Non-line-of-sight Surface Reconstruction
Using the Directional Light-cone Transform

Sean I. Young
Stanford University
sean0@stanford.edu

David B. Lindell
Stanford University
lindell@stanford.edu

Bernd Girod
Stanford University
bgirod@stanford.edu

David Taubman
UNSW Sydney
d.taubman@unsw.edu.au

Gordon Wetzstein
Stanford University
gordon.wetzstein@stanford.edu

Abstract

We propose a joint albedo–normal approach to non-line-of-sight (NLOS) surface reconstruction using the directional light-cone transform (D-LCT). While current NLOS imaging methods reconstruct either the albedo or surface normals of the hidden scene, the two quantities provide complementary information of the scene, so an efficient method to estimate both simultaneously is desirable. We formulate the recovery of the two quantities as a vector deconvolution problem, and solve it using the Cholesky–Wiener decomposition. We show that surfaces fitted non-parametrically using our recovered normals are more accurate than those produced with NLOS surface reconstruction methods recently proposed, and are 1,000× faster to compute than using inverse rendering.

1. Introduction

Non-line-of-sight (NLOS) imaging deals with the capture and rendering of a scene that is not in the direct line of sight from the sensor. In recent years, NLOS imaging has emerged as an important vision problem, with applications in remote sensing, defense, robotic vision and autonomous driving. A common imaging setup is to "look around the corner" using the confocal configuration depicted in Figure 1. Typically, a light source, such as a laser beam, indirectly illuminates the scene by reflecting onto a surface that can be seen from both the sensor and the scene. The sensor then captures the scene reflections from the same surface location, and records them as a time-resolved sequence of two-dimensional images (or transients), from which one can computationally reconstruct the scene. Apart from transient-based imaging, other NLOS imaging modalities include those based on speckle [1–3], or incoherent intensity measurements [4, 5], as well as passive sensing [6–9] and acoustic imaging [10] techniques.

Here, we will consider exclusively NLOS imaging based on transients [11–19], from which the hidden NLOS scene is typically rendered as a spatial, three-dimensional volume of albedo (volumetric albedo), or as a set of object surfaces. In the volumetric albedo paradigm, the objective is to estimate albedo values for scene voxels [11–16], while in the surface
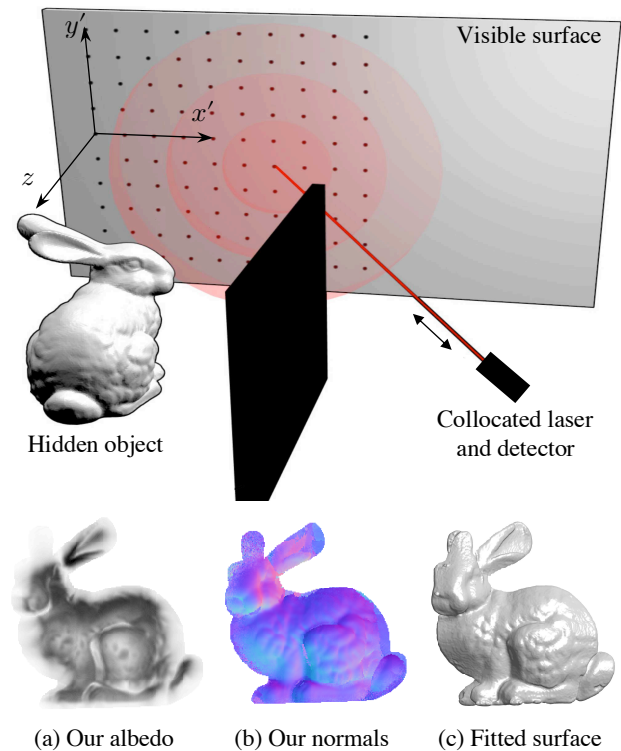

Figure 1. **NLOS surface reconstruction via the D-LCT:** Existing NLOS imaging methods typically recover only the albedo of the hidden scene. The Directional LCT recovers both the albedo (a) and the surface normals (b) of the scene, allowing us to reconstruct the hidden object surface with finer detail (c).

(a) Our albedo   (b) Our normals   (c) Fitted surface

reconstruction paradigm, one seeks more directly to recover object surfaces in the three-dimensional scene by estimating their surface normals [17–19]. While surface-based methods have the potential to reconstruct object geometry with finer detail than the albedo ones, current approaches to estimating surface normals are sensitive to noise, limited to scenes with simpler object geometry [18], sensitive to initialization [19] or entail a high computational complexity [13], all of which motivate our present work. Embedded in transients is a mix of surface normal and albedo information, so that explicitly accounting for the presence of both in transients can present new opportunities for recovering both quantities robustly.

(a) Transient volume τ (b) Normals υ from τ (c) Surface fitted onto υ (d) Ground-truth surface
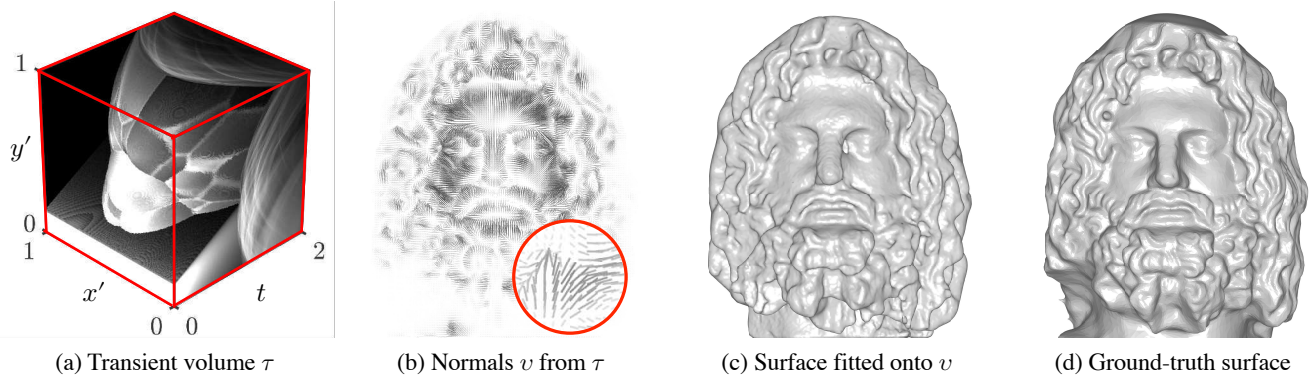
Figure 2. **Method overview:** A $1\text{m} \times 1\text{m} \times 2\text{ns}$ volume τ of transients (a) is filtered using the directional light-cone transform to obtain the surface normals (b). We integrate the surface normals to obtain the final reconstructed surface (c), which is similar to the ground-truth (d).

In this work, we propose a joint albedo–normal approach to NLOS scene reconstruction based on the directional light-cone transform (D-LCT), which is a vectorial generalization of the (scalar) light-cone transform (LCT) recently proposed by O'Toole *et al.* [12]. We formulate the recovery of normals and albedos as a vector deconvolution problem, and having found the two quantities, we fit a surface onto the recovered normals. By solving this vectorial deconvolution problem in the joint albedo–normal space, we recover the two quantities robustly and efficiently, allowing us to obtain a better surface reconstruction. In particular, our approach is many orders of magnitude faster than recent work [13], which can similarly estimate albedos and surface normals in a joint manner. Our proposed approach also works with existing confocal NLOS imaging hardware. We illustrate our overall method pipeline in Figure 2. To summarize, our main contributions are:

- **Directional LCT:** We express non-line-of-sight surface normal recovery as a vector deconvolution problem on time-resolved measurements, via the Directional Light-cone Transform (D-LCT).

- **Cholesky–Wiener Solver:** We solve the above vector-deconvolution problem efficiently in the Fourier domain to recover the surface normals.

