

Iterating the Transient Light Transport Matrix for Non-Line-of-Sight Imaging

Supplementary Information

In Section S1, we derive camera functions, $\Phi(\cdot)$, that enable 3D and 4D reconstructions of the hidden scene. In Section S2, we derive two projector functions: one for reconstructing the hidden scene under synthesized illumination and another for extracting TLTM-2 from TLTM-1 by using beamforming principles to synthesize a virtual wave that focuses in space-time. Section S3 introduces mathematical tricks to accelerate 4D reconstruction, making TLTM-2 computation feasible by reducing computational complexity without increasing memory complexity. Section S4 provides implementation details for the TLTM-2 applications, including angular relighting, synthesized illumination, and direct-indirect separation. Section S5 validates the TLTM-2 reconstruction pipeline. Sections S6 and S7 then analyze its main artifacts, noise sensitivity, and temporal resolution limits. In Section S8, we discuss the computational complexity of computing an additional linear inverse using TLTM-2 to optimize projector functions for generating synthetic coherent and incoherent illumination in the hidden scene. Section S9 examines how increasing the number of SPAD pixels is equivalent to increasing exposure times during TLTM-1 acquisition. Section S10 demonstrates how focused virtual illumination enables separation of direct and indirect light transport components in a corner scene with strong interreflections.

S1 Phasor Field Imaging

In this section, we derive the camera functions in Eq. 4 that enable the generation of 3D and 4D reconstructions. We can use the filtered backprojection algorithm¹, to reconstruct each 3D voxel, $\vec{x}_v$, as a function of time t . This algorithm accounts for the time shifts to each voxel in our hidden scene from each combination of illumination position $\vec{x}_p$ and detector position $\vec{x}_c$ at the relay surface:

$$I_{\mathcal{P}}(\vec{x}_v, t) \triangleq \iint [\mathcal{P}(\vec{x}_p, t) * H(\vec{x}_p, \vec{x}_c, t)] \delta\left(t - \frac{|\vec{x}_p - \vec{x}_v| + |\vec{x}_c - \vec{x}_v|}{c}\right) d\vec{x}_c d\vec{x}_p \quad (\text{S1})$$

where $\mathcal{P}(\vec{x}_p, t)$ is the phasor field projector function². This 4D video can also be implemented efficiently in the *temporal* frequency, ω , domain:

$$\begin{aligned} I(\vec{x}_v, t) &= \mathcal{F}^{-1}\{I_{\mathcal{F}}(\vec{x}_v, \omega)\} \\ &= \iiint \mathcal{P}(\vec{x}_p, \omega) H(\vec{x}_p, \vec{x}_c, \omega) e^{-j\frac{\omega}{c}(|\vec{x}_p - \vec{x}_v| + |\vec{x}_c - \vec{x}_v|)} e^{j\omega t} d\omega d\vec{x}_c d\vec{x}_p \end{aligned} \quad (\text{S2})$$

where we have used the time shift property of the Fourier transform to convert a time shift into a frequency dependent phase shift³. After reconstruction, each voxel in the reconstructions lights up at $t = 0$ since we have shifted the time index to account for the photon time of flight from the illumination to the voxel and back to the sensor. To generate a transient video of the illumination moving through the hidden scene, we can add a voxel-dependent time shift to generate a global time index $t' = t + |\vec{x}_p - \vec{x}_v|/c$. For this shifted index, $t' = 0$ is when our illumination source starts emitting light and light arrives at each voxel at a time index determined by the speed of light. In the Fourier domain, this is equivalent to adding a frequency-dependent phase shift, $e^{-j\phi(\omega)} = e^{j\omega \frac{|\vec{x}_p - \vec{x}_v|}{c}}$:

$$\begin{aligned} I(\vec{x}_v, t') &= \mathcal{F}^{-1}\left\{I_{\mathcal{F}}(\vec{x}_v, \omega) e^{j\omega \frac{|\vec{x}_p - \vec{x}_v|}{c}}\right\} \\ &= \iiint \mathcal{P}(\vec{x}_p, \omega) H(\vec{x}_p, \vec{x}_c, \omega) e^{-j\frac{\omega}{c}(|\vec{x}_c - \vec{x}_v|)} e^{j\omega t'} d\omega d\vec{x}_c d\vec{x}_p \end{aligned} \quad (\text{S3})$$

where the phase offset no longer depends on $\vec{x}_p$. The non-transient, 3D reconstruction from prior art⁴⁻⁶ can be rewritten as the intensity of Eq. S1 at $t = 0$ i.e. we get:

$$\begin{aligned} I(\vec{x}_v) &= I(\vec{x}_v, t = 0) \\ &= \iiint [\mathcal{P}(\vec{x}_p, \omega) H(\vec{x}_p, \vec{x}_c, \omega)] e^{j\frac{\omega}{c}(|\vec{x}_p - \vec{x}_v| + |\vec{x}_c - \vec{x}_v|)} d\vec{x}_p d\vec{x}_c d\omega. \end{aligned} \quad (\text{S4})$$

where the inverse Fourier kernel, $e^{j\omega t}$, simplifies to 1 since $t = 0$.

S2 Projector Functions

S2.1 Synthesized Illumination

The phasor field kernel utilized for phasor field imaging^{4,6} in prior art is a modulated Gaussian in the time domain with some central wavelength of modulation given by $\omega_0 = 2\pi c/\lambda_C$ and the FWHM of the Gaussian given by $\sigma = 5\lambda_C/c$:

$$\begin{aligned}\mathcal{P}(\vec{x}_p, t) &= \delta(\vec{x} - \vec{x}_p) \mathcal{P}(t) \\ &= \delta(\vec{x} - \vec{x}_p) \frac{1}{\sqrt{2\pi}\sigma} e^{i\omega_0 t} e^{-\frac{t^2}{2\sigma^2}}\end{aligned}\quad (\text{S5})$$

First, we show that this projector function is a bandpass filter in the temporal Fourier domain. This filters our signal using a frequency-dependent complex number:

$$\begin{aligned}\mathcal{P}_{\mathcal{F}}(\vec{x}_p, \omega) &= \int \mathcal{P}(\vec{x}_p, t) e^{-i\omega t} dt \\ &= \delta(\vec{x} - \vec{x}_p) \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(\omega - \omega_0)^2 (\sigma)^2}{2}} \\ &= \delta(\vec{x} - \vec{x}_p) \mathcal{P}(\omega)\end{aligned}\quad (\text{S6})$$

This projector function illuminates the hidden scene with a gaussian wave⁴ propagating from $\vec{x} = \vec{x}_p$ in all directions, starting at $t' = 0$. By adding additional illumination positions $\vec{x}_p$ on the relay surface, we can generate 4D and 3D reconstructions under synthesized illumination using Eq. S2 and Eq. S4 respectively.

S2.2 Extracting TLTM-2 using Delay and Sum Beamformer

To extract TLTM-2, we need to focus our illumination at a specific point $\vec{x}_p^{(2)}$ in the hidden scene while imaging all the other voxels, $\vec{x}_c^{(2)}$. Each temporal Fourier component of Gaussian wave can be modeled as a wavefront of a coherent, monochromatic, spherical wave emitted at illumination positions $\vec{x}_p$ and detected at positions $\vec{x}_c$ on the relay wall with wavelength $\lambda_c = 2\pi\omega_0/c$ ^{4,7-11}. This enables us to model each $\vec{x}_p$ as an omnidirectional element of a virtual phased array system on the relay surface that emits phasor field waves. Using the delay-and-sum beamforming principles, we introduce timeshifts that depend on $\vec{x}_p^{(2)}$, i.e., $\Delta t(\vec{x}_p^{(2)})$, that shape the emitted wavefront to steer and focus the virtual illumination on a specific 3D point, $\vec{x}_p^{(2)}$, in the hidden scene. Thus, our new projector function can be written as:

$$\begin{aligned}\mathcal{P}_{\mathcal{F}}(\vec{x}_p, \omega) &= \int \mathcal{P}(\vec{x}_p, t - \Delta t(\vec{x}_p^{(2)})) e^{-i\omega t} dt \\ &= \delta(\vec{x} - \vec{x}_p) e^{-j\omega \Delta t(\vec{x}_p^{(2)})} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(\omega - \omega_0)^2 (\sigma)^2}{2}} \\ &= e^{-j\omega \Delta t(\vec{x}_p^{(2)})} \mathcal{P}(\omega)\end{aligned}\quad (\text{S7})$$

By linking our time delay to the spatial location of our virtual illumination source on the relay wall, we can define appropriate functions that can steer or focus our virtual beam.

BeamSteering

Our virtual illumination position x_p is a 3D vector with an x-component, x_{px} , y-component x_{py} , and z-component x_{pz} . The mean illumination location, $\bar{x}_p$, can be considered to be the beam origin and a spherical coordinate system consisting of (r, Θ, Φ) can be defined to steer the beam to any voxel location given by $\vec{x}_p^{(2)} = \vec{x}_v$, along a radial vector $\vec{r}$. For this coordinate system $r = |\vec{r}| = |\vec{x}_v - \bar{x}_p|$ is the radial distance, Θ is the zenith angle between the z-axis and the radial vector $\vec{r}$, and Φ is the azimuthal angle between the x-axis and the orthogonal projection of the radial vector, $\vec{r}$, onto the relay surface given by $z = 0$ plane. Assuming our relay wall is planar, we can steer our beam in 3D by introducing time delays that depend only on x_{px} and x_{py} . Then, using beamforming principles from ultrasound phased arrays¹², we can show that the beam can be steered to a voxel $\vec{x}_v$ using the following formulas:

$$\begin{aligned}
\Theta &= \arccos\left(\frac{x_{vz} - \bar{x}_{pz}}{r}\right) \\
\Phi &= \arctan\left(\frac{x_{vy} - \bar{x}_{py}}{x_{vx} - \bar{x}_{px}}\right) \\
\Delta t(x_p) &= \left(\frac{(x_{px} - \bar{x}_{px}) \sin \Theta \cos \Phi + (x_{py} - \bar{x}_{py}) \sin \Theta \sin \Phi}{c}\right)
\end{aligned} \tag{S8}$$

Beam Focusing

To steer a focused beam to a voxel $\vec{x}_v$, we introduce time delays that impart a spherical curvature to each virtual illumination point x_p and focus the beam originating from $\bar{x}_p$ at a focal distance $r = |\vec{r}| = |\vec{x}_v - \bar{x}_p|$:

$$\Delta t(x_p) = \frac{1}{c} \sqrt{(r \sin \Theta \cos \Phi - (x_{px} - \bar{x}_{px}))^2 + (r \sin \Theta \sin \Phi - (x_{py} - \bar{x}_{py}))^2 + (r \cos(\Theta))^2} \tag{S9}$$

S3 Efficient Computation for TLTM-2

In this section, we demonstrate how we can accelerate the computation of Eq. S3 in the temporal frequency domain. First, we recognize that the shifted time index, t' removes the phase shift dependence of each voxel, $\vec{x}_v$, on the illumination position so we first integrate along $\vec{x}_p$ to generate $\mathcal{P}_{\mathcal{F}}(\vec{x}_c, \omega)$. Rearranging Eq. S3:

$$\begin{aligned}
I(\vec{x}_v, t') &= \iint \underbrace{\left[\int \mathcal{P}(\vec{x}_p, \omega) H(\vec{x}_p, \vec{x}_c, \omega) d\vec{x}_p \right]}_{\mathcal{P}_{\mathcal{F}}(\vec{x}_c, \omega)} e^{-j\frac{\omega}{c}(|\vec{x}_c - \vec{x}_v|)} e^{j\omega t'} d\vec{x}_c d\omega \\
&= \int \underbrace{\left[\int \mathcal{P}_{\mathcal{F}}(\vec{x}_c, \omega) e^{-j\frac{\omega}{c}(|\vec{x}_c - \vec{x}_v|)} d\vec{x}_c \right]}_{R_{\vec{x}_v}(\mathcal{P}_{\mathcal{F}}(\vec{x}_c, \omega))} e^{j\omega t'} d\omega
\end{aligned} \tag{S10}$$

Provided all the detection points, x_c , lie on a regular grid on a planar surface, then the inner integral can be written as a convolution:

$$\begin{aligned}
R_{\vec{x}_v}(\mathcal{P}_{\mathcal{F}}(\vec{x}_c, \omega)) &= R_{\vec{x}_v}(\mathcal{P}_{\mathcal{F}}(x_c, y_c, z_v, \omega)) \\
&= \mathcal{P}_{\mathcal{F}}(x_c, y_c, 0, \omega) \underset{2D}{*} G(x_c, y_c, z_v, \omega)
\end{aligned} \tag{S11}$$

where $G(x_c, y_c, z_v, \omega) = e^{i\frac{\omega}{c}\sqrt{x_c^2 + y_c^2 + z_v^2}} / \sqrt{x_c^2 + y_c^2 + z_v^2}$. The reconstruction for all $\vec{x}_v$ at a fixed depth z_v and frequency ω can be computed simultaneously by using a single 2D FFT which significantly reduces the computational complexity^{5,13}. These mathematical tricks are valid as long as $\vec{x}_c$ is sampled using a regular grid and the projector function, $\mathcal{P}(\vec{x}_p, t)$, contains no implicit dependence on $\vec{x}_c$. Since this is true for all projector functions discussed in this paper, columns of TLTM-2 and 4D reconstructions under synthesized illumination can be generated using this fast reconstruction scheme.

