

ToF ReSTIR: Time-of-Flight Rendering with Spatio-temporal Reservoir Resampling

JUHYEON KIM, Dartmouth College, USA
 WOJCIECH JAROSZ, Dartmouth College, USA
 ADITHYA PEDIREDLA, Dartmouth College, USA

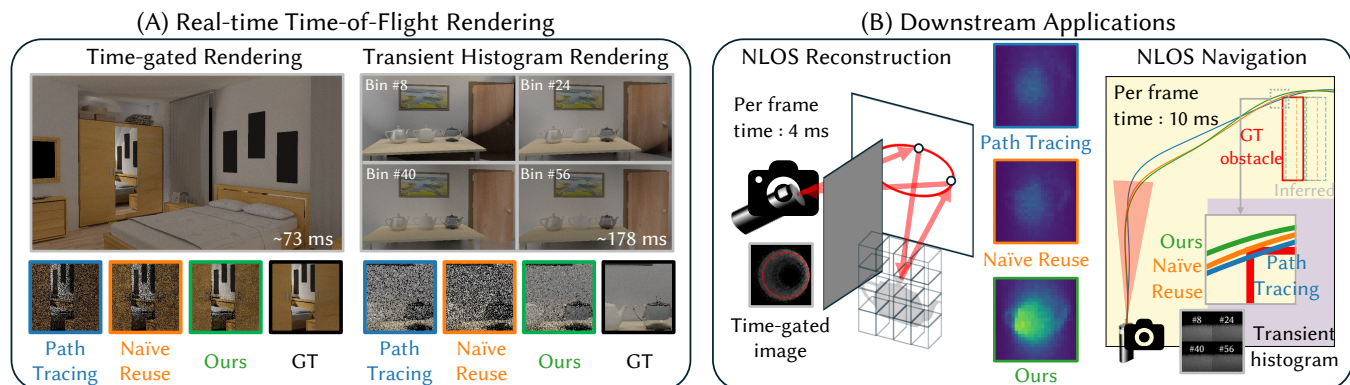


Fig. 1. We propose a real-time Time-of-Flight (ToF) rendering method inspired by ReSTIR, a state-of-the-art technique for steady-state real-time rendering, augmented with a novel path-length-aware shift mapping. (A) Compared to conventional path tracing and naive path reuse adapted from steady-state ReSTIR [Lin et al. 2022], our method produces improved visual results for both time-gated image rendering and transient histogram rendering in dynamic scenes at interactive frame rates. (B) We further demonstrate two interactive downstream applications enabled by our renderer. For non-line-of-sight (NLOS) reconstruction, we recover hidden voxel geometry using temporally focused imaging [Pediredla et al. 2019a], where time-gated image simulation with our method reliably reconstructs the shape of the teapot. We also simulate NLOS navigation tasks [Young et al. 2025], where the reduced noise in our rendered transient histograms enables robust neural network inference and reliable path planning, whereas baseline methods fail to produce safe paths due to noise.

We present a novel spatio-temporal reuse framework for time-resolved light transport, enabling efficient Monte Carlo rendering of time-of-flight (ToF) phenomena such as time-gated imaging and transient light capture. Existing ToF rendering methods are computationally expensive, scale poorly to complex dynamic scenes, and are therefore unsuitable for applications with strict latency constraints. To address this limitation, we draw inspiration from *ReSTIR*, a reuse-based technique for steady-state real-time rendering, and adapt its core principles to interactive-rate ToF simulation. However, naively applying existing *ReSTIR* methods to ToF rendering leads to severe inefficiency, as reused paths frequently violate optical path-length constraints and thus contribute little or no signal. We overcome this challenge by introducing a path reuse formulation that explicitly enforces physically valid optical path lengths. The key idea is *path-length-aware shift mapping*, a geometric transformation based on Newton’s method that adjusts reused light paths to satisfy temporal gating constraints, inspired by specular manifold exploration in steady-state caustics rendering. The resulting framework substantially improves the efficiency of ToF rendering across a wide range of scenarios, including complex scenes with glossy or specular materials and

dynamic motion. Our method supports both time-gated and transient rendering at interactive frame rates, enabling simulation under practical latency constraints. We demonstrate the effectiveness of our approach through two downstream applications, including shape reconstruction and navigation.

