))$ that capture the scene's geometric and radiometric properties. This neural representation can be used to render scenes from different viewpoints by tracing rays through the volume and computing the radiometric response at each point sampled along the ray [18]. Given a camera's pose and intrinsic parameters, rays are cast from the camera's optical center $\mathbf{x}_c$ through each pixel in direction $\boldsymbol{\omega}_o$. Volume density, normals, reflectance, and outgoing radiance functions are queried at a set of 3D points along the ray, and radiance is accumulated at the camera pixel by computing the following integral:

$$L^{\alpha}(\mathbf{x}_c, \boldsymbol{\omega}_o) = \int_{t_{\text{n}}}^{t_{\text{f}}} T(\mathbf{x}_c, \mathbf{x}) \sigma(\mathbf{x}) L_o^{\alpha}(\mathbf{x}, \boldsymbol{\omega}_o) \, \mathrm{d}t, \tag{4}$$

where

$$T(\mathbf{x}_c, \mathbf{x}) = \exp\left(- \int_{t_{\text{n}}}^{t} \sigma(\mathbf{x}_c - \boldsymbol{\omega}_o s) \, \mathrm{d}s \right), \tag{5}$$

and $\mathbf{x} = \mathbf{x}_c - \boldsymbol{\omega}_o t$. The transmittance function $T(\mathbf{x}_c, \mathbf{x})$ represents the proportion of light that travels from $\mathbf{x}$ to $\mathbf{x}_c$.

As discussed in Equation 3, in this work, we independently model three different components of illumination: (i) the neural network directly outputs the ambient term, as done in previous works [18]; (ii) we provide a physics-based model for the direct term, and (iii) we propose to optimize a term that approximates the indirect component. We focus on the latter two items in the remainder of this section.

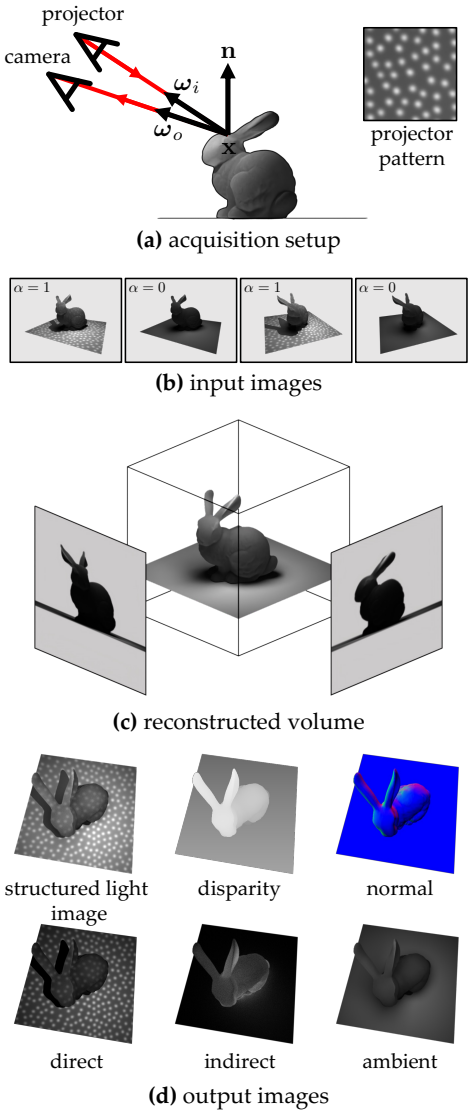


Figure 2. **Overview of structured light reconstruction procedure. (a)** The acquisition setup consists of a single camera and projector illuminating the scene with a fixed projection pattern. **(b)** The projector strobes the illumination as the setup moves around the scene, producing an image sequence where the pattern alternates between on and off. **(c)** The proposed volume reconstruction problem recovers the appearance and shape of the scene. **(d)** The constituent components that make up appearance and shape can then be synthesized for novel views.

The Direct Component. L_o^{direct} models the single-bounce light transport component, where light travels from the projector to a scene point and back to a camera. For a projector at position $\mathbf{x}_p$, the direct lighting incident at a particular point $\mathbf{x}$ is given by the following function:

$$L_i^{\text{direct}}(\mathbf{x}, \boldsymbol{\omega}_i) = \frac{P(\mathbf{x})}{\|\mathbf{x}_p - \mathbf{x}\|^2} T(\mathbf{x}_p, \mathbf{x}) \delta(\boldsymbol{\omega}_p - \boldsymbol{\omega}_i), \tag{6}$$

where

$$\boldsymbol{\omega}_p = \frac{\mathbf{x}_p - \mathbf{x}}{\|\mathbf{x}_p - \mathbf{x}\|}. \tag{7}$$

Function $P(\mathbf{x})$ queries the intensity of the projector pixel illuminating point $\mathbf{x}$, identified through perspective projection, as the projector has similar geometric properties to a camera; note that the output depends on the pose of the projector. The $1/\|\mathbf{x}_p - \mathbf{x}\|^2$ term models the inverse square light fall-off; because the projected area grows proportionally to squared distance, intensity per unit area follows an inverse square falloff. The transmission function $T(\mathbf{x}_p, \mathbf{x})$ determines the proportion of light transmitted between the projector at $\mathbf{x}_p$ and point along the ray $\mathbf{x}$. Finally, the Dirac distribution $\delta(\cdot)$ ensures that the lighting comes from a single direction based on the projector's position.

When combined with Equation 2, we obtain:

$$L_o^{\text{direct}}(\mathbf{x}, \boldsymbol{\omega}_o) = \frac{f(\mathbf{x}, \boldsymbol{\omega}_p, \boldsymbol{\omega}_o)}{\|\mathbf{x}_p - \mathbf{x}\|^2} P(\mathbf{x}) T(\mathbf{x}_p, \mathbf{x})(\mathbf{n}(\mathbf{x}) \cdot \boldsymbol{\omega}_p). \tag{8}$$

This expression is non-trivial to evaluate for two reasons. First, this requires knowledge of the full BRDF at every point in space. Second, this requires evaluating the projector transmission function, which would be a computational bottleneck in neural volume rendering.