S3.1 Computational Complexity

Let $L \times L$ denote the number of laser positions, $M \times M$ denote the number of SPAD pixels focused on the wall, T denote the number of time bins, and F the number of frequency components. Assuming we use the RSD algorithm to reconstruct a volumetric cube of side length N , we can show that the complexity of this algorithm is $O(N^3 \log N)$ ⁵ for a single SPAD pixel. To get the steady reconstruction for each SPAD pixel, we simply run this algorithm M^2 times and add the results together. This yields a complexity of $O(M^2 N^3 \log N)$. Sequential reconstruction can be used to preserve the memory complexity relative to reconstructing an individual SPAD pixel, while the process can be easily parallelized to speed up reconstruction⁶.

To generate a transient video for k total frames, we first compute the inner integral in Eq. S3 with respect to x_p . Adding the transient measurements together preserves the memory complexity. Then each frame has the same reconstruction complexity as RSD algorithm for a single SPAD, and running this k times gives the complexity $O(k N^3 \log N)$. This is the complexity for generating the LTM for a single

spatial focus for the illumination. When $\vec{x}_c$ is not sampled using a regular grid, Eq. 6 can be computed using numerical integration or using the backprojection solver⁴ with $O(kN^5)$ complexity.

This complexity is orders of magnitude faster than previous work¹⁴, where the corresponding complexity for a single spatial focus was given by $O(N^3 * L^2 * M^2)$ and can be approximated as $O(N^7)$ when $N \approx L \approx M$. Additionally, the output was a light transport matrix without any timing information. To generate the full TLTM, we need to repeat the previous computation for each spatial location in the hidden volume, yielding an overall complexity of $O(SkN^3 \log(N))$, where S is the number of spatial focus locations.

S3.2 Memory Complexity

For sequential reconstruction, the memory complexity remains the same as that of previous algorithms reported for single pixel SPAD⁵. For steady-state reconstruction, we load each histogram of dimension L^2T into memory and compute the reconstruction for an N^3 volume, which yields a complexity of $O(L^2T + N^3)$. This can be improved further if we directly load the temporal Fourier transform of the histogram, yielding a memory complexity of $O(L^2F + N^3)$ where $F < T$, usually by an order of magnitude.

Reconstructing the video for a single spatial focus has a similar memory complexity, since we load the histogram for each SPAD and add them together after applying a SPAD dependent temporal shift to focus the wavefront before reconstructing a 3D volume for each frame. To avoid increasing the memory complexity, we only save an image of a 2D surface for each frame (the location with maximum intensity along depth) yielding a complexity of $O(L^2F + kN^2)$ and avoiding storing a 4D cube. GPU-based reconstructions can increase reconstruction speed at the expense of requiring additional memory. To provide context, our GPU-based implementation can output a video for a single focus (one column of the TLTM-2) within 1-2 minutes while MATLAB-based sequential reconstruction takes around 30 minutes for 216 SPAD pixels and 141 video frames.

S4 Applications enabled by TLTM-2: implementation details

After computing the second-order TLTM, we can extract various aspects of light transport to showcase different applications, which we outline in this section.

S4.1 Angular relighting with virtual illumination

We can adjust our projector function to isolate the reconstruction from a single SPAD pixel, $\vec{x}_p = \vec{x}_0$:

$$\begin{aligned} \mathcal{P}(\vec{x}_p, \omega) &= \delta(\vec{x}_p - \vec{x}_0) \mathcal{P}(\vec{x}_p, \omega) \\ &= \mathcal{P}(\vec{x}_0, \omega) \end{aligned} \tag{S12}$$

Since each SPAD pixel samples a unique position on the relay wall, we apply Helmholtz reciprocity to reinterpret that sampled relay-wall position as a virtual point light source. Incorporating the modified projector function into Eq. S4 yields a corresponding 3D reconstruction, $I(\vec{x}_v, \vec{x}_p)$, of the hidden scene under illumination from the virtual source position $\vec{x}_p$ on the relay wall. Additionally, by incorporating this modified projector function into Eq. S2, we generate 4D videos, $I(\vec{x}_v, \vec{x}_p, t)$, that visualize how the hidden scene responds over time when synthetically relit from different illumination positions on the relay wall.

Adding up the reconstructions incoherently, by summing up the individual magnitudes, simulates the result of illuminating the hidden scene using an array of incoherent point light sources on the relay surface. In contrast, adding the reconstructions coherently enables the design of optimized projector functions — synthesized illumination patterns that focus virtual illumination at specific points on hidden surfaces, effectively creating virtual point light sources at those locations, as described in the following section. Then, we can synthesize arbitrary illumination patterns as a combination of these individual point sources.

S4.2 Scene Relighting with Synthesized Illumination

Coherent Illumination

We can create a virtual coherent point light source in the hidden scene by synthetically focusing the illumination using coherent beamforming outlined in Eq. 5. Alternately, we can optimize our projector function over $\vec{x}_p$ repeatedly by recomputing the complex coefficients, $\mathcal{X}$, in Eqs. 7–9 to focus on individual

points in the hidden scene. Retaining temporal coherence for the point light source in the hidden scene is important for iterating on TLTM-2 and extracting higher-order TLTM. In either case, the output can be combined linearly to generate arbitrary spatial patterns. However, the spatial resolution is limited by the number of SPAD pixels (Eq. 3), and is too low for our current setup to generate high-resolution structured illumination patterns.

Incoherent Illumination

To gain further control over the synthesized illumination, we can improve spatial resolution by sacrificing temporal coherence. This is accomplished by optimizing the projector function over both $\vec{x}_p$ and ω , to focus to a point, $\vec{x}_c^{(2)}$, in the hidden scene. We modify Eq. S3 to generate a matrix $\mathcal{A}_*$:

$$\mathcal{A}_* \left(\vec{x}_c^{(2)}, \vec{x}_p, \omega, t' = t_i \right) = \int [\mathcal{P}(\omega) H(\vec{x}_p, \vec{x}_c, \omega)] e^{-j \frac{\omega}{c} (|\vec{x}_c^{(2)} - \vec{x}_c|)} e^{j \omega t_i} d\vec{x}_c \quad (\text{S13})$$

Intuitively, $\mathcal{A}_*(\vec{x}_c^{(2)}, \vec{x}_p, \omega)$ stores the light received at each hidden voxel $\vec{x}_c^{(2)}$ from each virtual source $\vec{x}_p$ at frequency ω and time t_i . Although we compute $\mathcal{A}_*$ over the entire volume, we later isolate values corresponding to surface voxels $\vec{x}_c^{(2)} \in \mathcal{S}$ to reduce memory and computation in subsequent steps.

We reshape $\mathcal{A}_*$ such that the i^{th} column of $\mathcal{A}_*$ is the reconstructed surface for a given illumination position $\vec{x}_p$, and frequency ω . Let $\mathcal{B}$ be a pattern such that

$$\mathcal{B}(\vec{x}_v) = \begin{cases} 1, & \vec{x}_v = \vec{x}_p^{(2)} \\ 0, & \text{otherwise} \end{cases} \quad (\text{S14})$$

Now, we can compute the complex coefficients $\mathcal{X}_*$:

$$\mathcal{X}_* = (\mathcal{A}_*^T \mathcal{A}_* + \lambda_R I)^{-1} \mathcal{A}_* \mathcal{B} \quad (\text{S15})$$

where the i^{th} entry of $\mathcal{X}_*$ is a complex coefficient, $\alpha(\vec{x}_p, \omega, \vec{x}_p^{(2)})$, corresponding to the i^{th} column of $\mathcal{A}_*$. We can define a new projector function:

$$\mathcal{P}(\vec{x}_p, \omega) = \alpha(\vec{x}_p, \omega, \vec{x}_p^{(2)}) \mathcal{P}(\omega) \quad (\text{S16})$$

Intuitively, $\mathcal{X}_*$ determines how to modulate the amplitude and phase of each SPAD pixel at each frequency so that their combined wavefront constructively interferes at the target location. Unlike the coherent case, each frequency component is optimized independently, allowing finer spatial control and enabling the formation of sharper focal spots—at the cost of losing temporal coherence. This frequency-specific shaping yields higher-resolution illumination patterns.

We then generate the desired pattern on the surface by incorporating this modified projector function into Eq. S3:

$$\begin{aligned} I_{\mathcal{P}} \left(\vec{x}_c^{(2)}, t' = t_i \right) &= \iiint \left[\alpha(\vec{x}_p, \omega, \vec{x}_p^{(2)}) \mathcal{P}(\omega) H(\vec{x}_p, \vec{x}_c, \omega) \right] \\ &\quad e^{-j \frac{\omega}{c} (|\vec{x}_c^{(2)} - \vec{x}_c|)} e^{j \omega t_i} d\vec{x}_p d\vec{x}_c d\omega. \\ &= \phi(\vec{x}_c^{(2)}, \vec{x}_p^{(2)}) \end{aligned} \quad (\text{S17})$$

This generates an incoherent point light source at $\vec{x}_c^{(2)} = \vec{x}_p^{(2)}$ with spatial resolution that is an order of magnitude better than that of a coherent point light source. To synthesize arbitrary patterns, we decompose the pattern into multiple $\vec{x}_{p_i}^{(2)}$ and repeat the computation for Eq. S13 – S17. The matrix $\mathcal{A}_*$ is large, and must be regenerated if t_i varies across $\vec{x}_p^{(2)}$, making this process computationally expensive (see Section S8). The final pattern, $I_{\mathcal{P}}(\vec{x}_v)$, can be computed using:

$$I_{\mathcal{P}}(\vec{x}_c^{(2)}) = \sum_i \eta_i \phi(\vec{x}_c^{(2)}, \vec{x}_{p_i}^{(2)}) \quad (\text{S18})$$

where $\eta_i = \left| \phi \left(\vec{x}_c^{(2)} v = \vec{x}_{p_i}^{(2)}, \vec{x}_{p_i}^{(2)} \right) \right|^{-1}$ is a normalization constant that improves visualization.