- **Surface Fitting:** We reconstruct highly-accurate object surface descriptions by fitting surface parameters on the recovered normals.

2. Related Work

Transient imaging was first conceptualized by Kirmani *et al.* [4], who believed that we can look around the corner by probing a wall with an ultrafast laser and detector. Later, the idea was demonstrated in practice by Velten *et al.* [11] using a femtosecond laser and a streak camera. Owing to the cost of such imaging hardware, researchers have also used other sensing technologies such as time-of-flight cameras [20, 21] or even regular consumer cameras [9, 22]. However, single-

photon avalanche diodes (SPADs) [23–27] have been shown to be particularly versatile for sensing, allowing us to image under ambient lighting, at fast rates [28], or at long distances [29]. Altmann *et al.* [30] provide a comprehensive review of SPADs as well as their applications.

Having captured the transient data, the NLOS scene can be reconstructed as a volume of albedo [11–16], or surfaces of objects [17–19]. Generally speaking, estimation of albedo can be posed as an inverse-filtering problem whereas that of surfaces is often posed as an inverse-rendering problem. We now provide a brief review of the two inverse approaches.

2.1. Inverse Filtering Approaches

Velten *et al.* [11] are the first to pose the recovery of the albedo volume as an inverse problem. Relating the transient measurements to some NLOS scene using higher-order light transport, they formulated scene reconstruction as a (linear) least-squares problem and solved the resulting dense system of equations with filtered back-projection (FBP). While FBP produces promising results, it only approximately solves the original least-squares problem, and the reconstructed scenes can lack fine details. Since the computational complexity of FBP is still high at $O(V^5)$ in the number V^3 of voxels, later authors sought to refine FBP by improving the quality of the iterative solvers [16, 31] or their speed on GPUs [26].

In the confocal case, O'Toole *et al.* [12] note that higher-order light transport can be expressed as a convolution with a change of variables. Their overall transformation, referred to as the light-cone transform (LCT), expresses the problem of Velten *et al.* [11] as a three-dimensional signal deblurring problem. In contrast to the FBP, the LCT solves the inverse problem exactly, and has a low computational complexity of $O(V^3 \log V)$ in the number V^3 of voxels, thanks to the use of the Fourier transform. Recently, Ahn *et al.* [32] proposed an approximate convolutional imaging transform similar to the LCT for the non-confocal setting.

In contrast with deconvolution, which we can ultimately relate to the diffusion equation, Lindell *et al.* [14] suggest to

solve the transient imaging problem by modeling high-order light transport as wave propagation in the three-dimensional space, and solve the resulting inverse problem efficiently in the Fourier domain using "f–k" migration. This method also has a $O(V^3 \log V)$ computational complexity due to the use of the Fourier transform. Methods based on diffractive wave propagation, e.g. phasor fields [33], aim further to overcome the limitations in the imaging model due to assumptions such as single scattering, and the lack of occlusions in the hidden scene. If the confocal setup is used, the phasor field method can be implemented in terms of the LCT for a $O(V^3 \log V)$ computational complexity.

Whereas both the LCT and the f–k migration approaches are extremely efficient, they do not innately have the ability to estimate the surface normals of scene objects. To find the surface normals along with the albedo volume, Heide *et al*. [13] pose the transient imaging problem as an optimization in the albedo and the surface normal variables. Although the reported results are promising, one major limitation of such a method is the $O(V^5)$ complexity in both computation and memory, as well as the nonconvexity of the overall problem formulation. The D-LCT helps us to solve a similar albedo–normal estimation problem with the same low complexity of the LCT and f–k migration. Given the shared Fourier roots across the LCT, f–k and phasor fields, it may be possible to apply our directional approach to f–k migration and phasor fields as well, although we do not attempt this in our work.

2.2. Inverse Rendering Approaches

In contrast with the inverse filtering approaches, inverse rendering (analysis-by-synthesis) methods, e.g. [19], search for values of the NLOS surface parameters (e.g. BRDFs and surface normals) that would produce the observed transients if the NLOS surface were to be rendered. Since a full search in the surface parameter space would be intractable, inverse rendering is typically performed via differentiable rendering (i.e., an energy minimization in surface parameters). Surface parameters recoverable in this way include surface locations and normals [34], and illumination effects such as scattering [35–38] and interreflections [39, 40].

2.3. Surface Fitting Methods

Object surface can be fit on the obtained surface normals either parametrically or non-parametrically. Non-parametric fitting is commonly used for stereo-based 3D reconstruction [41–43] whereas parametric approaches are more suited for mesh refinement [44–47]. Surface fitting may be seen as an inverse problem where the goal is to interpolate a smooth manifold without violating the given normal conditions. We can naturally formulate surface fitting as diffusion processes or energy minimization methods, both of which seek to find the right trade-off between regularity (smoothness) and data fidelity (surface orthogonality to the given normals). We use the energy-minimization approach of [48] to fit a surface on our recovered normals.

3. Mathematical Framework

After briefly reviewing the volumetric albedo model and discussing its limitations, we develop our directional albedo model, and propose efficient ways for solving the associated inverse problem of estimating the surface normals.

3.1. The Volumetric Albedo Model

In transient imaging approaches, a time-resolved detector is used to measure the incident flux of photons as a function of emitted light impulses. Each of these time measurements records the impulse response of the NLOS scene at positions on a visible surface to produce a volume of transients.

Let us denote the three-dimensional scene coordinates by (x, y, z), and assume the visible surface is positioned along $z = 0$. We denote by $(x', y', z = 0)$ positions on this visible surface; see Figure 1. A common transient imaging model is the confocal volumetric albedo model

$$\tau(x', y', t) = \iiint_\Omega \mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z\ \frac{\rho(x, y, z)}{r^4}$$
$$\cdot\, \delta\!\big(2\sqrt{(x'-x)^2 + (y'-y)^2 + z^2} - tc\big), \tag{1}$$

in which ρ denotes a three-dimensional albedo volume with finite support Ω, and $\delta(\cdot)$ relates the round-trip time of flight of light with twice the distance r between the scene (x, y, z) and the sensing $(x', y', z = 0)$ locations. Here, $c \approx 3 \times 10^8$ denotes the speed of light while $1/r^4 = (2/tc)^4$ models the radiometric fall-off due to distance. The scaling $1/r^4$ can be removed from (1) if we prescale τ by $(2/tc)^4$ in advance. In the case of retro-reflective surfaces, a fall-off factor of $1/r^2$ is more commonly assumed.

To discretize model (1), we sample Ω using N, N and M points on the x-, y- and z-axes, respectively. Assuming that the transient τ has been pre-scaled by $(2/tc)^4$, we can write the discretized model compactly using matrix notation as

$$\boldsymbol{\tau} = \mathbf{K}\boldsymbol{\rho}, \tag{2}$$

in which $\boldsymbol{\tau}, \boldsymbol{\rho} \in \mathbb{R}^{N^2 M}$ and $\mathbf{K}$ is a binary matrix with values obtained by sampling $\delta(\cdot)$. Since $\mathbf{K}$ is a low-pass operator of high condition number, the task of finding $\boldsymbol{\rho}$ from given $\boldsymbol{\tau}$ is an ill-posed problem [49]. Rather than compute the solution directly as $\boldsymbol{\rho}^{\mathrm{opt}} = \mathbf{K}^{-1}\boldsymbol{\tau}$, we should find it as the solution of the regularized least-squares problem

$$\text{minimize } f(\boldsymbol{\rho}) = \|\mathbf{K}\boldsymbol{\rho} - \boldsymbol{\tau}\|_2^2 + \lambda\|\boldsymbol{\rho}\|_2^2, \tag{3}$$

in which λ represents the trade-off between data fidelity and regularity (smoothness) of the solution.