To help, we make two assumptions: (i) the projector light casts no shadows, *i.e.*, $T(\mathbf{x}_p, \mathbf{x}) = 1$;[1] and (ii) the BRDF can be approximated with the reflectance function $f_r(\mathbf{x}, \boldsymbol{\omega}_o) = f(\mathbf{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_o)$, representing the ratio of light reflected in the direction of the illumination source. This holds approximately for small-baseline projector-camera systems, provided that the distance to the scene is sufficiently large.

When combined with Equation 4, the expression for the contribution of direct light is given as follows:

$$\int_{t_n}^{t_f} \frac{T(\mathbf{x}_c, \mathbf{x})}{\|\mathbf{x}_p - \mathbf{x}\|^2} \sigma(\mathbf{x}) f_r(\mathbf{x}, \boldsymbol{\omega}_o) P(\mathbf{x})(\mathbf{n}(\mathbf{x}) \cdot \boldsymbol{\omega}_p)\, \mathrm{d}t. \tag{9}$$

The Indirect Component. As a byproduct of our framework, it is possible to recover an approximation of the indirect component for a scene. L_o^{indirect} models the component of light that misbehaves (*e.g.*, bounces around a scene multiple times). However, the global nature of the indirect channel makes it non-trivial to model accurately. This is because, in a volume rendering framework, the indirect component at any given 3D point $\mathbf{x}$ would also depend on 6D pose of the projector-camera system—making it far too challenging to model and reconstruct explicitly.

Our ability to separately recover the direct and indirect components of a scene is based on the work by Nayar et al. [20]. The key idea is to illuminate a scene with a high-frequency pattern and observe the response at a point $\mathbf{x}$ to *different* illumination conditions, *e.g.*, the result of moving the structured light pattern across the scene. Provided that the indirect component is smooth relative to this illumination pattern, the indirect component at a point in the scene stays more or less constant with respect to small perturbations to the global position of the structured light pattern. Therefore, to approximate the indirect component, we propose using a function $L_o^{\text{indirect}}(\mathbf{x}, \boldsymbol{\omega}_o)$ that *only* takes as input the scene point $\mathbf{x}$ and viewing direction $\boldsymbol{\omega}_o$.

In simpler terms, the indirect channel absorbs light contributions that cannot be modelled via direct reflections (Equation 9). Consider a scenario where the projected pattern $P(\mathbf{x})$ is a Dirac delta function, *i.e.*, it is non-zero at only one point. Without indirect light, most of the scene would be black. With indirect light, however, a non-zero contribution of light can potentially reach any camera pixel by bouncing around the scene multiple times. Since the direct reflection model has no way of producing non-zero values in these regions using Equation 9, this forces the model to produce these non-zero responses through the indirect channel.

4. Neural Volume Optimization

We optimize the neural volume framework using both ambient-only measurements $\hat{L}^0(\mathbf{x}_c, \omega_o)$ and structured light measurements $\hat{L}^1(\mathbf{x}_c, \omega_o)$ using the camera-projector poses estimated with COLMAP [25].

We build our framework on top of NeRF-PyTorch [16], which we extend to output normal $\mathbf{n}(\mathbf{x})$, reflectance $f(\mathbf{x}, \boldsymbol{\omega}_o)$, and indirect radiance $L_o^{\text{indirect}}(\mathbf{x}, \boldsymbol{\omega}_o)$. We train coarse and fine networks using an ambient photometric loss:

$$\mathcal{L}^{\text{ambient}} = \left\| L^0(\mathbf{x}_c, \boldsymbol{\omega}_o) - \hat{L}^0(\mathbf{x}_c, \boldsymbol{\omega}_o) \right\|^2, \tag{10}$$

and a structured light photometric loss:

$$\mathcal{L}^{\text{SL}} = \left\| L^1(\mathbf{x}_c, \boldsymbol{\omega}_o) - \hat{L}^1(\mathbf{x}_c, \boldsymbol{\omega}_o) \right\|^2, \tag{11}$$

where L^0 and L^1 are the predicted ambient ($\alpha = 0$) and structured light ($\alpha = 1$) terms from Equation 4. Because the contribution of indirect radiance is often weaker than the other components, we scale the network output corresponding to indirect radiance by 0.1 to implicitly bias the resultant indirect radiance towards a small value. This is an empirically determined hyperparameter value that works well for our experiments. For scenes with minimal indirect component, we simply omit this channel.

We follow the approach proposed by Verbin et al. [30] to tie predicted normals to the gradient density normals:

$$\mathcal{L}_p^{\text{normal}} = \frac{1}{K} \sum\nolimits_{k=1}^{K} w_k \left\| \mathbf{n}_k - \hat{\mathbf{n}}_k \right\|^2, \tag{12}$$

[1] An alternative option is to square the transmission function in Equation 9, to model attenuation of both incident and outgoing light. In practice, we found that this change does not impact reconstruction results however.