CCS Concepts: • **Computing methodologies** → **Ray tracing**; **Computational photography**.

Additional Key Words and Phrases: Time-of-Flight imaging, Time-of-Flight rendering, real-time ray tracing

ACM Reference Format:

Juhyeon Kim, Wojciech Jarosz, and Adithya Pediredla. 2026. ToF ReSTIR: Time-of-Flight Rendering with Spatio-temporal Reservoir Resampling. *ACM Trans. Graph.* 45, 4, Article 89 (July 2026), 18 pages. <https://doi.org/10.1145/3811299>

1 INTRODUCTION

Light travels fast, but not infinitely fast—and even small delays in a photon’s arrival carry rich information about the physical world. Time-of-flight (ToF) imaging leverages this principle by emitting controlled illumination and measuring when photons return to the sensor after interacting with the scene. These time-resolved measurements capture both direct reflections from visible surfaces and multiply scattered transport, often originating from partially or fully hidden objects. As a result, ToF imaging enables advanced sensing capabilities—such as non-line-of-sight reconstruction and material characterization in challenging environments—that are inaccessible to conventional steady-state intensity cameras.

Authors’ addresses: Juhyeon Kim, Dartmouth College, USA, juhyeon.kim.gr@dartmouth.edu; Wojciech Jarosz, Dartmouth College, USA, wojciech.k.jarosz@dartmouth.edu; Adithya Pediredla, Dartmouth College, USA, adithya.k.pediredla@dartmouth.edu.

Please use nonacm option or ACM Engage class to enable CC licenses. This work is licensed under a Creative Commons Attribution 4.0 International License. © 2026 Copyright held by the owner/author(s). ACM 0730-0301/2026/7-ART89 <https://doi.org/10.1145/3811299>

Time-of-flight rendering provides a physically grounded way for simulating measurements from time-resolved imaging systems and plays an important role in understanding, validating, and developing modern ToF sensors. Such simulations support offline analysis and algorithm development, and enable the generation of ground-truth datasets that may be difficult or impractical to acquire experimentally. Beyond these offline uses, improved rendering performance enables ToF-based systems to be evaluated in interactive and decision-driven settings. For example, applications such as real-time reconstruction, navigation, and closed-loop perception require simulation to operate within strict latency budgets in order to be applicable. Efficient and physically accurate ToF rendering therefore, can extend the scope of time-resolved simulation from offline analysis to interactive sensing and perception tasks.

Despite recent progress, existing ToF rendering methods face a fundamental trade-off. Physically accurate approaches [Jarabo et al. 2014; Liu et al. 2022; Marco et al. 2019; Pediredla et al. 2019b] are often computationally expensive and limited to offline use, while faster methods [Iseringhausen and Hullin 2020; Klein et al. 2016] rely on simplified assumptions such as reduced geometry, materials, or light transport. As a result, current techniques struggle to achieve both efficiency and physical generality in dynamic, large-scale scenes.

Motivated by these limitations, it is natural to ask whether techniques developed for real-time physically based steady-state rendering (e.g., games or commercial rendering) can be adapted to support interactive and scalable ToF simulation. In particular, reuse-based methods from Monte Carlo path tracing provide a promising starting point, as they aim to reduce computational cost while preserving physical correctness. Reservoir-based spatio-temporal resampling, so-called *ReSTIR* [Bitterli et al. 2020] and related approaches [Lin et al. 2022; Ouyang et al. 2021] are representative methods in this direction, demonstrating how aggressive spatio-temporal reuse can scale path tracing to dynamic and interactive settings. However, these methods are fundamentally designed for steady-state rendering and do not account for optical path length. When applied directly to ToF rendering, reused paths frequently violate temporal constraints, leading to invalid contributions (Fig. 2).