O'Toole *et al*. [12] note that problem (3) can be solved in an efficient manner with their so-called light-cone transform (LCT). If we denote their resampling operator by $\mathbf{T}$, we can
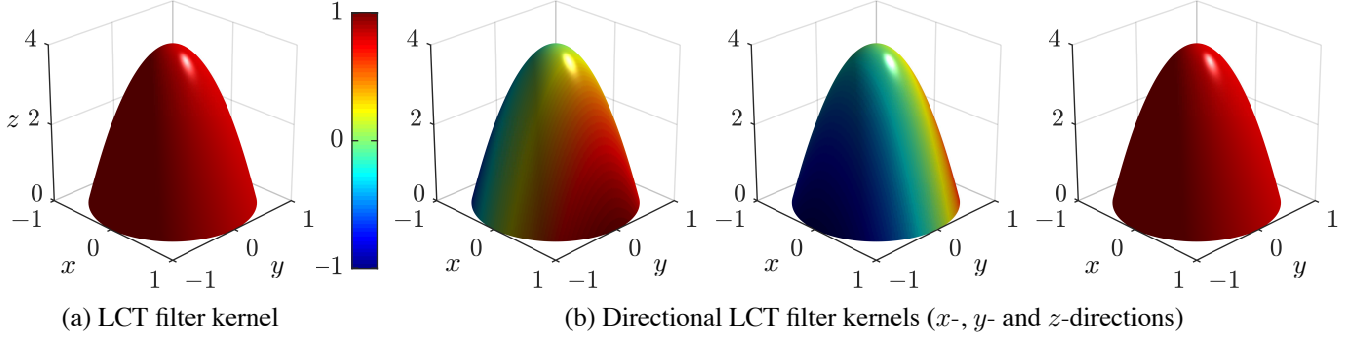
(a) LCT filter kernel   (b) Directional LCT filter kernels (x-, y- and z-directions)

Figure 3. Constructing the D-LCT filter kernels: The light-cone transform produces a three-dimensional, shift-invariant kernel (a). The D-LCT (b) consists of three shift-invariant kernels that relate directional albedo (albedo + normal) to the transients. The z-directional D-LCT kernel (b, far right) is identical to the LCT kernel (a).
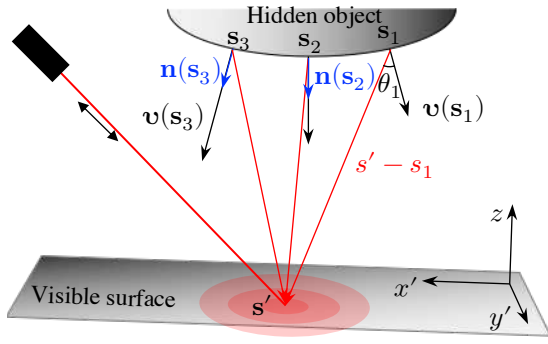

Figure 4. Directional albedo model: At location $\mathbf{s} = (x, y, z)$ on the object, directional albedo $\boldsymbol{\upsilon}(\mathbf{s})$ has direction and magnitude of the normal $\mathbf{n}(\mathbf{s})$ and the albedo $\rho(\mathbf{s})$, respectively. Contribution of albedo $\rho(\mathbf{s})$ to the surface at $\mathbf{s}' = (x', y', z = 0)$ decreases, in the first order, as the cosine of the angle θ between $\boldsymbol{\upsilon}(\mathbf{s})$ and $\mathbf{s}' - \mathbf{s}$.

express problem (3) equivalently in terms of the resampled albedo $\bar{\boldsymbol{\rho}} = \mathbf{T}^*\boldsymbol{\rho}$ and transients $\bar{\boldsymbol{\tau}} = \mathbf{T}^*\boldsymbol{\tau}$ as

$$\text{minimize } f(\bar{\boldsymbol{\rho}}) = \|\mathbf{H}\bar{\boldsymbol{\rho}} - \bar{\boldsymbol{\tau}}\|_2^2 + \lambda\|\bar{\boldsymbol{\rho}}\|_2^2, \tag{4}$$

in which $\mathbf{H} = \mathbf{T}^*\mathbf{K}\mathbf{T}$ is a three-dimensional filter (that is, a linear space-invariant operator), whose impulse response is shown in Figure 3 (a). Section 1 of the supplement provides the details of the resampling operator $\mathbf{T}^*$.

Since $\mathbf{H}$ is a three-dimensional filter, we can compute the solution $\bar{\boldsymbol{\rho}}^{\text{opt}} = (\mathbf{H}^*\mathbf{H} + \lambda\mathbf{I})^{-1}\mathbf{H}^*\bar{\boldsymbol{\tau}}$ of (4) efficiently in the Fourier domain using Wiener deconvolution. The regularity parameter λ can be interpreted as the noise-to-signal ratio in this filtering context. The solution of our original problem (3) is obtained by resampling the deconvolved solution using the adjoint resampling operator, that is, $\boldsymbol{\rho}^{\text{opt}} = \mathbf{T}\bar{\boldsymbol{\rho}}^{\text{opt}}$.

3.2. The Directional Albedo Model

In the case of isotropic point emitters, (1) is an adequate model for higher-order light transport. However, for typical diffuse or Lambertian object surfaces, such a model ignores the radiometric fall-off due to Lambert's cosine law, i.e., the fall-off due to the angle between the incident light rays and the surface normals; see [50]. Incorporating cosine terms in (1) not only yields a more accurate forward model, but more importantly, it enables recovery of surface normals from the transients via the inverse model.

Denoting the two spatial coordinates by $\mathbf{s} = (x, y, z)$ and $\mathbf{s}' = (x', y', z = 0)$ for brevity, we update model (1) as

$$\tau(x', y', t) = \iiint_\Omega d\mathbf{s} \; \left\langle \frac{\rho(\mathbf{s})\,\mathbf{n}(\mathbf{s})}{r^4}, \frac{\mathbf{s}' - \mathbf{s}}{\|\mathbf{s}' - \mathbf{s}\|} \right\rangle$$
$$\cdot \delta\big(2\sqrt{(x' - x)^2 + (y' - y)^2 + z^2} - tc\big), \tag{5}$$

in which $\mathbf{n}(\mathbf{s}) = (n_x, n_y, n_z)(\mathbf{s}) \in \mathbb{R}^3$ is the surface normal at $\mathbf{s}$. Our model can be seen as a first-order approximation of the physically based one in [19], allowing us to reformulate normal estimation as a linear least-squares problem. Model (5) is also identical to the one in [13] save for the absence of occlusion terms. Section 3 of our paper supplement derives the relationship between (5) and the physical one. Assuming further that the projections

$$\left\langle \mathbf{n}(\mathbf{s}), \frac{\mathbf{s}' - \mathbf{s}}{\|\mathbf{s}' - \mathbf{s}\|} \right\rangle = \cos\theta = 1, \quad \forall\, \mathbf{s}', \mathbf{s}, \tag{6}$$

model (5) reduces to the volumetric albedo model (1).