S4.3 Separation of Direct and Indirect Components

With the TLTM, the direct and indirect components can be separated efficiently with computational time gating on the temporal dimension. For each voxel, x_v , we can compute the time of flight, $t_i = |x_v - \bar{x}_p|/c + K$, for the virtual illumination to reach x_v from the center of the SPAD array $\bar{x}_p$, i.e., the center of the illumination pulse. We add a constant $K = \sigma$ from Eq. S5, to account for the temporal width of the illumination pulse as it propagates in time. Then, the direct component is the sum of video frames that occur before this time t_i , whereas the indirect component is obtained by summing frames that occur after t_i :

$$I_{\mathcal{D}}(\vec{x}_{v_p}, \vec{x}_{v_c}, t) = \int M_1(t) I_{\mathcal{P}}(\vec{x}_{v_p}, \vec{x}_{v_c}, t) dt \quad (\text{S19})$$

$$I_{\mathcal{I}}(\vec{x}_{v_p}, \vec{x}_{v_c}, t) = \int M_2(t) I_{\mathcal{P}}(\vec{x}_{v_p}, \vec{x}_{v_c}, t) dt \quad (\text{S20})$$

where $M_2(t)$ is the complement of $M_1(t)$, and $M_1(t)$ is given by:

$$M_1(t) = \begin{cases} 1, & t \leq t_i \\ 0, & t > t_i \end{cases} \quad (\text{S21})$$

We note that $M_1(t)$ and $M_2(t)$ need to be computed only once per hidden scene with $O(N^3)$ complexity, where N is the side length of the reconstructed volume. In Section S3.1, we discuss how our algorithm has significantly lower complexity than previous work¹⁴.

S5 TLTM-2 Validation and Performance Analysis

We validate TLTM-2 extraction using controlled simulations, comparisons with experimental data, and analyses of the synthesized spatial focus. These results assess both the accuracy of the reconstruction pipeline and the factors that govern spatial localization in the extracted TLTM-2.

S5.1 Simulation-to-Experiment Comparison

To assess real-world consistency and validate our method, we compare a simulated and experimental reconstruction of a $1.25 \text{ m} \times 1.25 \text{ m}$ planar surface at $z = 1.6 \text{ m}$. The simulation uses a uniformly spaced 16×16 SPAD grid that roughly matches the spatial extent of the 16×16 experimental SPAD grid, which is non-uniform and has 40 inactive pixels (see Fig. 6). Each timebin is 10 ps in simulation compared to 9.375 ps in experiments. Both datasets exhibit similar focal spot locations and shapes for the direct component of the reconstructed TLTM-2 column (see Row 2 in Fig. S1). Minor differences — such as tilt, residual background, and a $\sim 0.03 \text{ ns}$ temporal shift ($\sim 0.9 \text{ cm}$) — are attributed to hardware limitations like inactive pixels and non-uniform pixel spacing. Nonetheless, the agreement in focusing behavior affirms the robustness of our reconstruction pipeline.

Using this simulation, we find that focusing on a 3D location $\vec{x}_v$ using Eq. 5 introduces residual illumination even after $\vec{x}_v$ has been illuminated by the main beam. These artifacts, which we refer to as *temporal sidelobes*, appear as spurious light localized at $\vec{x}_v$ and persist for some time after the main peak. They arise from the finite spatial and temporal resolution of our reconstruction and decay slowly over time. While simulation results show smooth and consistent sidelobe profiles, the experimental results exhibit additional speckle-like variations due to measurement noise (see Row 3 in Fig. S1). Although sidelobes are typically two to three orders of magnitude weaker than the main peak, their intensity can overlap with that of multiply scattered light, making it challenging to disentangle true higher-order scattering from reconstruction artifacts. We provide further analysis and propose suppression strategies for these effects in Supplement Section S6.

S5.2 Simulation-Based Validation of Spatial Resolution

Having established consistency between simulation and experiment, we next use synthetic datasets to systematically analyze how scene depth, phasor field wavelength, and illumination aperture affect the spatial resolution of TLTM-2 extraction.

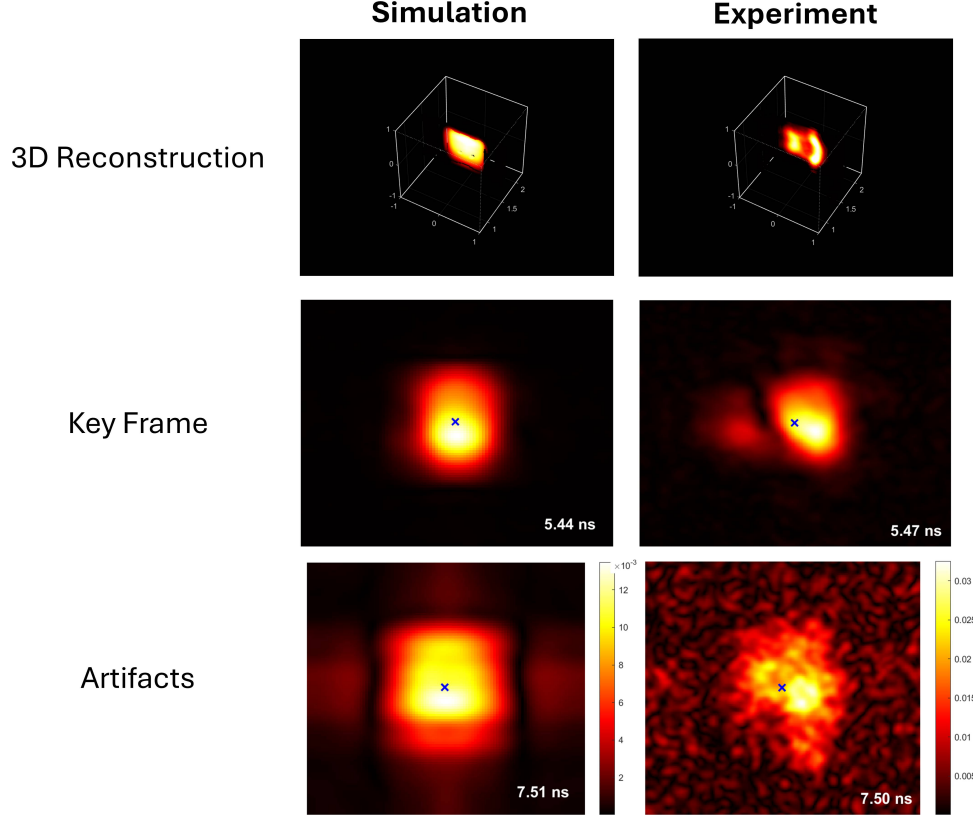


Fig. S1 Comparison between simulation (Column 1) and experiment (Column 2) for a hidden scene consisting of a planar surface. Row 1 shows the 3D reconstruction of the hidden scene. Row 2 displays the key transient frame from a TLTM-2 column when the virtual illumination is focused at the center of the plane. Row 3 shows a later frame that exhibits beamforming artifacts, which we refer to as *temporal sidelobes*. All frames are normalized to the maximum value of the reconstructed TLTM-2 column, which occurs in the key frame shown in Row 2.

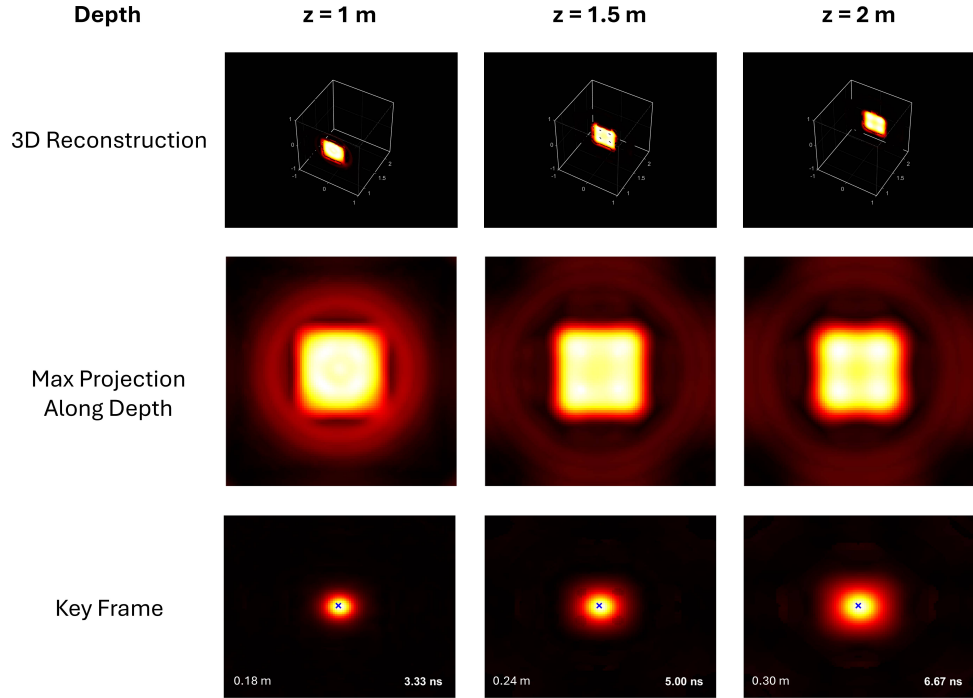


Fig. S2 We demonstrate using simulated datasets that the focal spot size increases as the same hidden plane is placed at greater depths behind the relay surface. Row 1 shows the 3D TLTM-2 reconstruction. Row 2 presents a 2D projection of the reconstruction along the depth axis using a maximum filter. Row 3 displays the transient frame corresponding to the moment when virtual illumination is focused onto the target location in the hidden scene. The time-of-flight (in nanoseconds) is indicated in the bottom right, while the estimated FWHM (in meters) is shown in the bottom left.

Since ground truth TLTM-2 is difficult to obtain in experiments, we first validate our approach using synthetic datasets generated with a transient renderer developed by Royo et al.^{15,16}. These simulations allow direct comparison between the reconstructed TLTM-2 columns and the known light transport in the hidden scene. Each synthetic TLTM-1 was rendered using a transient path tracer with (50×50) illumination points and (100×100) detection points. Both grids are uniformly sampled over a $2 \text{ m} \times 2 \text{ m}$ planar relay surface. The number of temporal bins varies with scene depth, and each bin has a duration of 10 ps.

We simulate a $1 \text{ m} \times 1 \text{ m}$ diffuse plane at depths $z = 1, 1.5$, and 2 m from the relay wall. Illumination is synthetically focused to the center of the plane using a fixed phasor field wavelength ($\lambda = 8 \text{ cm}$), and a single column of TLTM-2 is extracted. In Fig. S2, we present the reconstructed planes (Rows 1 and 2) along with the key transient frame (Row 3) when the virtual illumination focuses onto the plane. The resulting focal spot increases linearly with depth, as predicted by the Rayleigh criterion (Eq. 2). We quantify this spot size by computing the full width at half maximum (FWHM) values, which are measured to be 0.18 m, 0.24 m, and 0.30 m at 1 m, 1.5 m, and 2 m depths, respectively.

Next, we fix the hidden plane at $z = 1.5 \text{ m}$ and analyze how the spatial resolution of TLTM-2 varies with phasor field wavelength and illumination aperture. As shown in Fig. S3(A), increasing the phasor field wavelength from $\lambda = 4 \text{ cm}$ to 8 cm and 16 cm results in a linear increase in focal spot size, with measured FWHM values of 0.12 m, 0.24 m, and 0.48 m, respectively. This behavior is consistent with the Rayleigh criterion (Eq. 2), which predicts that resolution degrades with increasing wavelength. Similarly, reducing the illumination aperture to one-third and one-quarter of its full area produces broader focal spots, with FWHM values increasing from 0.24 m to 0.34 m and 0.42 m, respectively, as demonstrated in Fig. S3(B). These results further support the expected inverse relationship between aperture size and spatial resolution.