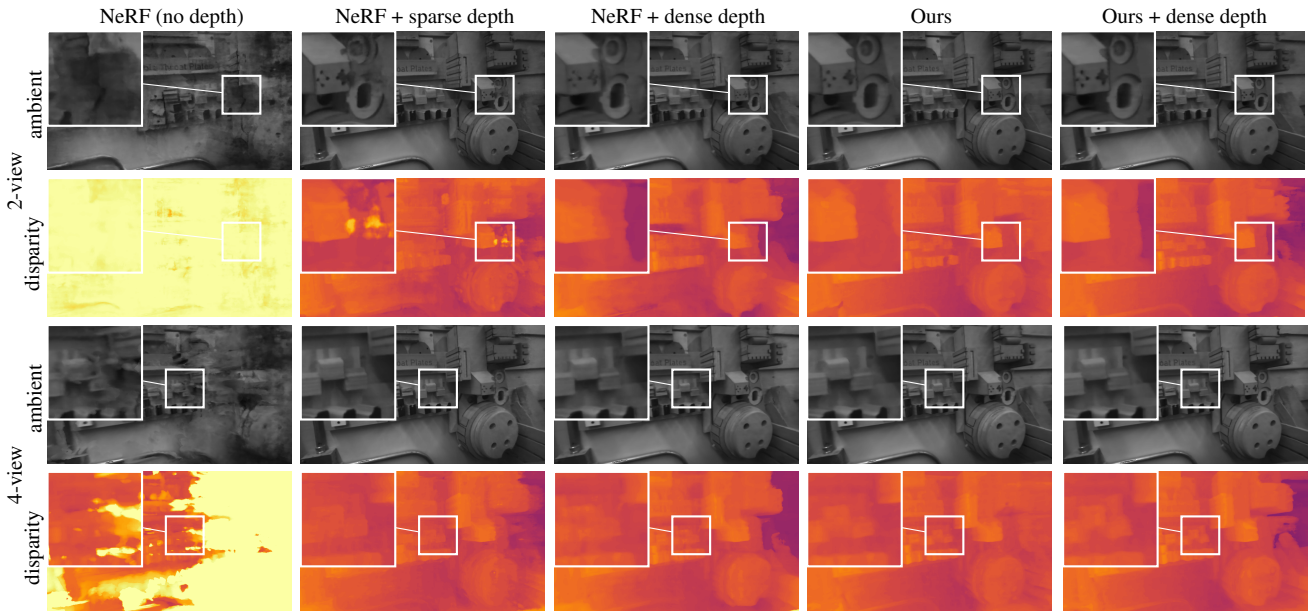


Figure 3. **Novel view reconstruction of ambient image and disparity map for the *woodshop* scene, trained with 2 and 4 views.**

where $\mathbf{n}_k$ are the predicted normals, $\hat{\mathbf{n}}_k = {}^{-\nabla\sigma(\mathbf{x})}/_{||\sigma(\mathbf{x})||}$ are the analytical normals, w_k are weights associated with each sample along a ray, and K represents the number of samples per ray. We also use their proposed penalty term for back-facing normals:

$$\mathcal{L}_o^{\text{normal}} = \frac{1}{K}\sum_{k=1}^{K} w_k \cdot \min(0, \mathbf{n}_k \cdot \boldsymbol{\omega}_o)^2. \tag{13}$$

The total loss for each ray $(\mathbf{x}_c, \boldsymbol{\omega}_o)$ is as follows:

$$\mathcal{L}^{\text{total}} = (1-\alpha)\mathcal{L}^{\text{ambient}} + \lambda_1 \alpha \mathcal{L}^{\text{SL}} + \lambda_2 \mathcal{L}_p^{\text{normal}} + \lambda_3 \mathcal{L}_o^{\text{normal}}, \tag{14}$$

where the value of $\alpha \in \{0, 1\}$ reflects the state of the projector (*off* or *on*) when capturing the ray. During training, we gradually decay the weight λ_1 to a small value. This ensures that the optimization procedure can initially make use of the structured light images to recover geometry (especially of low-texture regions), while the ambient loss dominates, allowing for better detail reconstruction during later iterations. In practice, we alternate between optimizing the $\mathcal{L}^{\text{ambient}}$ and $\mathcal{L}^{\text{SL}}$ objectives, as the poses (and thus rays) for the structured light and ambient images are different. For all our experiments, we train our model for 100K iterations and use the same learning rate decay and optimizer as in NeRF [18]. Training with our method takes 4–6 hours on an NVIDIA 3090 RTX GPU (24 GB RAM).

5. Experiments and Results

5.1. Data Generation

We use an Intel RealSense D435 system [14] with a built-in infrared dot projector to capture real data. Although the device has three cameras, we choose to use only one monochromatic camera for our experiments, but our framework can easily be extended to account for all three cameras. During our calibration procedure, we compute the intrinsics and extrinsics of the projector and camera, and an image representing the dot pattern emitted by the projector. See the supplemental document for details on the calibration process.

While streaming data, the device strobes the illumination to capture a set of frames when the projector is *on*, and another set when *off*. We use COLMAP [25] to obtain poses for the ambient images. To account for the drift between the projector *on* and *off* images, we additionally optimize the structured light image poses as a pre-processing step for each scene and use these calibrated poses for all our experiments—see our supplemental document for more details.

All synthetic scenes are rendered using Blender [5].

5.2. Reconstruction from Sparse Views

For real scenes, we test the novel-view reconstruction and disparity map of our method trained with 2, 4 and 8 views. Since the goal is to perform accurate novel-view synthesis for the ambient images, we decay the structured lighting objective's weight λ_1 from 1 to 0.1 over the first 40K iterations for the 2-view case, and to 0.001 for the 4- and 8-view cases. The large value for λ_1 at the beginning of training helps recover more accurate geometry and prevents over-fitting, and a smaller value for λ_1 at the end helps prioritize image quality (*e.g.*, the ambient term). We compare our model's performance with the following baseline approaches:

NeRF (no depth supervision). We train NeRF [18] with the photometric loss on ambient images only (Equation 10).