Note in (5) that albedo $\rho(\mathbf{s}) \in \mathbb{R}$ is a scalar quantity, and surface normal $\mathbf{n}(\mathbf{s}) \in \mathbb{R}^3$ is a unit-norm vector. Rather than represent the two quantities using separate variables, we can combine them into a single directional-albedo vector

$$\boldsymbol{\upsilon}(\mathbf{s}) = (\upsilon_x, \upsilon_y, \upsilon_z)(\mathbf{s}) = \rho(\mathbf{s})\,\mathbf{n}(\mathbf{s}) \in \mathbb{R}^3, \tag{7}$$

such that the direction and the magnitude of $\boldsymbol{\upsilon}(\mathbf{s})$ encode its surface normal and albedo, respectively. Substituting (7) in model (5) and using $r = \|\mathbf{s}' - \mathbf{s}\|$, we obtain the directional albedo model

$$\tau(x', y', t) = \iiint_\Omega d\mathbf{s} \; \frac{\langle \boldsymbol{\upsilon}(\mathbf{s}), \mathbf{s}' - \mathbf{s} \rangle}{r^5}$$
$$\cdot \delta\big(2\sqrt{(x' - x)^2 + (y' - y)^2 + z^2} - tc\big), \tag{8}$$

relating the directional albedo $\boldsymbol{\upsilon}$ to the transients τ. Figure 4 illustrates and summarizes the directional albedo model.
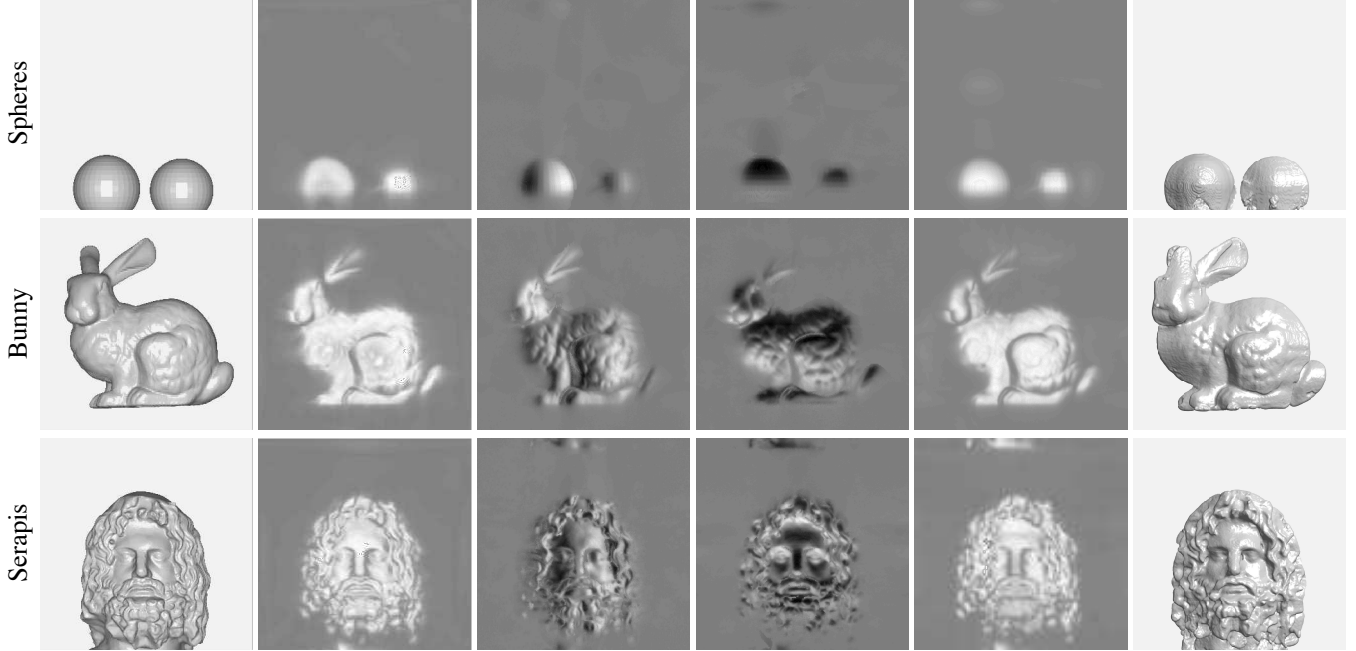
| | (a) Hidden object | (b) LCT volume | (c) D-LCT normal volumes (x-, y- and z-components) | (d) D-LCT surface |

Figure 5. **Transient imaging using the D-LCT:** D-LCT (c) captures fine details of object surfaces (a) not captured by the LCT (b). Volumes (b)–(c) are rendered using maximum intensity projection. D-LCT surfaces (d) are fit directly onto the D-LCT normals (c). For (d), we used known background masks to first remove the background points. All hidden objects have diffuse surfaces.

To discretize our directional albedo model, we sample Ω similarly to (2). This produces the system of linear equations in our directional albedos $\boldsymbol{\upsilon} = \left(\boldsymbol{\upsilon}_x^*, \boldsymbol{\upsilon}_y^*, \boldsymbol{\upsilon}_z^*\right)^*$:

$$\boldsymbol{\tau} = \mathbf{KS}\boldsymbol{\upsilon} \qquad (9)$$

in which $\mathbf{K}$ is the matrix from (2), and we obtain the entries of $\mathbf{S} = (\mathbf{S}_x, \mathbf{S}_y, \mathbf{S}_z)$ by sampling $(\mathbf{s}' - \mathbf{s})$ on Ω. Recovering the directional albedo $\boldsymbol{\upsilon}$ given the transients $\boldsymbol{\tau}$ is ill-posed in that it requires us to find the values of $3N^2M$ variables with only N^2M equations. Such a rank-deficient problem can be solved by formulating (9) as the least-squares problem

$$\text{minimize } f(\boldsymbol{\upsilon}) = \|\mathbf{KS}\boldsymbol{\upsilon} - \boldsymbol{\tau}\|_2^2 + \lambda\|\boldsymbol{\upsilon}\|_2^2 \qquad (10)$$

similarly to the regularized approach in (3). While we could alternatively optimize the TV-L1 version of problem (10) to obtain a better solution, we focus on our L2 variant for now to solve (10) efficiently as a vector deconvolution problem.

3.3. Directional Light-cone Transform

Whereas our least-squares optimization problem (10) has the simple, closed-form solution

$$\boldsymbol{\upsilon}^{\text{opt}} = (\mathbf{S}^*\mathbf{K}^*\mathbf{KS} + \lambda\mathbf{I})^{-1}\mathbf{S}^*\mathbf{K}^*\boldsymbol{\tau}, \qquad (11)$$

this solution is too expensive to compute naively for typical problems with $V^3 = N^2M \approx 10^8$ voxels. Computing $\boldsymbol{\upsilon}^{\text{opt}}$ directly with numerical methods such as Cholesky and LDL decompositions would incur a $O(V^9)$ cost whereas iterative ones (e.g. conjugate gradients), a $O(V^6)$ cost. These general

methods are therefore unsuited to practical problem sizes.

To solve problem (10) efficiently, we generalize the LCT technique used in (4) to the vectorial problem. We can write problem (10) equivalently as

$$\text{minimize } f(\overline{\boldsymbol{\upsilon}}) = \|\mathbf{H}(\mathbf{S}_x, \mathbf{S}_y, \mathbf{I})\overline{\boldsymbol{\upsilon}} - \overline{\boldsymbol{\tau}}\|_2^2 + \lambda\|\overline{\boldsymbol{\upsilon}}\|_2^2, (12)$$

in which

$$\underbrace{\begin{bmatrix} \overline{\boldsymbol{\upsilon}}_x \\ \overline{\boldsymbol{\upsilon}}_y \\ \overline{\boldsymbol{\upsilon}}_z \end{bmatrix}}_{\overline{\boldsymbol{\upsilon}}} = \underbrace{\begin{bmatrix} \mathbf{T}_x^* & & \\ & \mathbf{T}_y^* & \\ & & \mathbf{T}_z^* \end{bmatrix}}_{\mathbf{T}_d^*} \underbrace{\begin{bmatrix} \boldsymbol{\upsilon}_x \\ \boldsymbol{\upsilon}_y \\ \boldsymbol{\upsilon}_z \end{bmatrix}}_{\boldsymbol{\upsilon}} \qquad (13)$$

is the resampled variable, and $\overline{\boldsymbol{\tau}} = \mathbf{T}^*\boldsymbol{\tau}$ as before. From the solution $\overline{\boldsymbol{\upsilon}}^{\text{opt}}$ of (12), we recover the solution of the original problem (10) as $\boldsymbol{\upsilon}^{\text{opt}} = \mathbf{T}_d\overline{\boldsymbol{\upsilon}}^{\text{opt}}$. Section 2 of the supplement derives the resampler $\mathbf{T}_d^*$ and its relationship to $\mathbf{T}^*$ in (4).