S5.3 Improving Spatial Focus for TLTM-2 Extraction

To improve the quality of the generated TLTM-2, we use the grid interpolation and side-lobe suppression strategies described in the Methods. The current bottleneck in our system is the limited number of SPAD pixels, which constrains the spatial focus of virtual illumination. While results in Fig. 3B utilize the full 16×16 SPAD array, we isolate a 1×16 row of our SPAD array and focus to a point on the back wall in the hidden scene, marked by a blue cross in each reconstructed image in Fig. S4. The dataset used to generate this figure is captured with an average SPAD pixel spacing of 3 cm, imposing a Nyquist phasor field wavelength of 6 cm. In this case, the linear array is able to focus to a vertical line near the blue cross at the back wall (Fig. S4B, Column 1). Grid interpolation reduces pixel spacing and enables the use of a 3 cm wavelength. The narrower vertical line demonstrates the improvement in spatial resolution, but interpolation on noisy data causes speckle-like artifacts in the reconstruction (Column 2). Although increasing exposure time reduces these artifacts, it is not always feasible.

In the top row of Fig. S4, we visualize different projector functions applied to the virtual illumination. The blue crosses indicate points on the discrete grid of virtual illumination positions. The orange line shows the spatially varying time delays applied across the virtual relay wall, which shape the curvature of the outgoing wavefront. The green curve represents the temporal waveform of the illumination pulse.

Columns 1–3 in Fig. S4 show methods for improving TLTM-2 extraction using temporally coherent illumination. A different way to improve the quality of TLTM-2 with limited exposure times is to solve an additional linear inverse to reduce sidelobes in our virtual illumination. To prevent the method from overfitting to the noise in the data, we split the dataset into two sets — the training set is used to generate optimal projector function coefficients Eqs. 9 – 10, while the test set is used for generating all the results displayed in Fig. S4. Row 1 shows the keyframe at the time when the illumination focuses to the blue cross, while Row 2 displays output after summing over all frames. Optimizing the spatial wavefront mainly reduces the out-of-focus illumination on the back wall and W (Column 3, Fig. S4) in both the keyframe (Column 3, Row 1) and sum over all frames (Column 3, Row 2) when compared to adding a spherical curvature to focus (Column 1). Additionally, this also reduces the residual illumination discussed in the previous section. Column 4, shown here for comparison, uses a spatiotemporal optimization method that sacrifices temporal coherence to achieve sharper spatial focusing. Although this method cannot be used to extract TLTM-2, it provides the higher spatial resolution needed with alphanumeric patterns, as shown in Fig. 5C.

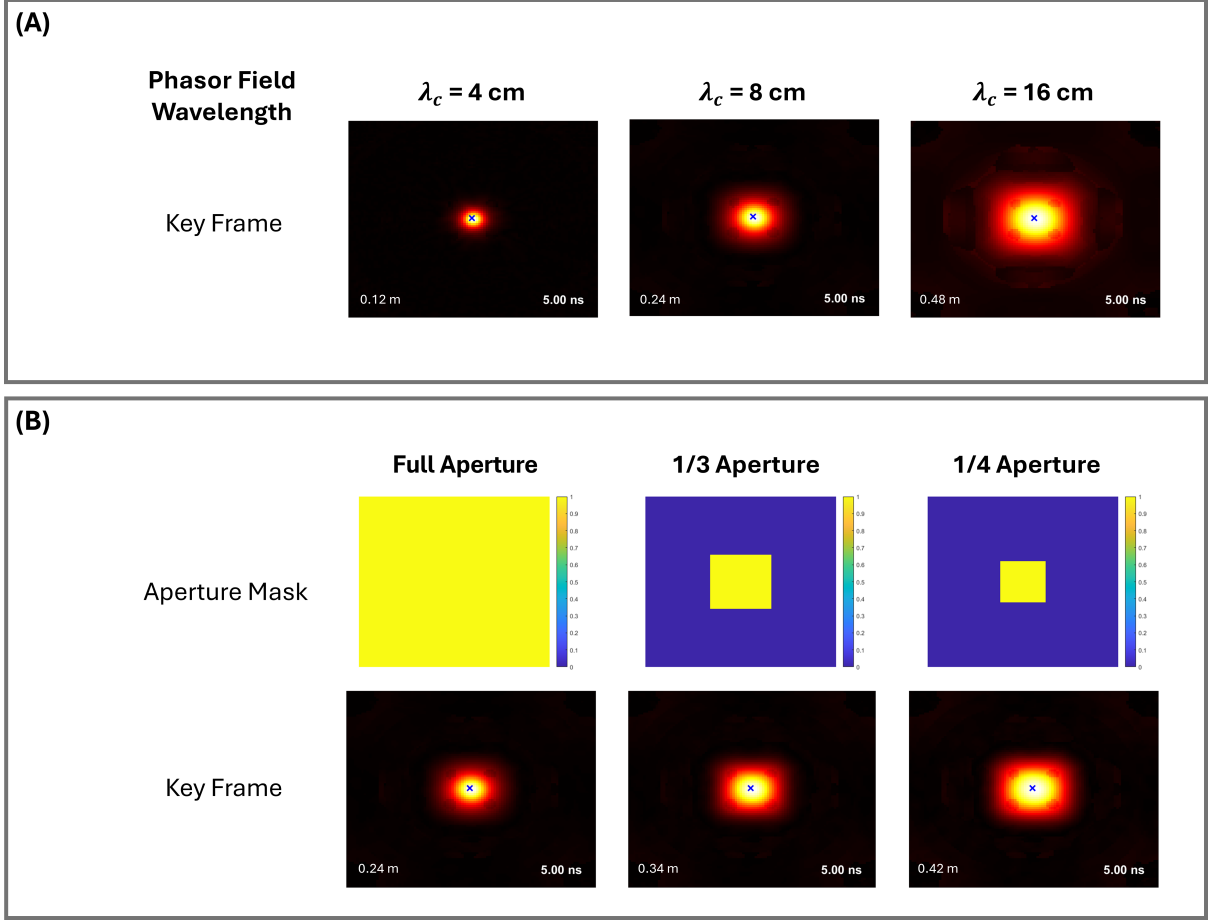


Fig. S3 Simulated analysis of focal spot size for a fixed hidden plane at $z = 1.5 \text{ m}$. **(A)** Increasing the phasor field wavelength leads to a larger focal spot, consistent with Rayleigh-based resolution limits. **(B)** Reducing the virtual illumination aperture similarly degrades spatial resolution. Row 1 shows the aperture mask; Row 2 shows the key transient frame when virtual illumination is focused on the target location. In both panels, time-of-flight (ns) is shown in the bottom right and FWHM (m) in the bottom left.

S6 Artifacts and Mitigation in TLTM-2 Extraction

We next analyze two artifact classes that affect TLTM-2 extraction: spatial artifacts introduced by voxel selection and temporal artifacts introduced by finite beamforming resolution. We then evaluate practical strategies for mitigating the latter in the current system.

S6.1 Max Filter Artifacts

One notable artifact is the discontinuity visible across the letter W in Fig. 1 and Fig. 3. This discontinuity can be traced to our voxel selection strategy: a max filter is applied during geometry initialization to localize likely surface voxels. This causes abrupt spatial transitions in the extracted TLTM-2 columns, particularly when the focused virtual illumination overlaps multiple surfaces at different depths or if the maximum occurs at an incorrect depth. To further investigate this, we compare reconstructions using the max filter against those using ground truth surface depths in simulation (Fig. S5). The max filter produces broader focal spots and introduces visible spatial discontinuities, even in noise-free settings, confirming its role as a source of reconstruction artifacts. These artifacts can be reduced by using slower, more accurate iterative 3D reconstruction methods and replacing the max filter with alternatives like centroiding or matched filtering for smoother voxel selection.

S6.2 Temporal Sidelobes

A second class of artifacts we examine are *temporal sidelobes*—residual illumination that persists at the focal point after the virtual wavefront has passed. These sidelobes arise from the system’s finite spatial and temporal resolution and are evident in both simulated and experimental TLTM-2 reconstructions.

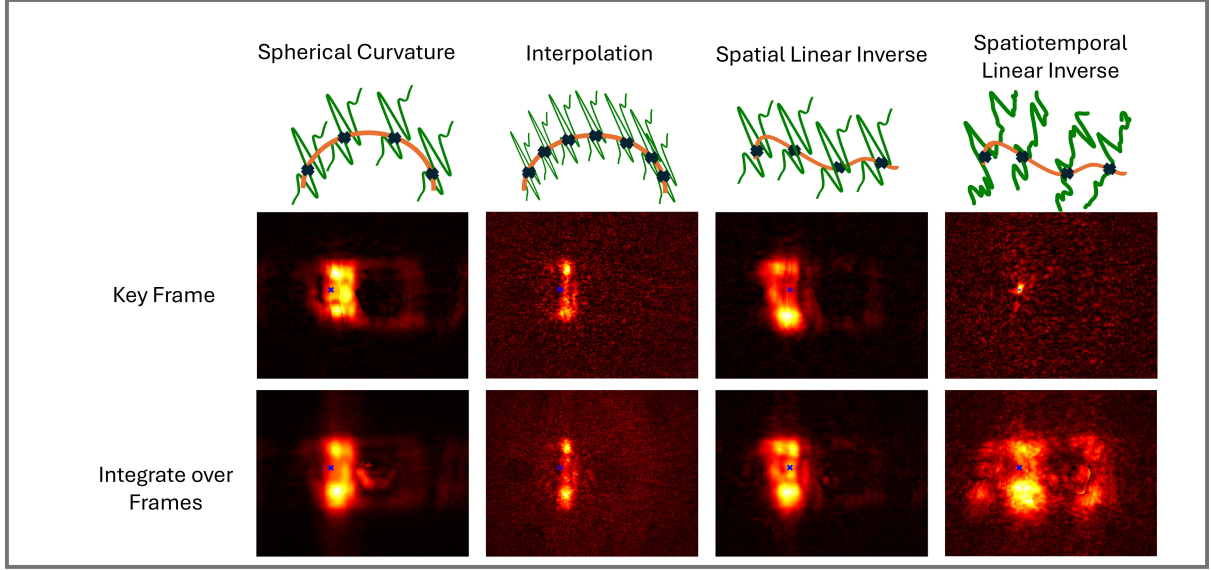


Fig. S4 (A) Column 1 applies spherical curvature to create a line-shaped focal spot. Column 2 interpolates the illumination grid to increase resolution enabling the use of a shorter phasor field wavelength and improving spatial focus relative to Column 1. Column 3 uses a spatial inverse solution to suppress sidelobes and further reduces out-of-focus illumination compared to Column 1. Columns 1–3 use temporally coherent illumination and are compatible with TLTM-2 extraction. Column 4 applies a spatiotemporal inverse to synthesize an incoherent point source with significantly improved spatial resolution, but at the cost of temporal coherence, making it unsuitable for TLTM-2 extraction. This method is included for comparison and is used to project the letter N in the pattern-projection result shown in main-text Fig. 5C.

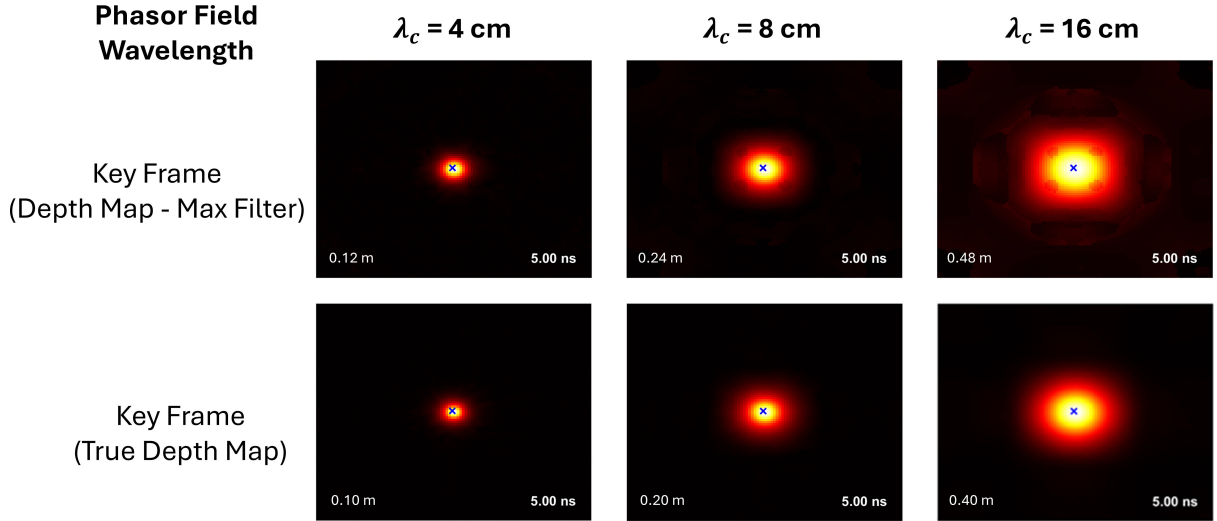


Fig. S5 Simulated TLTM-2 reconstructions at different phasor wavelengths (4 cm, 8 cm, 16 cm) using max-filtered depth (Row 1) vs. ground truth depth (Row 2). Time-of-flight (ns) is shown bottom right; focal spot FWHM (m) is bottom left. Max filtering yields broader spots (0.12, 0.24, 0.48 m vs. 0.10, 0.20, 0.40 m) and visible discontinuities, highlighting depth selection artifacts even in noise-free, simulated settings.