NeRF + sparse depth. We train DS-NeRF [8], which includes a depth supervision loss for its fine network with a

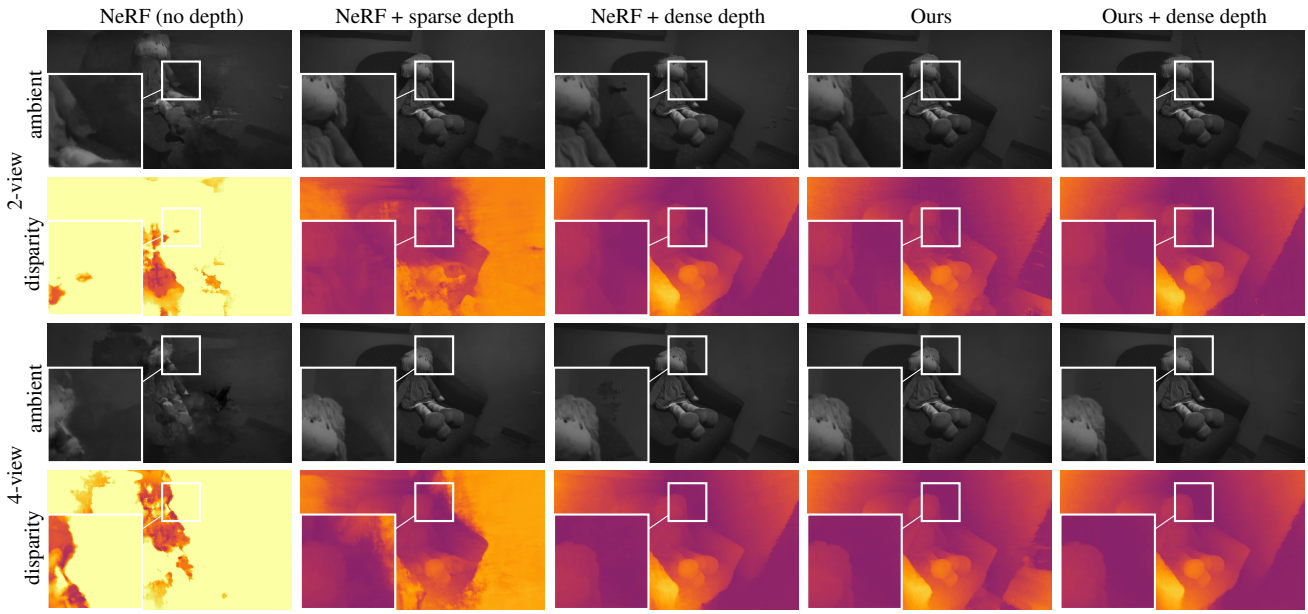


Figure 4. **Novel view reconstruction of ambient image and disparity map for the *doll* scene, trained with 2 and 4 views.**

weight of 0.1 using COLMAP's sparse point cloud.

NeRF + dense depth. Similar to DS-NeRF [8], we add a depth loss to the fine network of NeRF, which uses dense RealSense depth maps for supervision. We mask out the unresolved regions in the RealSense depth maps to avoid supervising with an unreliable signal. Since the RealSense's active depth sensing comes into play when the projector is on, we add depth supervision for only these poses. The initial depth loss weight is set to 0.1, and we decay it at the same rate as λ_1 in our method to 0.01 for 2 views, and 0.0001 for 4 and 8 views.

Ours + dense depth. We combine our approach with the RealSense depth supervision, using the same loss parameters for the structured light and depth losses as described above.

We perform this comparative study on 50 to 100 held-out views each for four real scenes: *woodshop*, *doll*, *sculpture*, and *translucent box*. For all the cases, we train using a batch size of 2048, 32 uniform samples, and 64 importance samples for 100K iterations. For the *translucent box* scene, we use 64 uniform samples for all methods. For these comparisons, we omit both the prediction of normals and the indirect component from our image formation model.

We report the quantitative analysis for novel-view synthesis in Table 2 using PSNR, SSIM [31], and LPIPS [36] metrics. We provide a per-scene breakdown of metrics in the supplement. Figure 3 and Figure 4 present qualitative results for the 2-view and 4-view cases.

NeRF recovers cloudy geometry in textureless regions and is unable to interpolate well with sparse views. It compensates for inaccurate geometry with spurious appearance estimates to overfit the integrated radiance value for training images, which does not scale to test views. Quantitatively, all the depth-based methods (sparse depth, dense depth, and ours) produce higher-quality representations of the scene than NeRF, as expected. Qualitatively, however, it is clear that both NeRF and sparse depth supervision struggle to capture the scene geometry, as shown in Figure 3 and Figure 4. Provided the scenes contain relatively simple geometry such that the RealSense depth is reliable, both dense depth supervision and our proposed method produce comparable results.

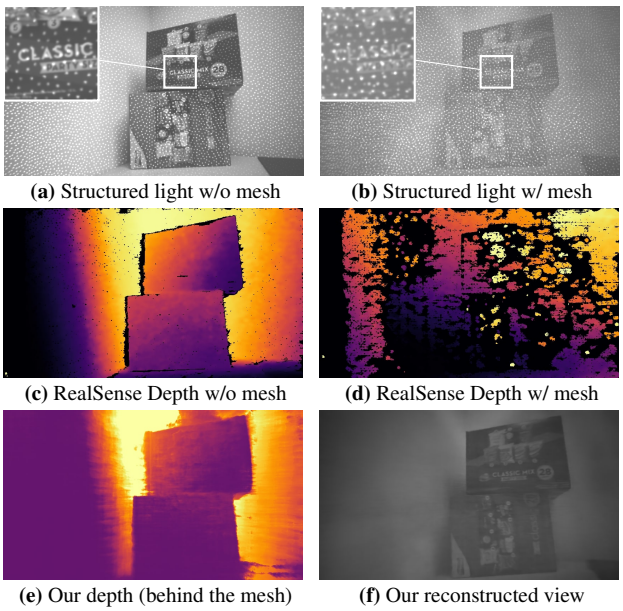
Table 2. **Novel-view synthesis quality is improved by depth supervision, especially in few-view cases.** Quantitative analysis on real dataset. '2-v' denotes two views.