To address these challenges, we introduce a path-length-aware spatio-temporal reuse framework for ToF rendering. The key idea is *path-length-aware shift mapping*, which geometrically adjusts a reused path so that its optical path length matches the active time gate (red arrow in Fig. 2). We formulate this adjustment as a constrained root-finding problem and solve it using Newton’s method. Our approach is inspired by manifold exploration techniques for specular transport [Hong et al. 2025; Jakob and Marschner 2012; Lehtinen et al. 2013], but is adapted to enforce path-length constraints specific to ToF rendering. By incorporating path-length awareness into ReSTIR-style reuse, our method enables scalable and efficient simulation of ToF sensors in complex, dynamic environments. Our reuse strategy supports both per-frame image and transient histogram rendering, and we demonstrate its effectiveness in two downstream NLOS applications: shape reconstruction and inference-based navigation (Fig. 1).

The code and data are available on the project page¹.

¹https://juhyeonkim95.github.io/project-pages/tof_restir

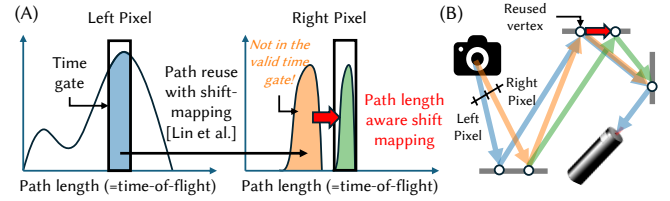


Fig. 2. Overview of proposed path-length-aware reuse strategies. (A) Naive path reuse from conventional steady-state ReSTIR (e.g., hybrid shift-mapping by Lin et al. [2022]) ends up going outside of the valid time gate. To address this issue, we introduce path-length-aware shift mapping, illustrated by the red arrow. (B) Illustration of a base path in the left pixel (blue), the naively reused path in the right pixel (orange), and the corrected path produced by our method (green), which satisfies the temporal constraint.

2 RELATED WORKS

2.1 Time-of-Flight Imaging

Here, we review representative ToF imaging systems that acquire either transient measurements—sequences of time-resolved images—or time-gated measurements, which capture a single image within a narrow temporal window. Jarabo et al. [2017] provide a more comprehensive review.

One common approach employs high-speed optical gating to selectively detect photons that arrive within short temporal windows after pulsed illumination. Intensified CCD/CMOS cameras achieve this using a fast-gating image intensifier, allowing selective capture of early or late light within sub-nanosecond windows [Cester et al. 2019]. By shifting these gates over repeated acquisitions, a full transient profile can be reconstructed. Streak cameras achieve even finer temporal resolution by mapping photon arrival times to spatial deflections, enabling picosecond-scale measurements [Velten et al. 2012, 2013]. Single-photon avalanche diode (SPAD) sensors have emerged as compact, scalable detectors operating at the level of individual photons [Zappa et al. 2007]. In time-correlated single-photon counting (TCSPC) mode, SPAD arrays time-stamp photon arrivals to build transient histograms at each pixel [Garipey et al. 2015; Henderson et al. 2019]. Continuous-wave ToF cameras are widely used for conventional range imaging, but transient information can also be recovered by multi-frequency phase measurements [Heide et al. 2013; Peters et al. 2015]. There are also interferometric approaches that measure optical phase with micron-scale path-length sensitivity [Gkioulekas et al. 2015; Kadambi et al. 2016].

ToF imaging has become an important tool in computational imaging and robotics. Time-resolved transient measurements enable advanced tasks such as non-line-of-sight (NLOS) reconstruction [Buttafava et al. 2015; Kirmani et al. 2011; Laurenzis and Velten 2014; Somasundaram et al. 2023; Tsai et al. 2017; Velten et al. 2012], separation of direct and indirect transport [O’Toole et al. 2014; Wu et al. 2014], and robust perception for autonomous navigation [Young et al. 2025]. Time-gated imaging has a narrower range of applications but remains effective for depth-selective sensing in autonomous vehicle [Grauer and Sonn 2015] or military applications [Baker et al. 2004], and some NLOS applications [Pediredla et al. 2019a]. Additionally, transient capture realized by shifting

gating delays behaves as time-gated imaging for dynamic scenes, since each gate samples a slightly different configuration.