In simulations, they exhibit a smooth and consistent decay profile, while in experimental results they show additional speckle-like variation due to measurement noise (see Fig. S1).

Our simulations confirm that sidelobes are spatially localized at the focal point and decay slowly over time (Fig. S6). Notably, their spatial extent varies from $1.3\times$ to $1.5\times$ that of the focal spot. We observe that increasing the phasor field wavelength or decreasing the aperture size leads to broader focal spots and larger sidelobes. As discussed in Eq. 3 of the main manuscript, this focal spot width is governed primarily by the number of SPAD pixels (N) and the depth of the hidden voxel (z), assuming the phasor field wavelength satisfies the Nyquist condition ($\lambda_c = 2\lambda_s$). As shown in Fig. S6(B), increasing the aperture size—i.e., the number and spatial coverage of SPAD pixels—leads to smaller sidelobes and, as discussed in Supplementary Section S3.1, this improvement does not increase computational complexity.

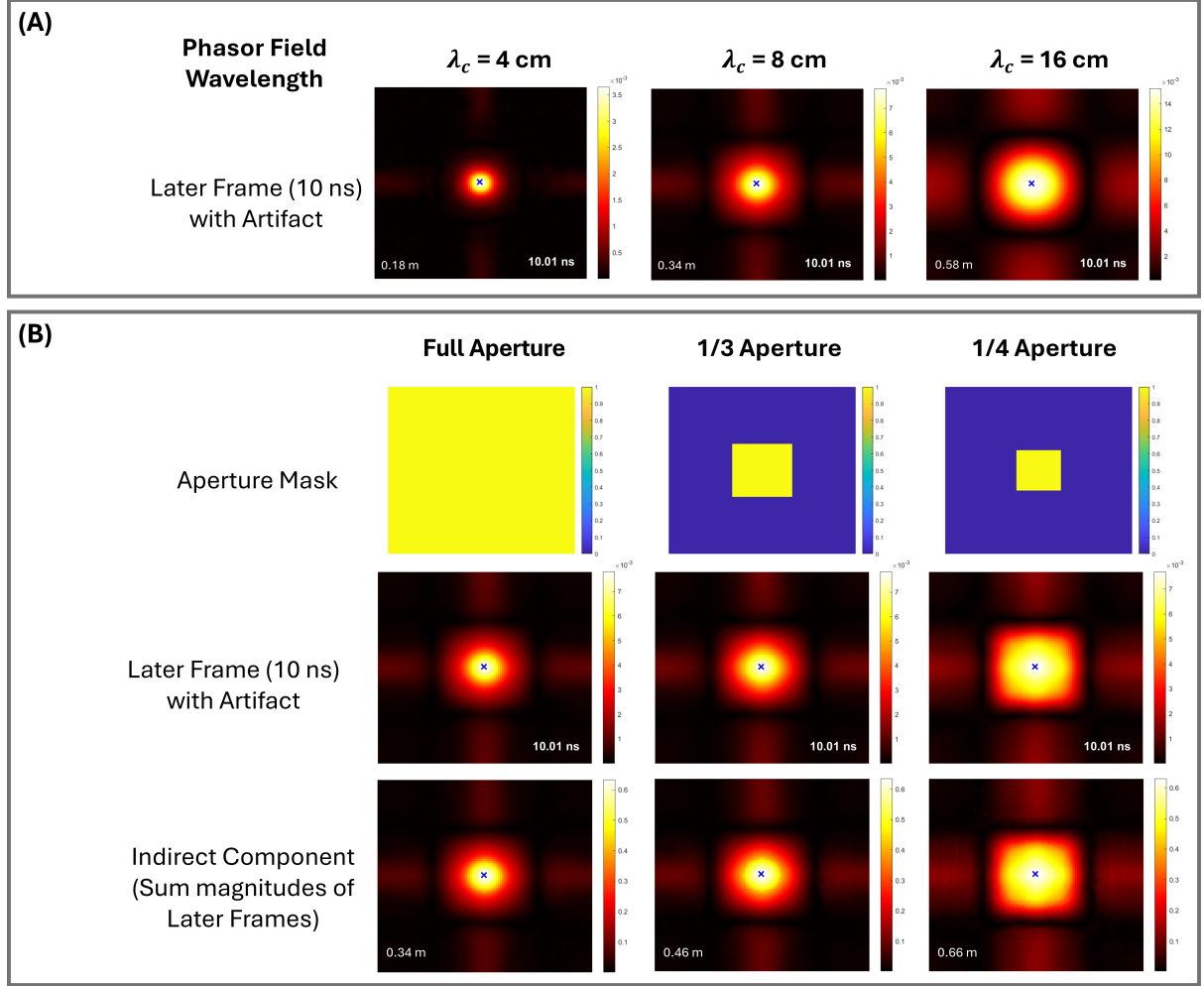


Fig. S6 Simulated analysis of temporal sidelobes spot size for a fixed hidden plane at $z = 1.5 \text{ m}$. Reconstructed intensity is normalized to the max value in the corresponding key frame in Fig. S3. **(A)** Increasing the phasor field wavelength leads to larger sidelobes. **(B)** Reducing the virtual illumination aperture similarly increases spatial extent of temporal sidelobes. Row 1 shows the aperture mask; Row 2 shows a snapshot of sidelobes at 10 ns. Row 3: Indirect component summed over frames after the key frame, showcasing that artifact shape and size remain stable over time.

S6.3 Mitigating Temporal Sidelobes

To address the temporal sidelobes described above, we explore two practical mitigation strategies that operate within the constraints of current SPAD array resolution. First, a temporal-domain filtering technique suppresses slow-varying components, which are characteristic of sidelobes but not of true time-resolved transport. Second, we replace the standard delay-and-sum beamformer with a Tikhonov-regularized inverse (see Eqs. 7 — 10), which directly optimizes the projector wavefront to focus energy while minimizing residual illumination.

While neither approach fully eliminates sidelobes, both offer limited improvements and validate that artifacts arise from algorithmic and hardware limitations rather than scene content. However, these approaches are limited by the underlying spatial resolution and should be viewed as interim solutions. As shown in our simulations, larger and denser SPAD arrays will more effectively localize temporal sidelobes—reducing their spatial extent without increasing computational cost (see Fig. S6 and Supplementary Section S3.1). In the longer term, these hardware improvements could be complemented by advanced spatiotemporal regularization or learning-based reconstruction techniques.

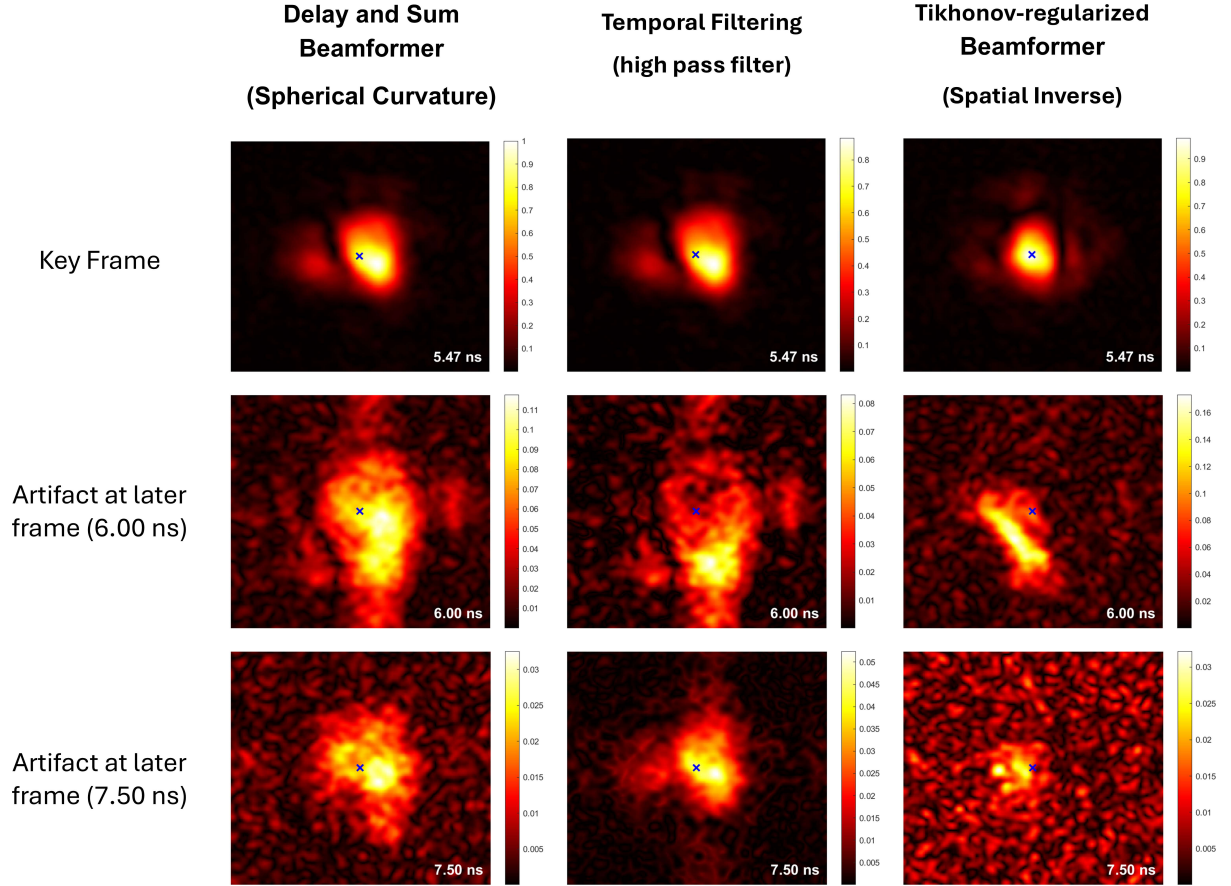


Fig. S7 Suppression of temporal sidelobes using alternative beamforming strategies on experimental data for a $1.25 \text{ m} \times 1.25 \text{ m}$ plane at $z = 1.6 \text{ m}$. Column 1: Standard delay-and-sum beamformer with spherical curvature. Column 2: Temporal filtering removes low-frequency components characteristic of sidelobes. Column 3: Tikhonov-regularized inverse beamformer optimizes the projector wavefront to reduce residual illumination. Both methods reduce the spatial extent of the temporal sidelobes, though they may increase sidelobe intensity at certain locations. All intensities are normalized to the maximum value of the key frame. Row 1 showcases key frames where virtual illumination focuses onto the plane; later rows visualize sidelobe artifacts at subsequent times.