Note in (12) that $\mathbf{S}_x$, $\mathbf{S}_y$ and $\mathbf{I}$ are shift-invariant, so they can be composed with the light-cone filter $\mathbf{H}$ to produce the directional light-cone filters $\mathbf{HS}_x$, $\mathbf{HS}_y$ and $\mathbf{HI}$. Their filter kernels are shown in Figure 3 (b). Since all the operators in problem (12) are filters, one can interpret (12) as a vectorial deconvolution problem, with a fixed noise-to-signal ratio λ across the vector frequencies. One can solve (12) efficiently using our vector extension of Wiener deconvolution.

3.4. Cholesky–Wiener Deconvolution

Denoting the matrices in (12) as $\mathbf{H}_x = \mathbf{HS}_x$, $\mathbf{H}_y = \mathbf{HS}_y$ and $\mathbf{H}_z = \mathbf{HI}$ for simplicity, we write the normal equations
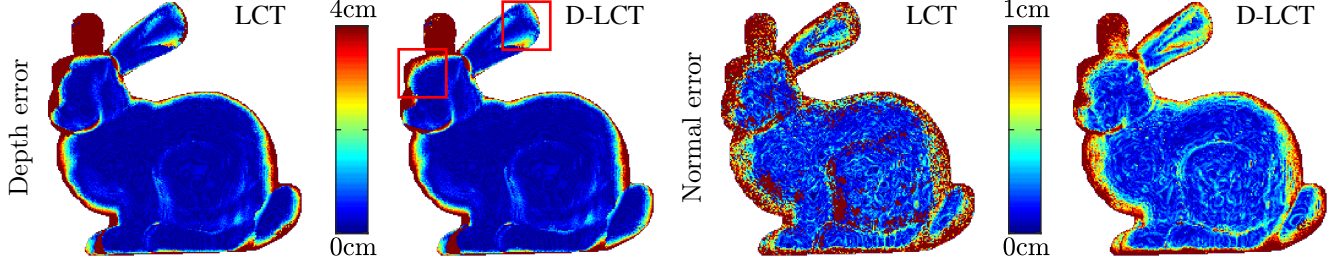
Figure 6. **Accuracy of D-LCT and LCT:** The LCT and D-LCT depths (left plots) have the RMSE of 5.97 and 4.96cm, and the MAE of 1.87 and 1.59cm, respectively. The LCT and D-LCT surface normals (right plots) have end-point RMSE 0.91 and 0.52cm, and MAE of 0.61 and 0.38cm, respectively. The LCT does not natively produce surface normals, so we obtain them using [54] with the LCT depth as the input.
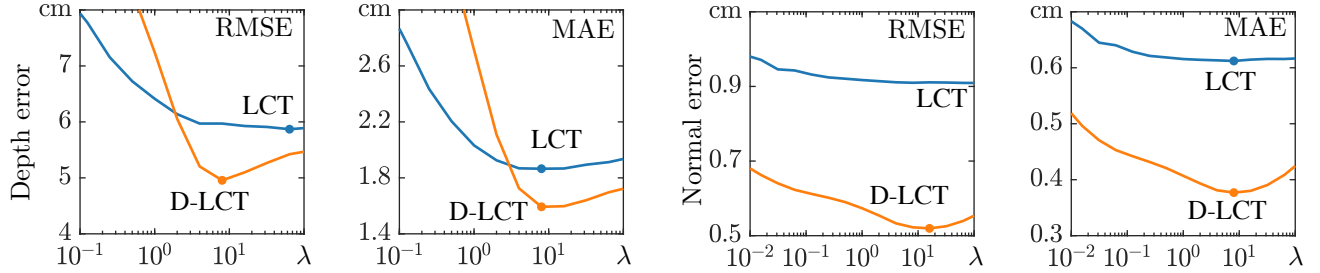

Figure 7. **Impact of regularity parameter on depth and surface normals:** The D-LCT produces accurate depth (left plots) and normals (right plots) over a wide range of values of λ, demonstrating its usefulness for cases where the SNR is not known exactly. The LCT normal errors are for the normal vectors obtained using [53] with the LCT depth as the input. Points of minimum error marked with dots.

associated with the least-squares problem (12) as

$$
\underbrace{\begin{bmatrix} \mathbf{H}_x^2 + \lambda\mathbf{I} & \mathbf{H}_x^*\mathbf{H}_y & \mathbf{H}_x^*\mathbf{H}_z \\ \mathbf{H}_y^*\mathbf{H}_x & \mathbf{H}_y^2 + \lambda\mathbf{I} & \mathbf{H}_y^*\mathbf{H}_z \\ \mathbf{H}_z^*\mathbf{H}_x & \mathbf{H}_z^*\mathbf{H}_y & \mathbf{H}_z^2 + \lambda\mathbf{I} \end{bmatrix}}_{\mathbf{H}^*\mathbf{H}+\lambda\mathbf{I}} \underbrace{\begin{bmatrix} \bar{\boldsymbol{\upsilon}}_x \\ \bar{\boldsymbol{\upsilon}}_y \\ \bar{\boldsymbol{\upsilon}}_z \end{bmatrix}}_{\widetilde{\boldsymbol{\upsilon}}} = \underbrace{\begin{bmatrix} \mathbf{H}_x^*\bar{\boldsymbol{\tau}} \\ \mathbf{H}_y^*\bar{\boldsymbol{\tau}} \\ \mathbf{H}_z^*\bar{\boldsymbol{\tau}} \end{bmatrix}}_{\mathbf{H}^*\bar{\boldsymbol{\tau}}} \quad (14)
$$

that is, a 3×3 block system of equations, where each block element is a filter or a filter signal. The structure of (14) thus suggests that we solve the 3×3 system using the Cholesky decomposition, performing the two associated forward- and back-substitutions using filtering operations. The right-hand side vector $\mathbf{H}^*\bar{\boldsymbol{\tau}}$ can be computed in the Fourier domain, as each $\mathbf{H}_x^*$, $\mathbf{H}_y^*$ and $\mathbf{H}_z^*$ is a filter.

Using the LDL variant of the Cholesky factorization, we factor the matrix $\mathbf{A} = \mathbf{H}^*\mathbf{H} + \lambda\mathbf{I}$ as $\mathbf{A} = \mathbf{L}\mathbf{D}\mathbf{L}^*$, where

$$
\mathbf{L} = \begin{bmatrix} \mathbf{I} & & \\ \mathbf{L}_{yx} & \mathbf{I} & \\ \mathbf{L}_{zx} & \mathbf{L}_{zy} & \mathbf{I} \end{bmatrix}, \mathbf{D} = \begin{bmatrix} \mathbf{D}_{xx} & & \\ & \mathbf{D}_{yy} & \\ & & \mathbf{D}_{zz} \end{bmatrix}, \quad (15)
$$

and the elements of $\mathbf{L}$ and $\mathbf{D}$ are given by

$$
\begin{aligned}
\mathbf{D}_{jj} &= \mathbf{H}_j^*\mathbf{H}_j + \lambda\mathbf{I} - \textstyle\sum_{k=1}^{j-1}\mathbf{L}_{jk}\mathbf{D}_{kk}\mathbf{L}_{jk}^* \\
\mathbf{L}_{ij} &= \mathbf{D}_{jj}^{-1}\big(\mathbf{H}_i^*\mathbf{H}_j - \textstyle\sum_{k=1}^{j-1}\mathbf{L}_{jk}\mathbf{D}_{kk}\mathbf{L}_{jk}^*\big),
\end{aligned} \quad (16)
$$

using the convention $1 = x, 2 = y, 3 = z$ in both sums. We can readily verify the dynamic programming procedure (16) by applying the elimination steps of the Cholesky algorithm [51] to the block elements of matrix $\mathbf{A}$.