	PSNR ▲			SSIM ▲			LPIPS ▼		
Method	2-v	4-v	8-v	2-v	4-v	8-v	2-v	4-v	8-v
NeRF	27.53	33.40	38.31	0.886	0.936	0.965	0.399	0.307	0.231
+ sparse depth	**36.74**	**41.22**	**42.38**	0.969	**0.988**	**0.990**	0.214	**0.158**	**0.148**
+ dense depth	36.61	40.74	42.06	0.971	0.986	0.988	0.207	0.173	0.165
Ours	34.89	40.97	42.02	0.959	0.986	0.989	0.218	0.167	0.163
+ dense depth	36.70	40.89	42.35	**0.973**	0.987	0.989	**0.196**	0.166	0.159

5.3. Reconstructing Translucent Objects

There are scenarios where structured light sensors completely fail to capture depth, however. For example, when imaging translucent objects, multiple depth planes contribute light to a sensor (Figure 5b). As a result, recovering a single depth map for such scenes is fundamentally ill-defined.

We perform a qualitative comparison of our approach to the depth from Intel RealSense on scenes containing partially transparent objects. In particular, we capture an additional *mesh* scene containing a plastic mesh placed in front of a



(a) Structured light w/o mesh **(b)** Structured light w/ mesh

(c) RealSense Depth w/o mesh **(d)** RealSense Depth w/ mesh

(e) Our depth (behind the mesh) **(f)** Our reconstructed view

Figure 5. **Reconstructing multiple surfaces for the *mesh* scene.** **(a–b)** Structured light images captured by the Intel RealSense without and with a large plastic mesh placed in between the camera and the scene. The partially-transparent mesh obscures the scene and effectively creates a second "copy" of the structured light pattern, leading to ambiguous correspondences (see insets). **(c–d)** We show the RealSense depth for the scene without and with the mesh. Note that the RealSense completely fails to predict reasonable depth in the latter case. **(e–f)** Novel view recovered from 128 images, as well as a depth map produced by filtering out the geometry of the mesh. Our method accurately reconstructs the geometry of the background, while the RealSense fails to do so.

table containing a stack of boxes. In Figure 5, we show that the Intel RealSense fails to accurately capture the scene's geometry due to the ambiguous correspondences caused by multiple direct reflections. In contrast, our approach has the ability to model the contribution of direct illumination from *multiple points* along the path of a ray, allowing us to capture both the geometry of the mesh and the objects behind it.

We further provide a qualitative comparison of our approach in Figure 6 on the *translucent box* scene, consisting of a plastic box filled with various balls. Here, we visualize the rendering weights w_k from Section 4 along a ray passing through the plastic box, demonstrating that our method recovers the geometry of all surfaces (translucent and opaque) along the path of a ray.

5.4. Predicting Normals

A unique advantage of working with the raw structured light images (instead of processed depth maps) is that it provides shading cues that can be used to predict surface normals. In this section, we provide quantitative and qualitative assessments of our proposed method in simulation.

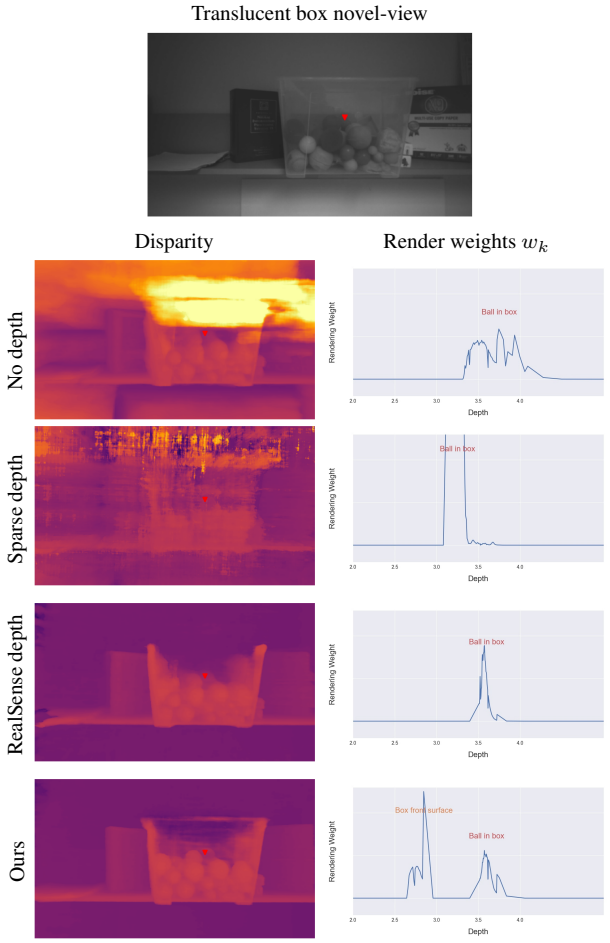


Figure 6. **Reconstructing multiple surfaces for the *translucent box* scene.** In this example, we show that our method recovers the geometry of a partially-transparent plastic container from 110 images, while other methods fail to do so. The red marker indicates in the pixel (and hence, the ray) chosen for rendering weights visualization. Our method recovers the depth of both front surface of the container (first peak), as well as a ball placed within the container (second peak), whereas other methods fail to detect or distinguish the front surface of the container.

We test the reconstruction of novel views, disparity, and normal maps for our method and compare them with the following: (i) the base NeRF model which does not incorporate any illumination information, and (ii) our model where we replace the structured light (dot) pattern with flood illumination (mimicking the setup by Bi et al. [4]). For both the dot pattern and flood illumination, we decay the structured light loss weight λ_1 from 1 to 0.05 over the first 40K iterations, and set the normal prediction and normal orientation loss weights to $\lambda_2 = 3 \times 10^{-4}$ and $\lambda_3 = 0.1$, respectively. All methods are trained for 100K iterations with a 1024 batch size, 64 uniform samples and 128 importance samples. We train using 25 views and test on 50 held-out views for 4 scenes. We omit the indirect component of our image forma-

tion model here to focus on normal recovery.