2.2 Time-of-Flight Rendering

Time-of-flight (ToF) rendering simulates measurements from transient and time-gated sensors by explicitly incorporating optical path length into light transport. Following Liu et al. [2022]; Pediredla et al. [2019b], we use this term to refer to both transient rendering (time-resolved image sequences, i.e., histogram) and time-gated rendering (single images within a temporal window).

A key challenge shared by both ToF rendering tasks arises from the presence of a narrow temporal manifold induced by a time gate or a temporal bin. As a result, naive steady-state rendering algorithms tend to fail, since most sampled light paths fall outside the valid temporal range and contribute nothing. For transient rendering under a static scene, this can be partially mitigated by sharing samples across all temporal bins², where each sampled path is accumulated into its corresponding time bin [Adam et al. 2016; Ament et al. 2014; Jarabo 2012; Marco 2013], performing histogram density estimation. To obtain accurate transient profiles, a small bin width is required, but this often leads to noisy histograms. To address this issue, Jarabo et al. [2014] introduced progressive kernel-density estimation in the temporal domain, trading variance for bias. This was later extended to progressive spatio-temporal density estimation using photon beams [Jarosz et al. 2011] under participating media [Marco et al. 2019]. An alternative strategy of the temporal blurring is directly determining the time stamp and only sampling light paths that are valid at that specific time stamp. This is essential for time-gated rendering and also beneficial for transient rendering, as uniformly distributed samples are helpful for kernel density estimation. Prior work has explored path-length-constrained sampling in participating media [Jarabo et al. 2014] and surface transport via ellipsoidal path connection [Pediredla et al. 2019b], while more recent methods slice photon primitives into temporal wavefronts for efficient filtering [Liu et al. 2022].

Beyond forward rendering, differentiable ToF formulations [Wu et al. 2021; Yi et al. 2021] and extensions to Doppler effects [Kim et al. 2025, 2023] have also been explored.

From a performance perspective, none of the previous ToF rendering methods targeted interactive rate rendering for complex scenes. Several fast radiosity-based approaches on GPU have been proposed to accelerate simulation by modeling objects as Lambertian patches [Klein et al. 2016] or interpolating radiance over mesh triangle vertices [Iseringhausen and Hullin 2020]. However, these methods are typically limited to two-bounce light transport and often require heavy mesh preprocessing. Recently, some GPU-based transient path tracers have also been introduced, but they are still not designed for interactive purposes [Royo et al. 2022, 2025].

Overall, while physically accurate ToF rendering techniques exist, they remain computationally expensive and difficult to apply to large-scale, dynamic environments at interactive rates on a GPU. This motivates the development of scalable ToF rendering approaches

inspired by modern real-time steady-state rendering techniques, while explicitly accounting for temporal constraints.

2.3 Spatio-Temporal Path Reuse

Reservoir-based spatiotemporal reuse, or ReSTIR, has recently transformed steady-state real-time Monte Carlo rendering [Bitterli et al. 2020]. ReSTIR leverages an importance-resampling strategy [Talbot et al. 2005] to reuse light samples across neighboring pixels and successive frames, achieving high-quality results with only a few samples per pixel. Follow-up works have extended these principles to global illumination [Ouyang et al. 2021], volumetric transport [Lin et al. 2021], general multi-bounce path tracing [Lin et al. 2022], depth of field and antialiasing [Zhang et al. 2024], motion blur [Liu et al. 2025], BDPT [Hedstrom et al. 2025], path guiding [Zeng et al. 2025], and specular manifold sampling [Hong et al. 2025]. These methods excel in scenarios where spatial and temporal coherence can be exploited, achieving high-quality results with real-time performance even in complex scenes. However, they assume a steady-state setting and are agnostic to optical path length. As a result, when directly applied to ToF sensors, reused paths often become temporally invalid and fail to yield meaningful variance reduction.