S7 Noise and Temporal Resolution Limits

We next examine two additional factors that limit the practical utility of TLTM-2 extraction in our current system: measurement noise and the temporal resolution of the synthesized phasor-field illumination. The former determines the exposure required to recover weak higher-order transport, while the latter sets the minimum temporal scale over which distinct transport events can be separated.

S7.1 TLTM-2 SNR Analysis

To analyze robustness under photon-limited conditions, we simulate reduced-exposure acquisitions by subsampling a 45-minute TLTM-1 capture and reconstructing TLTM-2 at varying exposure durations (90–1090 s). PSNR grows approximately linearly with $\log_2(\text{exposure})$, gaining approximately 3.9 dB per doubling of exposure time in seconds.

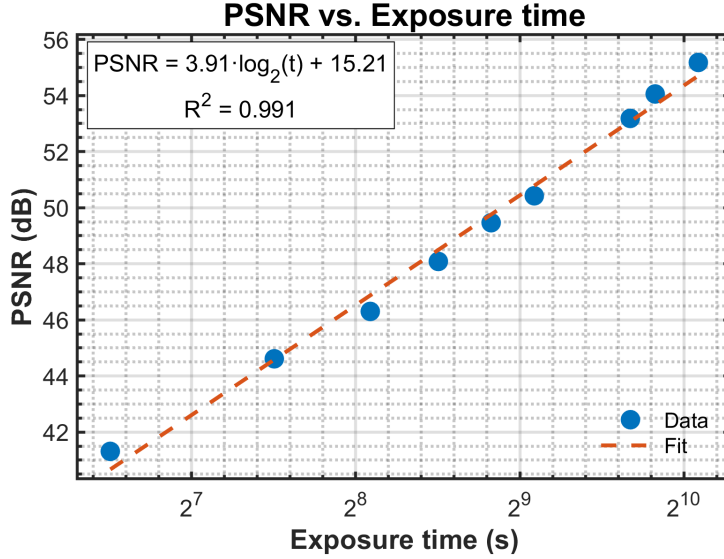


Fig. S8 We plot the PSNR as a function of exposure for a single column of TLTM-2. The ground truth reference is set to a 45 minute long exposure.

While global PSNR captures overall reconstruction quality, it does not distinguish between direct and indirect transport. To illustrate this, we plot and examine individual TLTM-2 frames at varying exposures in Fig. S9. Direct components—such as the letter n lighting up—appear clearly even at 90 s, while indirect components—such as reflection of the letter n from a mirror—require significantly longer exposures (up to 450 s) to become discernible. We quantify this trend using SSIM, computed per frame relative to the 45-minute reference. Early frames (e.g., Frame 9) improve modestly (approximately 35%) from 90 to 500 s, while later frames (e.g., Frame 68) show over 400% improvement, highlighting the greater exposure required to resolve multi-bounce light paths.

S7.2 TLTM-2 Temporal Resolution Limits

The temporal resolution of TLTM-2 is set by the characteristics of the phasor field projector, modeled as a modulated Gaussian pulse (see Eq. S5). It depends on both the carrier wavelength and the envelope width, which jointly define the temporal width of the virtual wavefront. In our system, we use a 7 cm carrier and a 28 cm envelope, yielding an effective resolution of 933 ps. Although the phasor field projector is synthetic, its temporal resolution is fundamentally limited by two constraints. First, the carrier wavelength must exceed twice the spatial sampling interval on the relay surface to avoid aliasing. Second, the resolution should not be finer than the temporal impulse response of the NLOS system used to measure TLTM-1—65 ps in our setup—since any sharper virtual pulse would be blurred by the system itself. While our current parameters balance resolution and SNR, adapting the pulse width to scene geometry or SPAD characteristics remains a promising direction for future work.

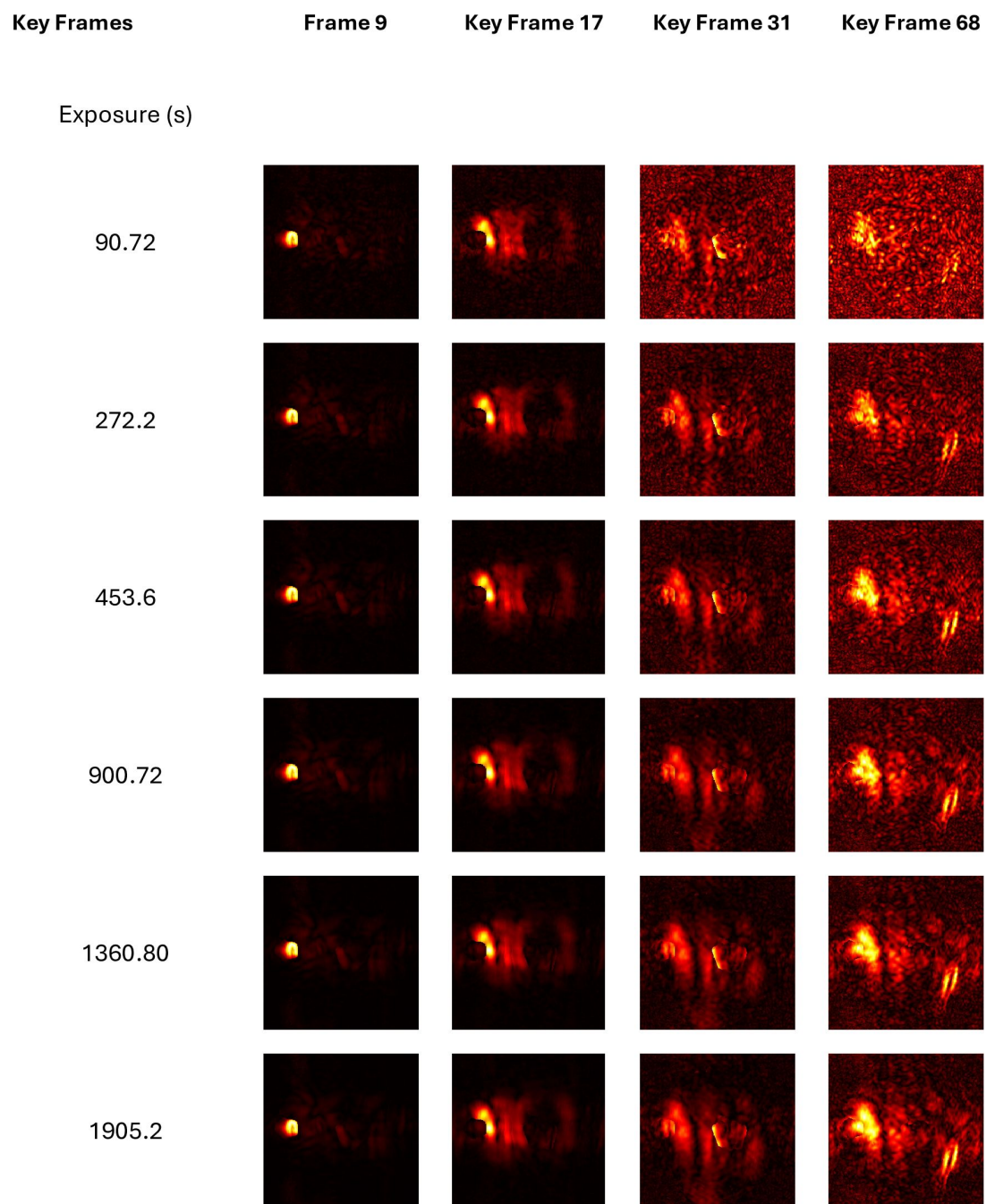


Fig. S9 Individual frames from a single column of TLTM-2, reconstructed at increasing exposure times. Showcasing different convergence times for direct and indirect light.

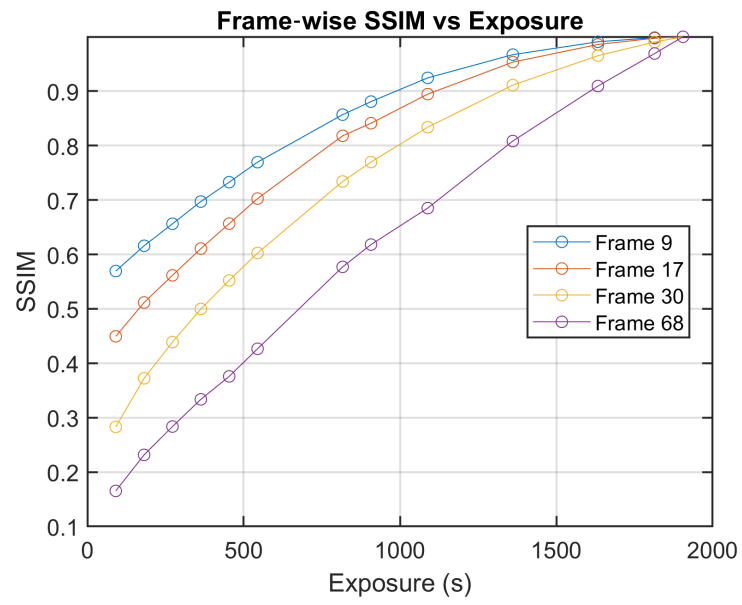


Fig. S10 SSIM per frame as a function of exposure time, using 45 minute exposure as ground-truth.

S8 Complexity for Optimizing Projector Functions

Let A denote an $n \times m$ matrix. In ridge regression, the total computational complexity is $O(nm^2 + m^3)$, where the dominant costs arise from forming the Gram matrix $A^\top A$ in $O(nm^2)$ time and inverting (or solving the linear system associated with) the resulting $m \times m$ matrix in $O(m^3)$ time. Let N denote the side length of the reconstruction volume, $M \times M$ denote the number of SPAD pixels, and F the number of frequency components we retain after applying the phasor field filter in Eq. S6. Suppose we aim to project a pattern that is decomposed into P points arriving at S unique times, where $S \leq P$. Then, both A and A_* need to be recomputed for S unique points, meaning the complexity of ridge regression in Eq. 9 and Eq. S15 also scales by S . When constructing the matrices A and A_* , we apply a max filter along depth to isolate N^2 voxels corresponding to surfaces in the hidden scene, onto which we can project the pattern. This also reduces computation by removing voxels with zero or small intensities.

The matrix A in Eq. 7 optimizes the projector function spatially across SPAD pixels and has size $N^2 \times M^2$, making the computational complexity for projecting S points to be $O(S(N^2M^4 + M^6))$. Similarly, matrix A_* in Eq. S13 optimizes the projector function over both space and temporal frequencies so the size of A_* is $N^2 \times M^2F$. The overall computational complexity of projecting S unique points onto surfaces in the hidden scene is therefore $O(S(N^2M^4F^2 + M^6F^3))$.

We note that both complexities scale in polynomial time with regard to M^2 , the number of SPAD pixels while the complexity for direct focusing using beamforming, given by $O(SkN^3 \log(N))$ does not. Since beamforming resolution also improves with the number of pixels, we anticipate that these optimization procedures will become unnecessary as SPAD array pixel counts increase.