For quantitative analysis in Table 3, we compute PSNR, SSIM [31] and LPIPS [36] on the reconstructed images, MSE (mean squared error) on the depth map, and MAE (mean angular error in degrees) for the analytical normal maps. Figure 7 shows the qualitative analysis for two of the scenes. Both quantitatively and qualitatively, the use of a structured light dot pattern significantly outperforms the case of single intensity or no active illumination in terms of normal and depth fidelity.

Explicitly predicting normals produces smoother results compared to the noisy analytical gradients computed from the density function. As shown in Figure 8, explicitly predicting normals, modeling them as the cosine shading term, and penalizing them as in Equation 12 helps improve the quality of analytical normals, making them less noisy and capturing more detail.

Table 3. **Adding structured light significantly reduces depth and normal error while maintaining novel-view synthesis quality.** 'Flood light' and 'structured light' refer to projecting a dot and single intensity pattern onto the scene, respectively.

Method	PSNR ▲	SSIM ▲	LPIPS ▼	Depth MSE ▼	Normals MAE °▼
Ambient-only (NeRF)	**44.98**	**0.985**	**0.368**	0.615	24.34
+ Flood light	43.51	0.982	0.375	0.786	8.76
+ Structured light (Ours)	44.13	0.984	0.370	**0.013**	**2.84**

5.5. Decomposing Scene Appearance

Finally, we showcase the effectiveness of modeling the direct and indirect radiance, and produce a complete decomposition of all components using our framework, including the ambient, direct, and indirect components for scene appearance, and the disparity and normals associated with scene geometry. We first construct a synthetic scene with a frog object, whose skin exhibits subsurface scattering. The frog is positioned close to the intersection of two planes, which also introduces diffuse inter-reflections. Using the same hyperparameters as described in Section 5.4, we show the ability to synthesize scenes from a novel viewpoint, and decompose the scene into its constituent components in Figure 9.

To demonstrate this in practice, we capture a real scene in Figure 1, consisting of a translucent candle placed within a large concave object. This scene exhibits strong subsurface scattering and inter-reflections. We train our model using 8 views, a 2048 batch size, 32 uniform samples, and 64 importance samples for 100K iterations. The structured light weight λ_1 is decayed to 0.1 over 40K iterations, and the normal prediction and orientation loss weights are set to $\lambda_2 = 0.001$ and $\lambda_3 = 0.1$, respectively. The result is a decomposition of all components of the scene, accurately capturing the presence of indirect light within this scene. Another example of a real scene decomposition into its shape and appearance components is shown in Figure 10. Trained using only 5 views, our method recovers the indirect component corresponding to the light scattering in a milk medium and other inter-reflections in the scene for a novel view.

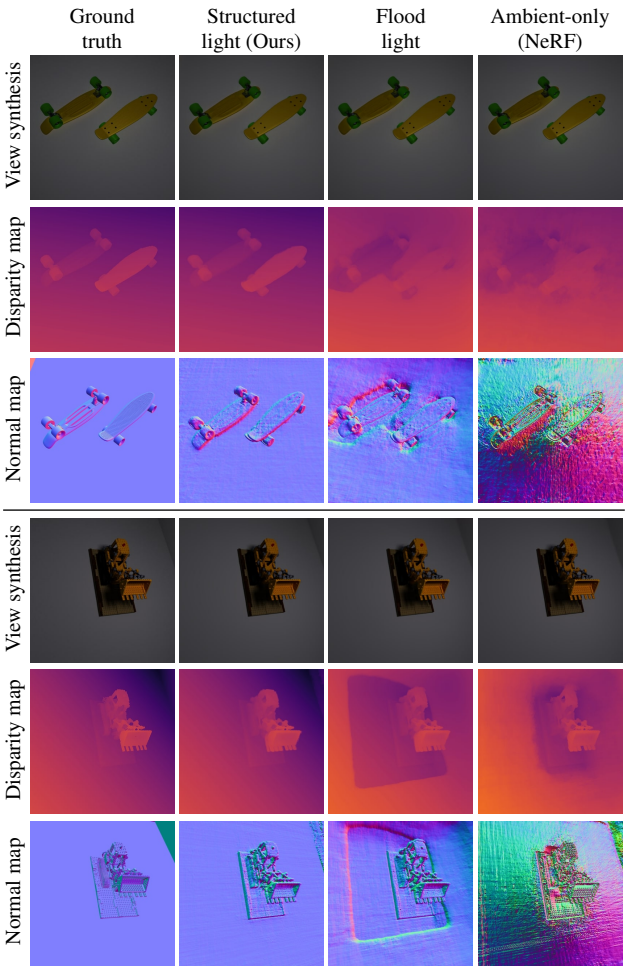


Figure 7. **Qualitative comparison of ambient novel-view synthesis, disparity and analytical normal maps for synthetic data.**

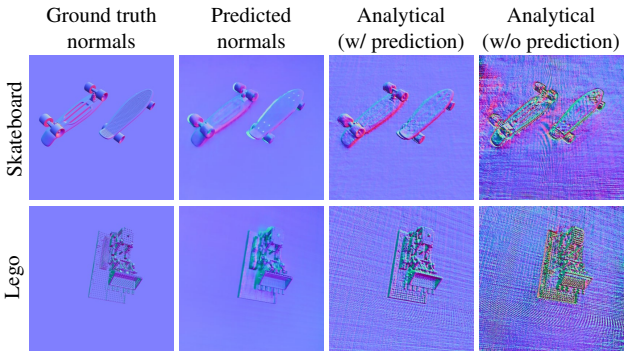


Figure 8. **Impact of explicitly predicting and modeling normals in the proposed method.** Novel-view normal map visualization for the *lego* and *skateboard* scenes.