2.4 Sampling on Specular Manifold

Sampling a narrow temporal manifold is closely related to sampling specular manifolds in steady-state rendering. A representative example is *manifold walk* [Jakob and Marschner 2012], which proposes finding valid specular paths from initially invalid mutated MCMC paths using Newton’s iteration. This idea was later extended to half-vector space light transport [Kaplanyan et al. 2014] and manifold next-event estimation [Hanika et al. 2015; Zeltner et al. 2020]. Several follow-up works further improved specular manifold sampling through techniques such as manifold path guiding [Fan et al. 2023; Li et al. 2022] and spatio-temporal reuse [Xu et al. 2023]. More recently, Hong et al. [2025] adopted sample-space partitioning to improve specular manifold sampling and demonstrated its effectiveness when implemented within a ReSTIR framework. Our temporal manifold path reuse is closely related to these specular manifold methods, but differs in the underlying degrees of freedom of the constraint manifold. We address this difference by introducing an additional gauge condition to uniquely determine valid solutions.

3 BACKGROUND

3.1 Time-of-Flight Rendering

The path integral formulation of conventional steady-state Monte Carlo rendering is given by [Veach 1997]

$$I = \int_{\mathcal{P}} f(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}) \quad (1)$$

where I denotes the measurement at a pixel, $\mathcal{P}$ is the set of all possible light paths $\bar{\mathbf{x}}$, consisting of vertices $\mathbf{x}_0, \dots, \mathbf{x}_K$, $f(\bar{\mathbf{x}})$ is the path throughput and μ is its Lebesgue measure. For ToF rendering, the path integral formulation is slightly modified as [Jarabo et al. 2014; Pediredla et al. 2019b]

$$I(\tau) = \int_{\mathcal{P}} W_{\tau}(\ell(\bar{\mathbf{x}})) f(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}) \quad (2)$$

²This is called *path reuse* in Jarabo et al. [2014]. This keeps the sampled path unchanged over temporal bins, different from path reuse in ReSTIR, which is in a more aggressive form that can even deform the path (e.g., reconnect to a new vertex) for reuse.

where W_τ is *path-length importance* function at time τ , and $\ell(\bar{\mathbf{x}})$ is the optical path length defined as $\sum_{k=0}^{K-1} \eta_k \|\mathbf{x}_{k+1} - \mathbf{x}_k\|$ where η_k is the refractive index. For simplicity, we set the speed of light as $c = 1$ throughout the paper. For transient histogram rendering, we record a sequence of images for different $\tau \in \{\tau_1, \dots, \tau_B\}$ where B is the number of histogram bins.

3.2 Resampled Importance Sampling and ReSTIR

We next review the theoretical foundations of ReSTIR for evaluating the path integral in Eq. (1). This summary is based on Lin et al. [2022]; Sawhney et al. [2022], and we recommend readers to read the original papers for additional details.

3.2.1 Basic RIS (Same Domain, Same PDF). Resampled Importance Sampling (RIS) [Talbot et al. 2005] aims to estimate the integral of a function f using a non-negative unnormalized target function $\hat{p}(x)$ in an unbiased manner. In its basic form, RIS begins by generating M independent candidate samples $\{x_1, \dots, x_M\}$ from a domain Ω according to a pdf p that is easy to sample. From these candidates, a single representative sample $y = x_s$ is selected with a probability proportional to $w_s / \sum_{i=1}^M w_i$, where a resampling weight w_i is

$$w_i = \frac{1}{M} \hat{p}(x_i) W_i, \quad W_i := \frac{1}{p(x_i)}. \quad (3)$$

The factor W_i is known as the *unbiased contribution weight* (UCW). The probability density function of the selected sample y , denoted by $p_Y(y)$, which is required for traditional Monte Carlo integration, is generally unavailable in a closed form. Instead, RIS computes the UCW of the winner as

$$W_y := \frac{1}{\hat{p}(y)} \sum_{i=1}^M w_i, \quad (4)$$

which can be used as an effective substitute for $1/p_Y(y)$ [Bitterli et al. 2020]. This yields the RIS estimator for an integrand f ,

$$\langle I \rangle_{\text{RIS}} = f(y) W_y. \quad (5)$$

Provided that $p_Y(y)$ is non-zero wherever $f(y) > 0$ (i.e., over the support of f), the estimator remains unbiased. A typical choice is to set $\hat{p}(x) = f(x)$, so that RIS asymptotically achieves an ideal importance sampling or zero-variance estimator.