Finally, the triangularized system $\mathbf{L}\mathbf{D}\mathbf{L}^*\boldsymbol{\upsilon} = \mathbf{H}^*\boldsymbol{\tau}$ can be solved using forward- and back-substitutions

$$
\boldsymbol{\nu} = \mathbf{L}^{-1}\mathbf{H}^*\boldsymbol{\tau}, \quad \boldsymbol{\upsilon} = \mathbf{L}^{-*}\mathbf{D}^{-1}\boldsymbol{\nu}, \quad (17)
$$

both of which can be performed as a series of filtering steps in the Fourier domain. For example, the block-elements of $\boldsymbol{\nu}$ can be obtained using $\boldsymbol{\nu}_1 = \mathbf{H}_x^*\bar{\boldsymbol{\tau}}, \boldsymbol{\nu}_2 = \mathbf{H}_y^*\bar{\boldsymbol{\tau}} - \mathbf{L}_{yx}\boldsymbol{\nu}_1$ and $\boldsymbol{\nu}_3 = \mathbf{H}_y^*\bar{\boldsymbol{\tau}} - \mathbf{L}_{zx}\boldsymbol{\nu}_1 - \mathbf{L}_{zy}\boldsymbol{\nu}_2$. Observe here $\mathbf{L}_{yx}\boldsymbol{\nu}_1, \mathbf{L}_{zx}\boldsymbol{\nu}_1$ and $\mathbf{L}_{zy}\boldsymbol{\nu}_2$ may be computed in the Fourier domain, as each matrix $\mathbf{L}_{yx}, \mathbf{L}_{zx}$ and $\mathbf{L}_{zy}$ represents a 3D filter. We compute the elements of $\boldsymbol{\upsilon}$ can be computed in a similar manner.

For large problems where storing the Fourier coefficients of the block elements of $\mathbf{L}$ and $\mathbf{D}$ is not feasible, we can use an iterative solver like conjugate gradients, and compute the forward mapping $\mathbf{x} \mapsto (\mathbf{H}^*\mathbf{H} + \lambda\mathbf{I})\mathbf{x}$ using, again, filtering operations in the Fourier domain.

3.5. Surface Reconstruction

Having obtained the field $\boldsymbol{\upsilon}$ of directional albedo, we use the method of [52] to fit a nonparametric surface. Fitting the surface amounts to recovering an indicator function χ of the scene object so that the gradient of χ equals $\boldsymbol{\upsilon}$. Expressed as an optimization problem, we have

$$
\text{minimize } f(\chi) = \|\mathbf{G}^*\mathbf{G}\chi - \mathbf{G}^*\boldsymbol{\upsilon}\|_2^2 + \lambda\|\chi\|_2^2, \quad (18)
$$

in which $\mathbf{G}$ denotes a discretization of the three-dimensional gradient operator. In practice, however, the zero-level-set of the solution χ^{opt} of (18) may deviate from the surface of the true scene object due to the noise in $\boldsymbol{\upsilon}$ and the discretization
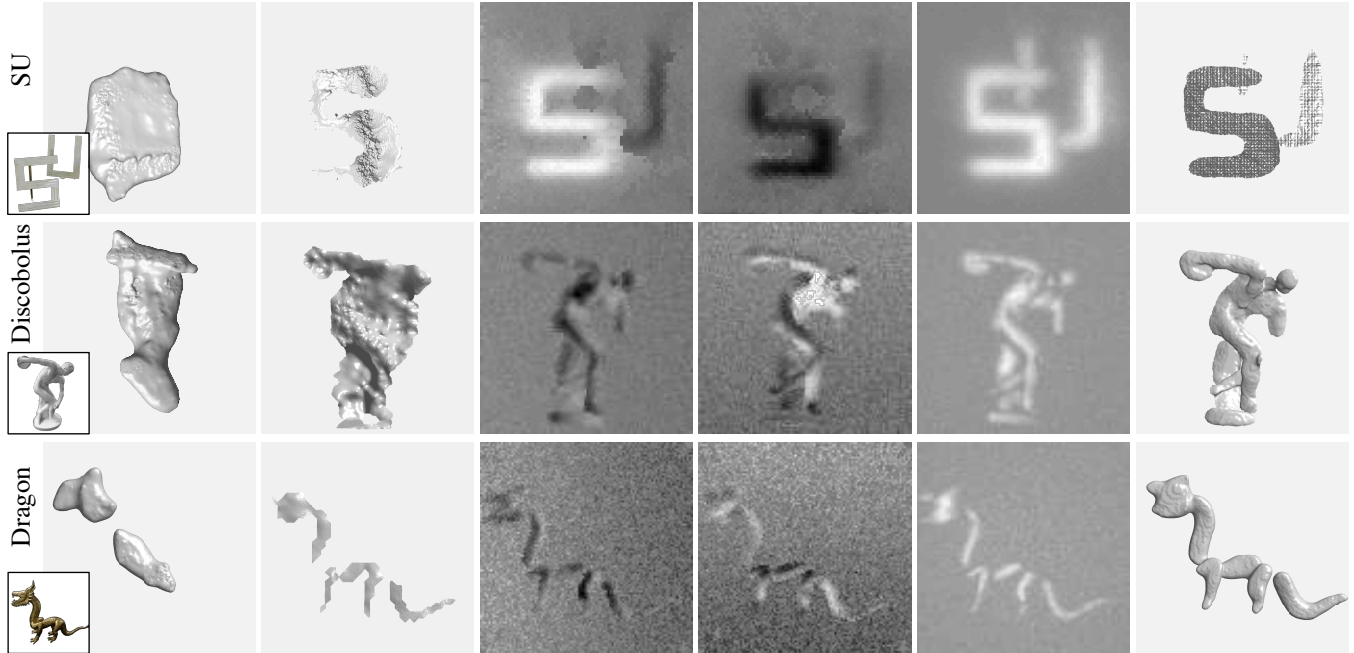
| | (a) Fermat flow | (b) Tsai *et al*. [19] | (c) D-LCT normal volumes (x-, y- and z-components) | (d) D-LCT surface |

Figure 8. **Surface reconstruction using captured data:** SU has a spatial resolution of 64×64 pixels (1min exposure), and the remaining scenes, 512×512 (180min exposure). We use $\lambda = 2^0$, 2^3 and 2^3 for SU, Discobolus and Dragon, respectively. Insets in the left-most column show the scene objects. Fermat flow [18] and the method of Tsai *et al*. [19] only partially reconstruct the surfaces.

	Methods	64×64	128×128	256×256	512×512
Albedo	FBP	0.6s	2.0s	8.3s	32.4s
	Phasor Fields	0.9s	3.1s	12.6s	49.3s
	f–k migration	1.5s	6.6s	20.9s	72.6s
	LCT	0.5s	2.4s	8.5s	30.4s
Normals	Fermat flow	1.6s	5.6s	21.3s	86.5s
	Heide *et al*.	>10h	N/A	N/A	N/A
	Tsai *et al*.	7h	N/A	N/A	N/A
	D-LCT (Ours)	5.2s	21.9s	89.5s	370.0s

Table 1. **Running times of various methods:** Measured using an 8-core, 2.70GHz CPU for the Bunny, all at temporal resolutions of 512 pixels. The methods of Heide *et al*. [13] and Tsai *et al*. [19] do not scale to resolutions higher than 64×64.

of the problem. To mitigate such issues, we follow again the approach of [52] to extract an isosurface of χ^{opt} instead.