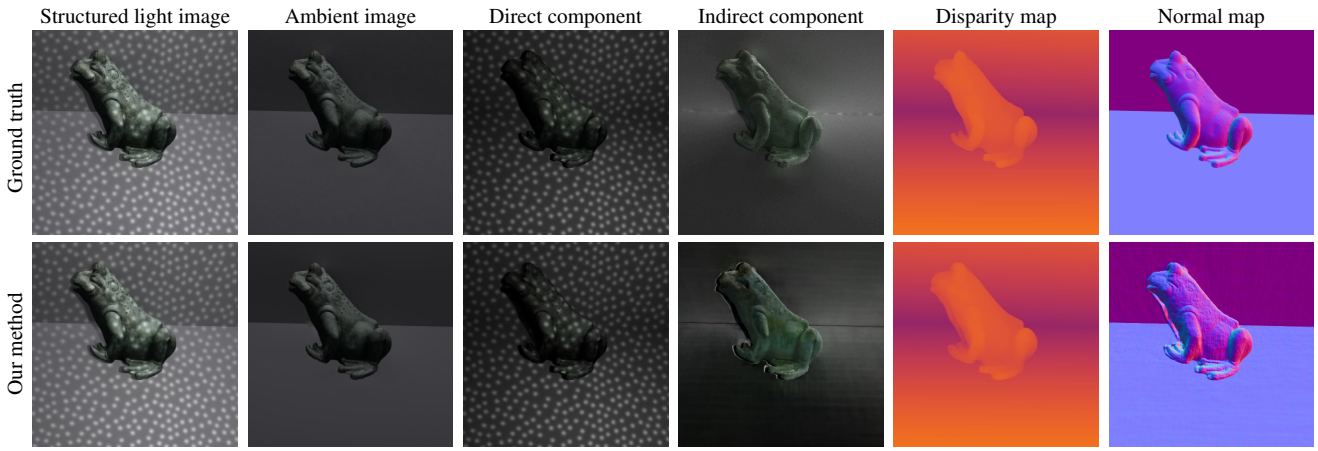
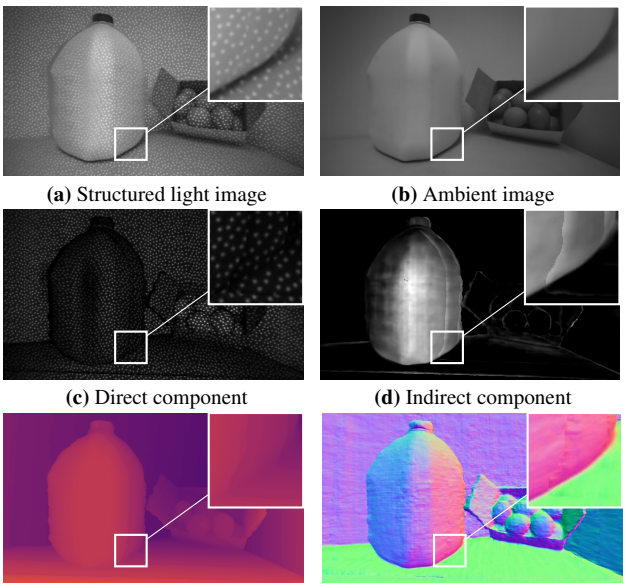


Figure 9. **Novel-view scene decomposition on a synthetic scene.** Our method enables the synthesis of the ambient, direct, and indirect appearance along with accurate estimation of the normals and disparity map for novel views, by physically-based modeling of scenes exhibiting complex light interactions like subsurface scattering and inter-reflections.



(a) Structured light image **(b)** Ambient image

(c) Direct component **(d)** Indirect component

(e) Disparity map **(f)** Normal map

Figure 10. **Novel-view scene decomposition on a real scene, containing a carton of eggs and a jug of milk.** Our method recovers the appearance and shape components for novel views of a scene containing a scattering medium (*i.e.*, the milk).

6. Discussion

Limitations. As with existing structured light systems, our method can be confused by objects that produce complex light transport effects. For example, recovering scene geometry in the presence of mirrors and refractive objects is a notoriously difficult problem [10, 22]. When imaging outdoors under bright ambient lighting or imaging objects placed far away, the illumination from the projector may be too weak to detect. In such cases, however, our proposed method is expected to fail gracefully and have similar performance to NeRF [18], because the ambient photometric loss dominates the structured light photometric loss.

While the overall quality of novel-view synthesis of our method is similar to other methods, our framework is unable to accurately disambiguate geometry for edges near projector shadows. In principle though, it may be possible to (i) identify shadow pixels in each measurement and (ii) disregard these pixels during the training process. Optical techniques can also be employed to further mitigate the effect of shadows from images [19].

Concluding Remarks. In this paper, we proposed a neural volume rendering framework for multi-view structured lighting. This framework recovers accurate geometry and synthesizes novel views by modeling the image formation process for a commodity Intel RealSense structured light system. We demonstrated that our depth-based framework provides a more principled approach to recovering scene geometry, enabling it to account for challenging scenes that contain partially transparent objects. Moreover, our ability to model the raw structured light images further enables our method to recover accurate surface normals, and to separate direct and indirect components. Looking forward, we believe that this framework can be extended to make use of all three cameras on RealSense devices, and potentially even support scanning scenes with multiple such devices in tandem.

7. Acknowledgements

We acknowledge support from NSF IIS 2008464 and a Meta gift. Benjamin Attal is supported by a Meta Research PhD Fellowship in AR/VR Computer Graphics. James Tompkin was supported by an NSF CAREER award (IIS 2144956) and Cognex. Matthew O'Toole acknowledges support from an NSF CAREER award (IIS 2238485).