3.2.2 Combining with MIS (Same Domain, Different PDF). RIS can be extended to incorporate multiple importance sampling (MIS) when candidate samples are generated using different sampling strategies (e.g., BSDF, light sampling), while still operating over the same domain Ω . In this setting, each candidate sample x_i is drawn from its own proposal distribution p_i . To combine these samples consistently, a resampling MIS scheme [Talbot 2005] is applied, analogous to the balance heuristic [Veatch 1997]. Specifically, each sample is assigned a mixing weight given by

$$m_i(x_i) = \frac{p_i(x_i)}{\sum_{j=1}^M p_j(x_i)}. \quad (6)$$

These mixing weights replace the uniform factor $1/M$ used in the i.i.d. case (Eq. (3)) and lead to the generalized resampling weight

$$w_i = m_i(x_i) \hat{p}(x_i) W_i, \quad W_i = \frac{1}{p_i(x_i)}. \quad (7)$$

As long as the collection of proposal distributions p_i jointly covers the support of $\hat{p}$, the resulting RIS estimator remains unbiased.

3.2.3 Generalized RIS (Different Domain). Resampling candidates may be drawn from different target domains (reused across different pixels or time steps), which makes the previous assumption that all samples share the domain Ω invalid. Generalized RIS [Lin et al. 2022] addresses this limitation by allowing candidate samples x_i to be drawn from different domains Ω_i , each associated with its own target function $\hat{p}_i$. The key idea that enables resampling across different domains is *shift mapping*, which defines a bijective transformation between domains. Note that shift mapping does not always need to be successful, as we can reject the reuse if shift mapping fails. Given a candidate sample $x_i \in \Omega_i$, a shift map $S_i : \Omega_i \rightarrow \Omega$ maps it to a corresponding sample $y_i = S_i(x_i) \in \Omega$. Under this formulation, the resampling weight is given by

$$w_i = m_i(y_i) \hat{p}(y_i) W_i \cdot \left| \frac{\partial S_i}{\partial x_i} \right|. \quad (8)$$

To compute the MIS weights m_i , Lin et al. [2022] introduce a generalized balance heuristic given by

$$m_i(y_i) = \frac{M_i \hat{p}_{\leftarrow i}(y_i)}{\sum_{j=1}^M M_j \hat{p}_{\leftarrow j}(y_j)} \quad (9)$$

where M_i denotes a confidence weight, and the term “ $\hat{p}$ from i ” is defined as $\hat{p}_{\leftarrow i}(y_i) = \hat{p}_i(x_i) \left| \partial S_i^{-1} / \partial y_i \right|$ or 0 if S_i^{-1} is not valid. This formulation remains applicable even when the x_i originate from RIS steps, such as merging winner samples from neighboring pixels, in which case the proposal density p_i may not be explicitly available.

Selecting a good shift-mapping S is critical, since a sample that is important in one domain may contribute little or nothing after being mapped to another domain where the integrand changes significantly. A variety of shift-mapping strategies have been proposed in prior works. Path reconnection [Lehtinen et al. 2013; Ouyang et al. 2021] reconnects base and offset paths at the earliest compatible surface interaction and performs well for moderately rough materials. Alternative strategies include half-vector copying [Kettunen et al. 2015] and random replay [Hua et al. 2019], which are particularly effective for specular transport. To handle a broad class of materials robustly, Lin et al. [2022] propose a hybrid shift-mapping approach that mixes path reconnection and random replay (Fig. 3).

We note that existing shift-mapping techniques are designed for steady-state rendering, which tries to match $f(\bar{\mathbf{x}})$ after the reuse. In ToF rendering, however, the integrand additionally includes a path-length-dependent term $W_\tau(\ell(\bar{\mathbf{x}}))$, which introduces new challenges that are not addressed by prior methods.