S9 SNR

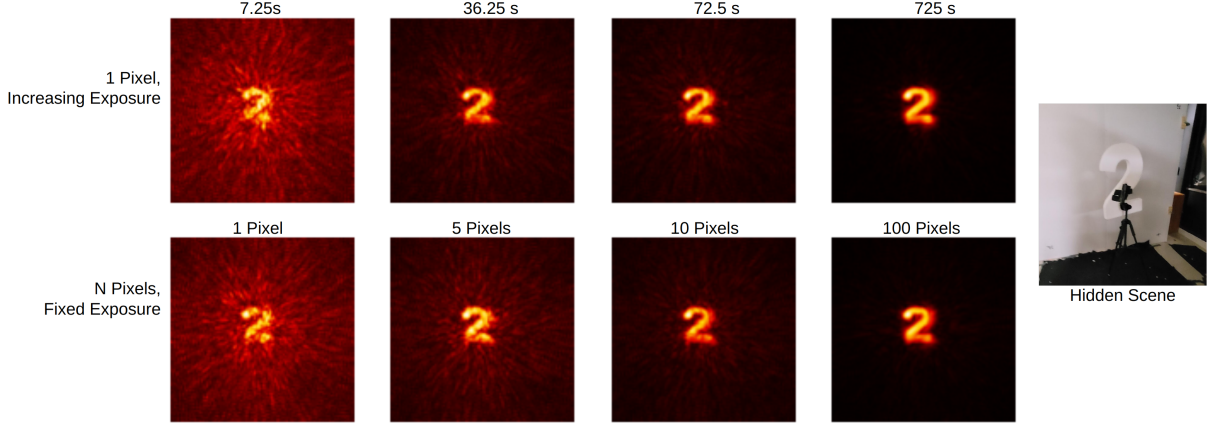


Fig. S11 Qualitative comparison of SNR. **Left, Top row:** Single-pixel reconstructions at progressively longer exposure times. **Left, Bottom row:** Reconstructions with a fixed 7.25s exposure time using an increasing number of SPAD pixels. **Right:** Hidden object

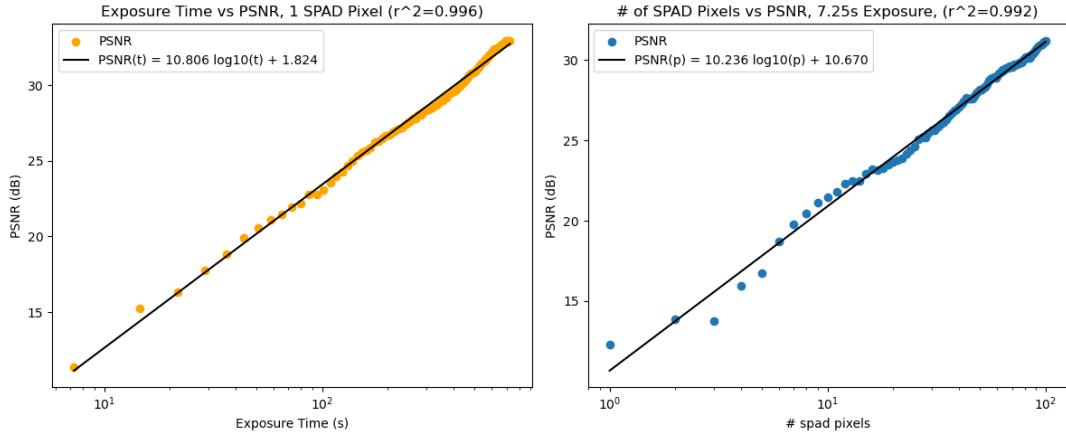


Fig. S12 Quantitative comparison of SNR, showcasing that the growth factor for PSNR is similar when increasing the exposure time (10.806) or the number of SPAD pixels (10.236).

A practical benefit of using a multi-pixel SPAD array is that it captures more photons per unit time than a single-pixel detector. This improves signal-to-noise ratio (SNR) without requiring proportionally longer exposure times, which is important for dynamic scenes and near real-time NLOS imaging⁶.

In transient imaging, SNR is largely determined by the number of detected photons. Shorter exposures collect fewer photons and therefore have higher relative Poisson noise, while longer exposures improve SNR by increasing the total photon count. Longer exposure times ameliorate the low-SNR problem because Poisson noise grows at a rate proportional to the square root of the number of photons detected. However, longer exposures are not always practical for NLOS imaging, particularly for dynamic scenes or applications requiring low-latency operation⁶.

The expectation is that for a given single-pixel exposure time, t_1 , an equivalent SNR can be achieved by instead using n pixels with an exposure time of $t_n = t_1/n$. We experimentally show that this expectation holds in our system. To achieve this trade-off, it is necessary to project the SPAD array onto a small area on the relay wall, ideally having dimensions of the same order as the phasor-field wavelength. By doing so, all pixels observe nearly the same perspective of the hidden scene and can be easily combined without introducing any additional blurring in the combined reconstruction. In our experimental data, all SPAD pixels were projected onto a rectangular area of 7.1×4.7 cm.

In our experiment, we constructed a simple scene containing a Lambertian number 2 cut from expanded polystyrene foam. We then captured 125 exposures of 7.25s each using all available SPAD pixels

on our array. Analysis was performed in the following manner: First, we selected a single pixel at random from the array. Using only photons collected by this pixel, we performed 100 phasor-field reconstructions having exposure times $7.25 \times n$ seconds for $n = 1 \dots 100$. Second, we selected a single 7.25-second exposure, and performed 100 phasor-field reconstructions, using n arbitrarily selected pixels for $n = 1 \dots 100$. As expected, we observed a linear increase in total detected photon count with both the increasing exposure and increasing pixels analyses. A sampling of the reconstructions for $n = \{1, 5, 10, 100\}$ are shown in Fig. S11.

For quantitative analysis, we used the widely accepted Peak Signal-to-Noise Ratio (PSNR) calculation as a metric to assess improvement in image quality with the addition of more photons, whether by increased exposure or increased number of pixels. The PSNR formula requires calculating the mean square error with respect to a noise-free image. For the noise-free image in our calculations, we used a reconstruction from 216 pixels of a 906.25 second exposure. The PSNRs of each of the 200 reconstructions were calculated and are shown in Fig. S12. A least-squares, best-fit exponential curve overlays each data set. Note that both methods of adding additional photons (longer exposure vs additional pixels) have nearly the same growth rate for the reconstructions' PSNR (0.186 vs 0.184). From this, we can conclude that when a tight spacing of SPAD pixels on the relay wall is used, capturing with multiple SPAD pixels is experimentally equivalent to capturing over a longer exposure time.

S9.1 Primal-Dual Equivalence

The photon time of flight is identical whether the photon travels from the illumination to the detection or from the detection to the illumination. Additionally, fast reconstruction utilizes Fourier transforms which require sampling the relay wall using a uniform grid. While the illumination and detection grids can be interchanged in principle, it is easier to generate a uniform laser grid since visual feedback can be used to focus the laser and the galvo can be used to scan the uniform grid sequentially. Conversely, the SPAD pixels must be focused simultaneously and the uniformity of the detection grid is ultimately limited by the focusing optics and the presence of hot pixels on the sensor, leading to distortions seen in Fig. 6. Since we don't have a uniform SPAD grid, we use a slow phasor field back projection algorithm to demonstrate the equivalence between the primal and dual images.

Fig. S13 shows the hidden scene (left) and the reconstruction (right) after adding up the reconstructions from all the SPAD pixels for the full laser grid. Next, we reconstruct the primal (left) and dual (right) images for a single laser and camera position respectively, and display them in Fig. S14. To get the SPAD and laser grids to match (Row 1, Fig. S14), the laser grid is downsampled leading to a loss of reconstruction resolution. Additionally, the laser positions outside the corners of the SPAD grid are discarded, which is why the bottom of the 2 and the T is missing in the reconstruction. The 2 in the laser grid reconstruction looks cleaner, but that may be due to the presence of hot pixels within the SPAD grid.

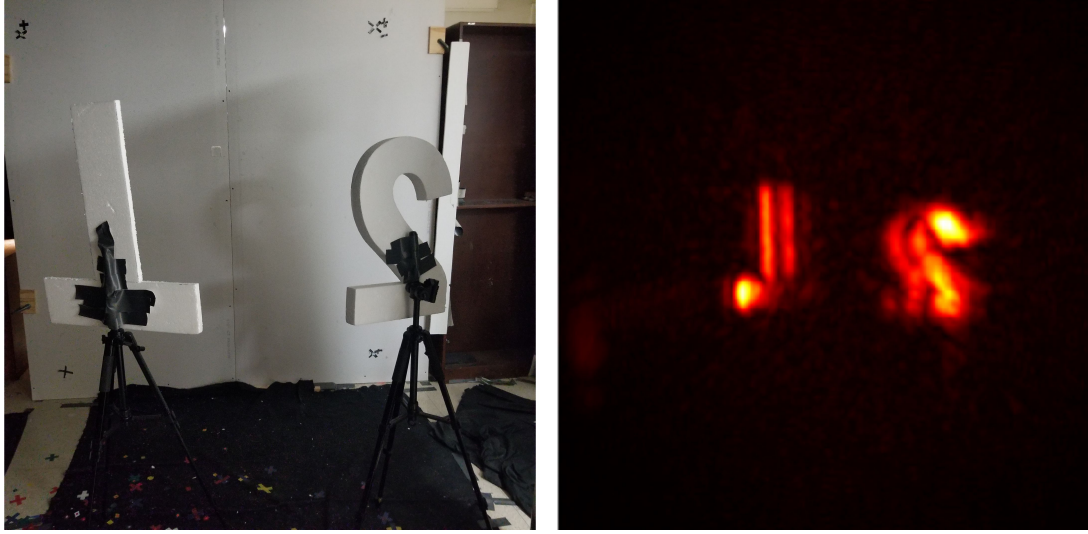


Fig. S13 Hidden scene shown on the left, and reconstruction from all SPAD pixels for the large field of view camera grid is shown on the right. Adding up all the contributions yields 3D shading and lighting effects for the hidden scene.

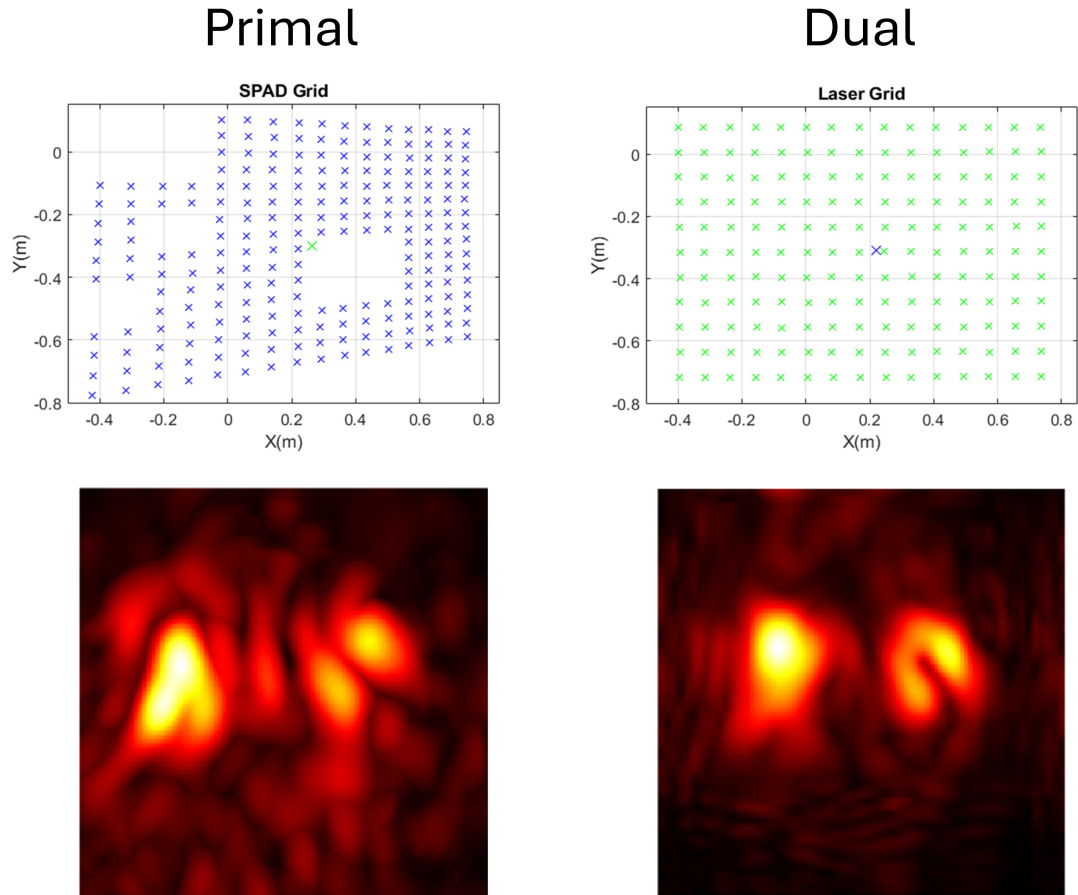


Fig. S14 Qualitative comparison of primal (column 1) and dual (column 2) images, where the dataset captured using the dual grid takes approximately 250x longer to acquire than that from the primal grid. Row 1 displays the two grids used, with camera positions marked in blue and laser positions marked in green. Row 2 displays the corresponding reconstruction.