4. Experimental Results

Since our directional LCT approach recovers both albedo and surface normals, it can be used in the traditional context of two-dimensional NLOS imaging as well as to reconstruct surfaces in three dimensions. We demonstrate our approach for both use-cases, also comparing it to methods specialized to each one. We use ZNLOS [53] and Stanford [14] datasets for experimental validation. The ZNLOS dataset consists of multiple-bounce transients of synthetic objects 0.50m away from a $1\mathrm{m} \times 1\mathrm{m}$ visible surface. The dataset has a temporal resolution of 512 pixels with bins of width 10ps, and spatial

resolutions of 256×256 pixels. The Stanford set consists of transients measured on a $2\mathrm{m} \times 2\mathrm{m}$ surface of natural hidden objects 1m away, with ambient light and noise. This dataset has a spatial resolution of 512×512 or 64×64 pixels, and a temporal resolution of 512 with bins of width 32ps.

4.1. NLOS Imaging Experiments

Directional Transient Imaging. Figure 5 shows the normal images obtained using the D-LCT. These images contain the fine variations in object surfaces such the smooth surface of the spheres and the fur of the bunny. These details would be difficult to recover post-hoc from albedo-only images using detail-enhancement techniques, for example. The results for the LCT resemble the z-component of the D-LCT ones with subtle differences that can be expected from the simplifying assumption (6). We render directional albedo volumes using maximum intensity projection: for each point $(x', y', z = 0)$ on the image plane, we find along the z-axis the directional-albedo with maximum z-component values. We reconstruct the surfaces by first masking out the background pixels with ground-truth masks, and performing Poisson reconstruction on the foreground points and the directional albedo. We use $\lambda = 2^3$ for all scenes.

Accuracy of Depth and Surface Normals. Figure 6 shows the error maps for the recovered depth and surface normals of the "Bunny". The recovered LCT and D-LCT depth maps have root-mean-squared errors (RMSE) 5.97cm and 4.96cm

and the mean absolute errors (MAE) of 1.87cm and 1.59cm respectively, across foreground pixels. The surface normals estimated using the LCT and the D-LCT have the end-point RMSE of 0.91cm and 0.52cm, MAE of 0.61cm and 0.38cm respectively. Note, the LCT does not, by itself, produce any surface normals, so we obtain the normals using the method [54] with the LCT depth as the input (we use 6 neighboring points to produce the optimum results). In Figure 7, we plot the influence of regularization parameter λ on the depth and surface normal errors, illustrating that the D-LCT performs stable over a wide range of λ. This can be useful in practical imaging scenarios in which the signal-to-noise ratio λ of the captured transient data is not known exactly.

Surface Reconstruction with Captured Data. To show the robustness of the D-LCT against different types of noise that are present in real capture environments, we perform surface reconstruction with the Stanford dataset. Figure 8 shows the directional albedo and surfaces of recovered SU, Discobolus and the Dragon objects. We reconstruct the surfaces by first thresholding the norm of directional albedo vectors to mask out the background points, then performing Poisson surface reconstructions on the remaining foreground points. We use $\lambda = 2^0, 2^3$ and 2^3 for the three scenes. In the SU scene, the normal volumes reveal the orientation of the letters S and U (S points to the upper-left, U points to the upper-right). The left part of U is partially occluded, so the different methods produce different maximal intensity projections along the z-axis. For these reconstruction tasks with noisy transients, the method of Tsai *et al.* [19] and Fermat flow [18] reconstruct only the rough shapes of the objects. We initialize [19] using the LCT, but other initializations are also possible. We used the `--density` flag in the Poisson reconstruction software [48] to avoid the fusion of nearby surface segments.

Computational Efficiency. While the D-LCT has the same computational complexity as the LCT, we perform $9\times$ more computations per voxel due to the outer Cholesky factoring required. The D-LCT is $1000\times$ faster compared with similar methods that are capable of recovering the surface normals of objects with a complex geometry. While Fermat flow [18] is $4\times$ faster than our approach, it is applicable mostly to the reconstruction of surfaces of objects with simpler geometry such as a bowl or a sphere (see Section 3 of the supplement for the reconstructions). Table 1 provides the running times of different methods on an 8-core, 2.70GHz CPU.

5. Discussion

Our work proposes an efficient method to jointly estimate the albedo and the surface normals of NLOS objects using a deconvolution approach. Our Directional LCT has the same low computational complexity as albedo-only methods, e.g. f–k migration and the LCT, but is capable of reconstructing high-quality surfaces.

Limitations. Our forward model (8) assumes the scene has mostly non-specular surfaces. Fortunately, our least-squares inverse method provides some degree of robustness against specularities by treating them as outliers (see e.g. the Dragon reconstruction, Figure 8). Similarly, we treat occlusions in the scene as outliers to our least-squares formulation. Using an L_1-based data fidelity term instead of our L_2-based one (10) could further improve the robustness of our method, at the cost of increased computation times. The L_1-based data-term also enforces sparsity, which may remove the need for masking out background pixels.

Our forward model also linearizes the cosine fall-off due to the interaction between surface normals and the two light rays (incident and reflected). Our linearized fall-off model is an under-estimator of the true fall-off, and surface locations that make larger angles on average with the visible wall are estimated to be at positions closer to the visible wall, where the fall-off is indeed less. This causes rounder surfaces to be estimated slightly flatter than they should be (see the arms of the Discobolus, Figure 8), but not as flat as the estimates of the LCT, which assumes zero cosine fall-off. This issue can be overcome by iteratively reweighting the first term of (10) using the ratio of the true fall-off to the linear one, based on the normals last estimated; see Section 3 of the supplement.

Future Work. To improve reconstruction times, we plan to implement the D-LCT procedure on a GPU. Similarly to the original LCT, the D-LCT is highly parallelizable and can be significantly accelerated using a GPU implementation. Like the GPU implementation of the LCT, we expect GPU-based D-LCT to require milliseconds of processing time for lower spatial resolutions, e.g. 32×32 or 64×64 pixels. We plan also to consider L_1 or TV regularizers to better preserve the discontinuities in the reconstructed surfaces.

6. Conclusion

NLOS imaging approaches have typically been classified as recovering either the albedo or the surface normals of the hidden objects. In this work, we showed that it is possible to recover both quantities jointly. In closing, reconstruction of surfaces of hidden 3D objects can be regarded as the next frontier for NLOS imaging because it allows us to represent better the 3D environment we ultimately live in. We believe the D-LCT is a big step towards pushing beyond volumetric albedo approaches, providing a practical way to estimate the hidden surface normals needed for surface reconstruction.

References

[1] Jacopo Bertolotti, Elbert G. van Putten, Christian Blum, Ad Lagendijk, Willem L. Vos, and Allard P. Mosk. Non-invasive imaging through opaque scattering layers. *Nature*, 491(7423):232–234, 2012.

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

[3] Brandon M. Smith, Matthew O'Toole, and Mohit Gupta. Tracking multiple objects outside the line of sight using speckle imaging. In *CVPR*, 2018.

[4] Ahmed Kirmani, Tyler Hutchison, James Davis, and Ramesh Raskar. Looking around the corner using transient imaging. In *ICCV*, 2009.

[5] Christos Thrampoulidis *et al*. Exploiting occlusion in non-line-of-sight active imaging. *IEEE Trans. Comput. Imaging*, 4(3):419–431, 2018.