3.2.4 ReSTIR. ReSTIR builds upon previous RIS-based resampling strategies to enable sample sharing across pixels, both spatially within a single frame and temporally across consecutive frames. Rather than explicitly storing all M samples used in RIS, ReSTIR employs Weighted Reservoir Sampling (WRS) [Chao 1982] to perform resampling in an incremental manner. This approach uses a compact data structure, referred to as a *reservoir*, which stores only the current winner sample and its associated contribution weight. A typical ReSTIR pipeline proceeds through the following stages:

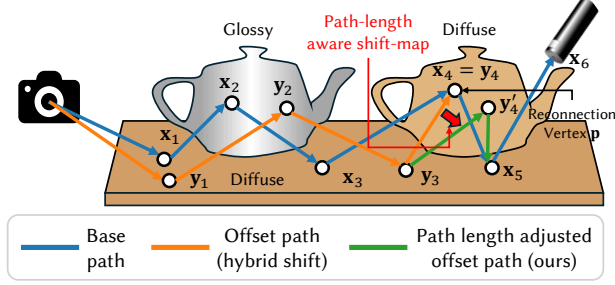


Fig. 3. Baseline shift mapping (hybrid shift mapping [Lin et al. 2022]) transforms the base path $\bar{x}$ (blue) into an offset path $\bar{y}$ (orange), with the reconnection vertex $\mathbf{p}$ at $y_4 (= x_4)$. We further perturb the offset path $\bar{y}$ to $\bar{y}'$ (green) to satisfy the path-length constraint, $\ell(\bar{y}') = \ell(\bar{x}) + \delta\ell$. Rather than perturbing the entire offset path, we apply path-length-aware shift mapping only at the reconnection vertex, similar to Sawhney et al. [2022]. (This figure is adapted from Fig. 6 of that paper to highlight the similarity.)

- (1) Initial sampling: Generate multiple initial candidates and apply RIS to select a single representative sample, which is then stored in the reservoir.
- (2) Temporal resampling: Reuse samples across consecutive frames, often using motion vectors.
- (3) Spatial resampling: Select reservoirs from neighboring pixels and merge them into the current reservoir.
- (4) Final shading: Compute the final pixel estimate by evaluating $f(y)W_y$ using the reservoir sample.

In the following sections, we describe how this ReSTIR framework can be extended to support ToF rendering, where path-length constraints must be explicitly enforced.

4 PATH LENGTH AWARE SHIFT MAPPING

In this section, we introduce path-length-aware shift mapping strategies that address the path-length mismatch arising from naive path reuse. We further show efficient initial RIS sample generation from previously proposed ellipsoidal path connections [Pediredla et al. 2019b], or *shrinking* from samples with wider gate, which also exploits path-length-aware shift mapping. Finally, we show how our method extends to transient and Doppler rendering scenarios.

4.1 Offset Path Further Perturbation

We begin with a baseline shift-mapping algorithm which transforms a *base path* $\bar{x}$ into a shift-mapped *offset path* $\bar{y}$. Specifically, we use hybrid shift-mapping [Lin et al. 2022] that uses random seed replay and reconnect at the first consecutive diffuse vertex (so called *reconnection vertex* $\mathbf{p}$, corresponding to y_4 in Fig. 3). Now, the goal of path-length-aware shift mapping is to find an additional mapping $S: \bar{y} \mapsto \bar{y}'$ that further perturbs the offset path $\bar{y}$ into a new path $\bar{y}'$ satisfying a prescribed path-length condition

$$\ell(\bar{y}') = \ell(\bar{x}) + \delta\ell, \quad (10)$$

where $\delta\ell$ is a constant determined by the reuse configuration (e.g., $\delta\ell = 0$ for reuse within the same time gate, or $\delta\ell = \Delta t$ for time-gate shifting). Since we perform the transformation starting from $\bar{y}$, it is