References

[1] Apple Inc. Face ID advanced technology. https://support.apple.com/en-us/HT208108, 2022. 1

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

[3] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural RGB-D surface reconstruction. In *CVPR*, 2022. 2

[4] Sai Bi, Zexiang Xu, Pratul Srinivasan, Ben Mildenhall, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. Neural reflectance fields for appearance acquisition. arXiv:2008.03824, 2020. 2, 7

[5] Blender Online Community. *Blender – a 3D modeling and rendering package*. Blender Foundation, 2023. 5

[6] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T Barron, Ce Liu, and Hendrik Lensch. NeRD: Neural reflectance decomposition from image collections. In *ICCV*, 2021. 2

[7] Ilya Chugunov, Yuxuan Zhang, Zhihao Xia, Xuaner Zhang, Jiawen Chen, and Felix Heide. The implicit values of a good hand shake: Handheld multi-frame neural depth refinement. In *CVPR*, 2022. 2

[8] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised NeRF: Fewer views and faster training for free. In *CVPR*, 2022. 2, 5, 6

[9] Jason Geng. Structured-light 3D surface imaging: a tutorial. *Adv. Opt. Photon.*, 3(2):128–160, 2011. 1

[10] Mohit Gupta, Amit Agrawal, Ashok Veeraraghavan, and Srinivasa G Narasimhan. Structured light 3D scanning in the presence of global illumination. In *CVPR*, 2011. 1, 9

[11] Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting NeRF on a diet: Semantically consistent few-shot view synthesis. In *ICCV*, 2021. 2

[12] Berk Kaya, Suryansh Kumar, Carlos Oliveira, Vittorio Ferrari, and Luc Van Gool. Uncalibrated neural inverse rendering for photometric stereo of general surfaces. In *CVPR*, 2021. 2

[13] Berk Kaya, Suryansh Kumar, Francesco Sarno, Vittorio Ferrari, and Luc Van Gool. Neural radiance fields approach to deep multi-view photometric stereo. In *WACV*, 2022. 2

[14] Leonid Keselman, John Iselin Woodfill, Anders Grunnet-Jepsen, and Achintya Bhowmik. Intel RealSense stereoscopic depth cameras. In *CVPR Workshops*, 2017. 1, 2, 5

[15] Junxuan Li and Hongdong Li. Self-calibrating photometric stereo by neural inverse rendering. In *ECCV*, 2022. 2

[16] Yen-Chen Lin. NeRF-pytorch. https://github.com/yenchenlin/nerf-pytorch/, 2020. 4

[17] Linjie Lyu, Ayush Tewari, Thomas Leimkühler, Marc Habermann, and Christian Theobalt. Neural radiance transfer fields for relightable novel-view synthesis with global illumination. In *ECCV*, 2022. 2

[18] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2, 3, 5, 9

[19] Shree K Nayar and Mohit Gupta. Diffuse structured light. In *ICCP*, 2012. 9

[20] Shree K Nayar, Gurunandan Krishnan, Michael D Grossberg, and Ramesh Raskar. Fast separation of direct and global components of a scene using high frequency illumination. In *SIGGRAPH*, pages 935–944, 2006. 4

[21] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. RegNeRF: Regularizing neural radiance fields for view synthesis from sparse inputs. In *CVPR*, 2022. 2

[22] Matthew O'Toole, John Mather, and Kiriakos N Kutulakos. 3D shape and indirect appearance by structured light transport. In *CVPR*, 2014. 1, 9

[23] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *CVPR*, 2022. 2

[24] Joaquim Salvi, Jordi Pagès, and Joan Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827–849, 2004. 1

[25] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1, 4, 5

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

[27] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J Davison. iMAP: Implicit mapping and positioning in real-time. In *ICCV*, 2021. 2

[28] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *NeurIPS*, 2020. 2

[29] Ayush Tewari, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, Yifan Wang, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, Tomas Simon, Christian Theobalt, Matthias Niessner, Jonathan T. Barron, Gordon Wetzstein, Michael Zollhoefer, and Vladislav Golyanik. Advances in neural rendering. In *Computer Graphics Forum*, pages 703–735, 2022. 1, 2

[30] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *CVPR*, 2022. 4

[31] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. 6, 8

[32] Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. NerfingMVS: Guided optimization of neural radiance fields for indoor multi-view stereo. In *ICCV*, 2021. 2

[33] Yiheng Xie, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tompkin, Vincent Sitzmann, and Srinath Sridhar. Neural fields in visual computing and beyond. *Computer Graphics Forum*, 2022. 2

[34] Wenqi Yang, Guanying Chen, Chaofeng Chen, Zhenfang Chen, and Kwan-Yee K Wong. PS-NeRF: Neural inverse rendering for multi-view photometric stereo. In *ECCV*, 2022. 2

[35] Kai Zhang, Fujun Luan, Zhengqi Li, and Noah Snavely. IRON: Inverse rendering by optimizing neural SDFs and materials from photometric images. In *CVPR*, 2022. 2

[36] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 6, 8

[37] Xiuming Zhang, Sean Fanello, Yun-Ta Tsai, Tiancheng Sun, Tianfan Xue, Rohit Pandey, Sergio Orts-Escolano, Philip Davidson, Christoph Rhemann, Paul Debevec, et al. Neural light transport for relighting and view synthesis. *ACM Transactions on Graphics*, 40(1):1–17, 2021. 2

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

[39] Zhengyou Zhang. Microsoft Kinect sensor and its effect. *IEEE MultiMedia*, 19(2):4–10, 2012. 1

[40] Quan Zheng, Gurprit Singh, and Hans-Peter Seidel. Neural relightable participating media rendering. In *NeurIPS*, 2021. 2

[41] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. NICE-SLAM: Neural implicit scalable encoding for SLAM. In *CVPR*, 2022. 2