S10 Supplementary Results: Direct-Indirect Separation of Light Transport

To better illustrate the separation of direct and indirect light transport, we introduce a corner dataset designed specifically to emphasize strong interreflections. This scene features two large planar surfaces arranged to form a corner in the hidden volume. Fig. S15 provides an overview of this setup. The left panel shows a smartphone image of the physical scene, the middle panel presents a top-down schematic of the experimental 3D geometry, and the right panel shows a 3D reconstruction of the hidden scene.

We then perform focused virtual illumination on each planar surface independently to isolate and analyze the resulting light transport components. As shown in Fig. S16, Column 1 corresponds to focusing on the left surface and Column 2 to the right surface, with a blue cross marking the virtual focal point. Rows 1 and 2 show top-down schematics of the scene: Row 1 illustrates the virtual illumination focusing onto the focal spot, and Row 2 depicts the expected light paths. Row 3 presents the reconstructed key frame at the time of focus. Rows 4 and 5 show the separated direct and indirect components, respectively, with the region of interest in the indirect component highlighted in green.

In both configurations, the indirect component is clearly observed at the expected location, validating the method’s effectiveness. However, we also observe a bright spot reconstructed near the focal point in both cases—visible in the indirect component (Row 5). This is a reconstruction artifact caused by *temporal sidelobes*, as described in Section S6.2. These sidelobes arise from the finite spatial and temporal resolution of the beamforming algorithm and do not indicate a failure to separate light paths.

Mitigation strategies are discussed in Section S6.3, where we show that increasing the spatial resolution of the virtual focus—by expanding the SPAD aperture or reducing the phasor field wavelength—reduces the spatial extent of temporal sidelobes. While the separation process itself remains unchanged, reducing these artifacts leads to cleaner direct and indirect components, thereby improving the overall quality and interpretability of the results. Future SPAD arrays with higher spatial resolution are expected to further enhance this effect.

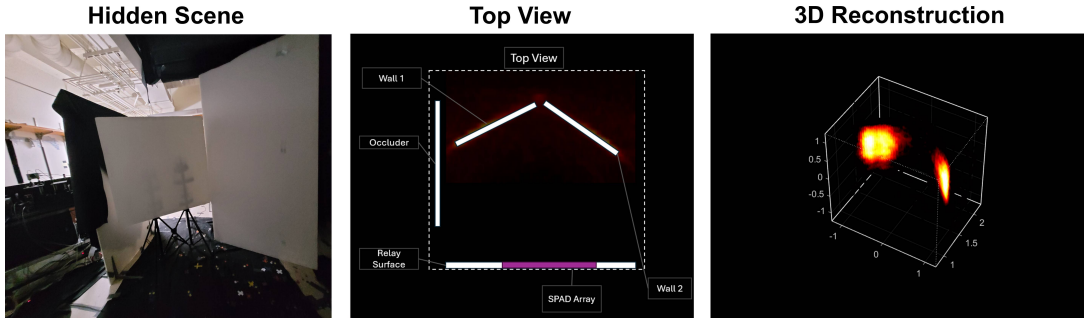


Fig. S15 Corner dataset used to demonstrate direct-indirect light separation. Left: Smartphone image of the physical scene showing two planar surfaces forming a corner. Middle: Top-down schematic of the experimental 3D geometry. Right: 3D reconstruction of the hidden scene, highlighting the recovered surfaces.

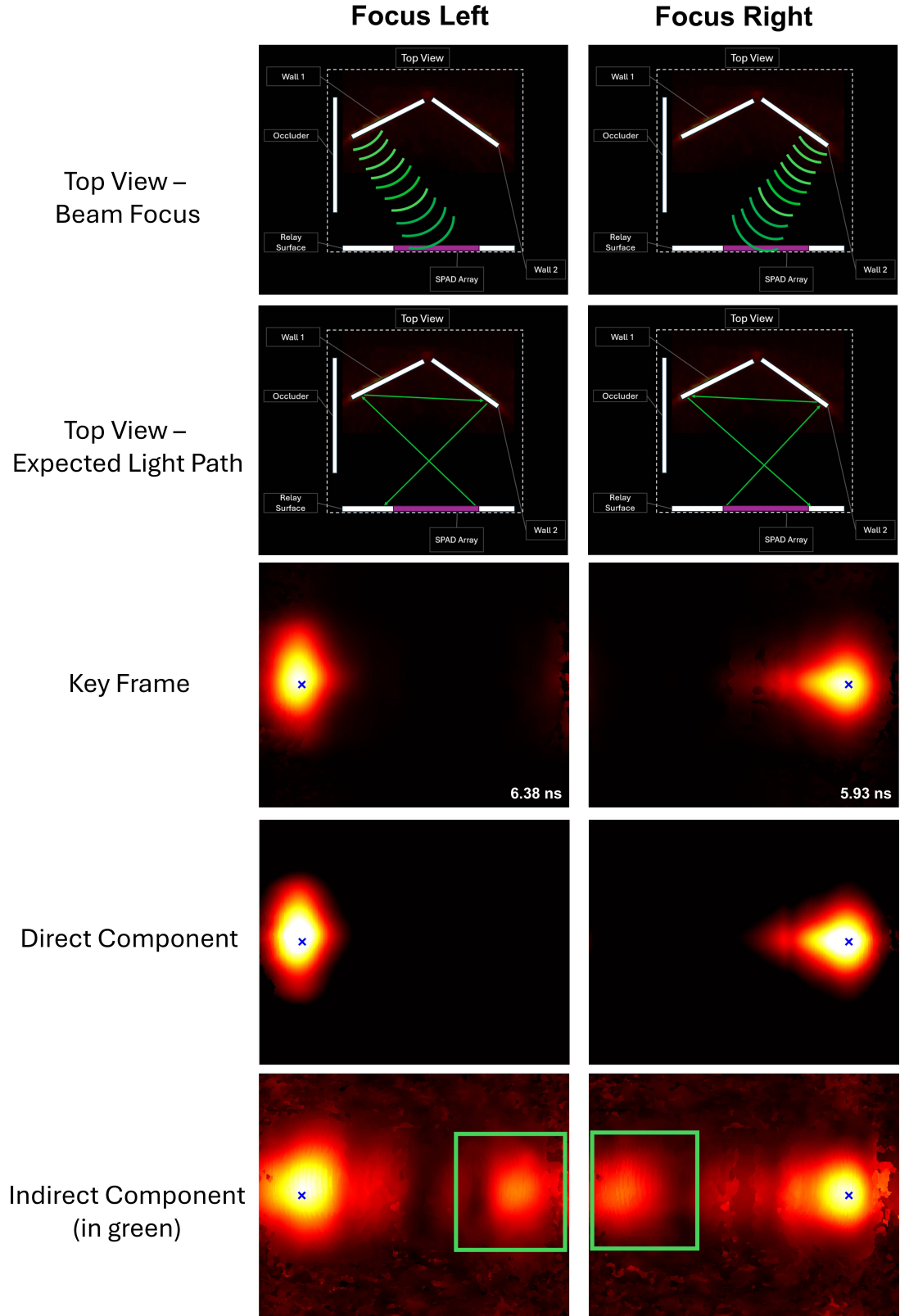


Fig. S16 Separation of direct and indirect light transport by focusing on the left planar surface (Column 1) and right planar surface (Column 2). A blue cross marks the virtual focal point in each image. Rows 1 and 2 show top-down schematics of the scene: Row 1 illustrates the virtual illumination focusing onto the focal spot, and Row 2 depicts the expected light paths. Row 3 presents the reconstructed key frame when the hidden surface is illuminated, with the frame time (in nanoseconds) indicated at the bottom right. Row 4 shows the separated direct component, while Row 5 shows the indirect component, with the region of interest highlighted in green. A bright residual near the focal point is also visible, resulting from reconstruction artifacts (temporal sidelobes).

References

- [1] Velten, A. *et al.* Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging. *Nature Communications* **3**, 745 (2012). URL <https://www.nature.com/articles/ncomms1747>.
- [2] Liu, X., Bauer, S. & Velten, A. *Analysis of Feature Visibility in Non-Line-Of-Sight Measurements*, 10132–10140 (IEEE, Long Beach, CA, USA, 2019). URL <https://ieeexplore.ieee.org/document/8953310/>.
- [3] Goodman, J. W. *Introduction to Fourier optics* 3rd ed edn (Roberts & Co, Englewood, Colo, 2005). OCLC: ocm56632414.
- [4] Liu, X. *et al.* Non-line-of-sight imaging using phasor-field virtual wave optics. *Nature* **572**, 620–623 (2019). URL <https://www.nature.com/articles/s41586-019-1461-3>. Number: 7771.
- [5] Liu, X., Bauer, S. & Velten, A. Phasor field diffraction based reconstruction for fast non-line-of-sight imaging systems. *Nature Communications* **11**, 1–13 (2020). URL <https://www.nature.com/articles/s41467-020-15157-4>. Number: 1.
- [6] Nam, J. H. *et al.* Low-latency time-of-flight non-line-of-sight imaging at 5 frames per second. *Nature Communications* **12**, 6526 (2021). URL <https://www.nature.com/articles/s41467-021-26721-x>.
- [7] Reza, S. A., La Manna, M., Bauer, S. & Velten, A. Phasor field waves: experimental demonstrations of wave-like properties. *Optics Express* **27**, 32587 (2019). URL <https://www.osapublishing.org/abstract.cfm?URI=oe-27-22-32587>.
- [8] Reza, S. A., Manna, M. L., Bauer, S., Velten, A. & Velten, A. Phasor field waves: A Huygens-like light transport model for non-line-of-sight imaging applications. *Optics Express* **27**, 29380–29400 (2019). URL <https://www.osapublishing.org/oe/abstract.cfm?uri=oe-27-20-29380>.
- [9] Teichman, J. A. Phasor field waves: a mathematical treatment. *Optics Express* **27**, 27500–27506 (2019). URL <https://www.osapublishing.org/oe/abstract.cfm?uri=oe-27-20-27500>.
- [10] Dove, J. J. M. *Theory of phasor-field imaging*. Ph.D. thesis, Massachusetts Institute of Technology (2020).
- [11] Sultan, T., Reza, S. A. & Velten, A. Towards a more accurate light transport model for non-line-of-sight imaging. *Optics Express* **32**, 7731–7761 (2024). URL <https://opg.optica.org/oe/abstract.cfm?uri=oe-32-5-7731>.
- [12] Schmerr, L. W. *Fundamentals of ultrasonic phased arrays* No. 215 in Solid mechanics and its applications (Springer, Cham Heidelberg, 2015).
- [13] Gu, C., Sultan, T., Masumnia-Bisheh, K., Waller, L. & Velten, A. *Fast Non-line-of-sight Imaging with Non-planar Relay Surfaces*, 1–12 (2023).
- [14] Marco, J. *et al.* *Virtual Light Transport Matrices for Non-Line-of-Sight Imaging*, 2420–2429 (2021).
- [15] Royo, D., García, J., Muñoz, A. & Jarabo, A. Non-line-of-sight transient rendering. *Computers & Graphics* **107**, 84–92 (2022). URL <https://www.sciencedirect.com/science/article/pii/S0097849322001200>.
- [16] Royo, D. *et al.* mitransient: Transient light transport in Mitsuba 3 (2025). URL <http://arxiv.org/abs/2510.25660>. ArXiv:2510.25660 [cs].