[6] Katherine L. Bouman *et al*. Turning corners into cameras: Principles and methods. In *ICCV*, 2017.

[7] Manel Baradad *et al*. Inferring light fields from shadows. In *CVPR*, 2018.

[8] Mahed Batarseh, Sergey Sukhov, Zhean Shen, Heath Gemar, Roxana Rezvani, and Aristide Dogariu. Passive sensing around the corner using spatial coherence. *Nat. Commun.*, 9(1):1–6, 2018.

[9] Charles Saunders, John Murray-Bruce, and Vivek K. Goyal. Computational periscopy with an ordinary digital camera. *Nature*, 565(7740):472–475, 2019.

[10] David B. Lindell, Gordon Wetzstein, and Vladlen Koltun. Acoustic non-line-of-sight imaging. In *CVPR*, 2019.

[11] Andreas Velten, Thomas Willwacher, Otkrist Gupta, Ashok Veeraraghavan, Moungi G. Bawendi, and Ramesh Raskar. Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging. *Nat. Commun.*, 3:745, 2012.

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

[13] Felix Heide, Matthew O'Toole, Kai Zang, David B. Lindell, Steven Diamond, and Gordon Wetzstein. Non-line-of-sight imaging with partial occluders and surface normals. *ACM Trans Graph*, 38(3):22:1–22:10, 2019.

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

[15] Christopher A. Metzler, David B. Lindell, and Gordon Wetzstein. Keyhole imaging: non-line-of-sight imaging and tracking of moving objects along a single optical path at long standoff distances. *ArXiv191206727 Cs Eess*, 2019.

[16] Marco La Manna, Fiona Kine, Eric Breitbach, Jonathan Jackson, Talha Sultan, and Andreas Velten. Error backprojection algorithms for non-line-of-sight imaging. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(7):1615–1626, 2019.

[17] Chia-Yin Tsai, Kiriakos N. Kutulakos, Srinivasa G. Narasimhan, and Ashwin C. Sankaranarayanan. The geometry of first-returning photons for non-line-of-sight imaging. In *CVPR*, 2017.

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

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

[20] Felix Heide, Lei Xiao, Wolfgang Heidrich, and Matthias B. Hullin. Diffuse mirrors: 3D reconstruction from diffuse indirect illumination using inexpensive time-of-flight sensors. In *CVPR*, 2014.

[21] Achuta Kadambi, Hang Zhao, Boxin Shi, and Ramesh Raskar. Occluded imaging with time-of-flight sensors. *ACM Trans Graph*, 35(2):15:1–15:12, 2016.

[22] Jonathan Klein, Christoph Peters, Jaime Martín, Martin Laurenzis, and Matthias B. Hullin. Tracking objects outside the line of sight using 2D intensity images. *Sci. Rep.*, 6:32491, 2016.

[23] Mauro Buttafava, Jessica Zeman, Alberto Tosi, Kevin Eliceiri, and Andreas Velten. Non-line-of-sight imaging using a time-gated single photon avalanche diode. *Opt. Express*, 23(16):20997–21011, 2015.

[24] Genevieve Gariepy *et al*. Single-photon sensitive light-in-fight imaging. *Nat. Commun.*, 6:6021, 2015.

[25] Feihu Xu *et al*. Revealing hidden scenes by photon-efficient occlusion-based opportunistic active imaging. *Opt. Express*, 26(8):9945–9962, 2018.

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

[27] Matthew O'Toole, Felix Heide, David B. Lindell, Kai Zang, Steven Diamond, and Gordon Wetzstein. Reconstructing Transient Images From Single-Photon Sensors. In *CVPR*, 2017.

[28] Matthew O'Toole, David B. Lindell, and Gordon Wetzstein. Real-time Non-line-of-sight Imaging. In *ACM SIGGRAPH 2018 Emerging Technologies*, 2018.

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

[30] Yoann Altmann, Stephen McLaughlin, Miles J. Padgett, Vivek K. Goyal, Alfred O. Hero, and Daniele Faccio. Quantum-inspired computational imaging. *Science*, 361(6403):eaat2298, 2018.

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

[32] Byeongjoo Ahn, Akshat Dave, Ashok Veeraraghavan, Ioannis Gkioulekas, and Aswin C. Sankaranarayanan. Convolutional Approximations to the General Non-Line-of-Sight Imaging Operator. In *ICCV*, 2019.

[33] Xiaochun Liu *et al*. Non-line-of-sight imaging using phasor-field virtual wave optics. *Nature*, :1–4, 2019.

[34] Matthew M. Loper and Michael J. Black. OpenDR: An

Approximate Differentiable Renderer. In *ECCV*, 2014.

[35] Pramook Khungurn, Daniel Schroeder, Shuang Zhao, Kavita Bala, and Steve Marschner. Matching real fabrics with micro-appearance models. *ACM Trans Graph*, 35(1):1:1–1:26, 2015.

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

[37] Shuang Zhao, Lifan Wu, Frédo Durand, and Ravi Ramamoorthi. Downsampling scattering parameters for rendering anisotropic media. *ACM Trans Graph*, 35(6):166:1–166:11, 2016.

[38] Adam Geva, Yoav Y. Schechner, Yonatan Chernyak, and Rajiv Gupta. X-ray computed tomography through scatter. In *ECCV*, 2018.

[39] Stephen Robert Marschner. Inverse Rendering for Computer Graphics, PhD Thesis, Cornell University, 1998.

[40] Gustavo Patow and Xavier Pueyo. A survey of inverse surface design from light transport behavior specification. *Comput. Graph. Forum*, 22:663–687, 2003.

[41] Ben Appleton and Hugues Talbot. Globally minimal surfaces by continuous maximal flows. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(1):106–118, 2006.

[42] Olivier Faugeras and Renaud Keriven. Complete dense stereovision using level set methods. In *ECCV*, 1998.

[43] Amaël Delaunoy and Emmanuel Prados. Gradient flows for optimizing triangular mesh-based surfaces: applications to 3d reconstruction problems dealing with visibility. *Int. J. Comput. Vis.*, 95(2):100–123, 2011.

[44] Mathieu Desbrun, Mark Meyer, Peter Schröder, and Alan H. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *SIGGRAPH*, 1999.

[45] Mark Meyer, Mathieu Desbrun, Peter Schröder, and Alan H. Barr. Discrete Differential-Geometry Operators for Triangulated 2-Manifolds. In *Visualization and Mathematics III*, 2003.

[46] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel. Laplacian Surface Editing. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, 2004.

[47] Ilya Eckstein, Jean-Philippe Pons, Yiying Tong, C.-C. Jay Kuo, and Mathieu Desbrun. Generalized surface flows for mesh processing. In *Proceedings of the Fifth Eurographics Symposium on Geometry Processing*, 2007.

[48] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, 2006.

[49] Per Christian Hansen. Discrete Inverse Problems: Insight and Algorithms. SIAM, 2010.

[50] Di Wu *et al*. Frequency Analysis of Transient Light Transport with Applications in Bare Sensor Imaging. In *Computer Vision – ECCV 2012*, 2012.

[51] Lloyd N. Trefethen and David Bau, III. Numerical Linear Algebra. SIAM, 1997.

[52] Michael Kazhdan and Hugues Hoppe. Screened Poisson Surface Reconstruction. *ACM Trans Graph*, 32(3):29:1–29:13, 2013.

[53] Graphics and Imaging Lab. Z-NLOS Dataset, Graphics and Imaging Lab. [Online]. Available: http://graphics.unizar.es/nlos_dataset. [Accessed: 13-Sep-2019].

[54] Hugues Hoppe, Tony DeRose, Tom Duchamp, John McDonald, and Werner Stuetzle. Surface Reconstruction from Unorganized Points. In *SIGGRAPH*, 1992.