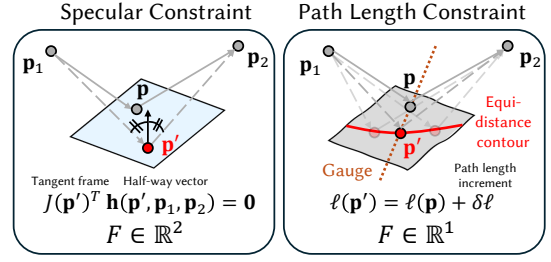


Fig. 4. Specular material imposes a 2D constraint, which uniquely determines $\mathbf{p}'$ (function of ξ') on the surface given $\mathbf{p}_1$ and $\mathbf{p}_2$. On the other hand, path length imposes a 1D constraint, the feasible $\mathbf{p}'$ lies on an equi-distance contour on the surface. To further reduce the degree of freedom, we introduce an additional constraint, referred to as a *gauge* in this paper.

convenient to express the constraint as

$$\ell(\bar{y}') = \ell(\bar{y}) + \delta\ell, \quad (11)$$

where $\delta\ell$ is a fixed constant given $\bar{x}$; we retain this notation by a slight abuse. Rather than modifying the entire offset path $\bar{y}$, we only adjust the reconnection vertex $\mathbf{p}$, following a strategy similar to the localized mutation used by Sawhney et al. [2022]. This further simplifies the constraint into

$$\ell(\mathbf{p}') = \ell(\mathbf{p}) + \delta\ell, \quad (12)$$

where $\ell(\mathbf{p})$ is now path length sum from two adjacent vertices $\mathbf{p}_1, \mathbf{p}_2$, on the offset path (corresponds to y_3, x_5 in Fig. 3). Because our transformation is deterministic, we refer to it as a mapping rather than a mutation.

4.2 Two-Dimensional Surface Parameterization

We focus on surface-based light transport and parameterize the reconnection vertex $\mathbf{p}$ using a two-dimensional coordinate $\xi \in \mathbb{R}^2$. This parameterization may take various forms, such as barycentric coordinates on a triangle, uv coordinates of an implicit surface, or polar coordinates on a hemispherical shadow map. Our objective is to find a new surface coordinate ξ' that satisfies the path-length constraint:

$$\ell(\xi') = \ell(\xi) + \delta\ell. \quad (13)$$

Mathematically, this problem closely resembles the computation of valid paths on thin (or delta) manifolds induced by specular reflection [Jakob 2013; Kaplanyan et al. 2014; Lehtinen et al. 2013]. As in prior work, we therefore employ Newton's method to solve for ξ' . However, there is a fundamental difference between specular manifolds and the path-length manifold considered in ToF rendering. Specular constraints restrict two degrees of freedom from halfway vector alignment, which locally determine a unique solution for ξ' . In contrast, the path-length constraint introduces only a single scalar constraint, leaving one remaining degree of freedom in ξ' (Fig. 4). As a result, ξ' is not uniquely defined under the path-length constraint alone.

To resolve this ambiguity, we introduce an additional constraint, which we refer to as a *gauge condition* by analogy. This terminology reflects the similarity to gauge fixing in physics, where redundant degrees of freedom are removed by enforcing invariance under a

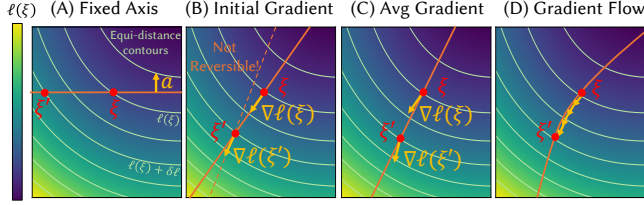


Fig. 5. Methods for imposing an additional constraint. (A) A fixed-axis gauge is reversible and easy to implement, but it may require traveling a long distance to find a valid solution. (B) Tracing along the gradient reduces the displacement, but it is not guaranteed to be reversible. (C) Our average-gradient method uses gradients at both the start and end points, ensuring reversibility. (D) Gradient flow is also bijective and yields small displacements, but it is not based on Newton's method and requires many small steps.

chosen constraint, although there is no direct physical connection to electromagnetism. Assuming that a chosen gauge function $G(\cdot)$ remains invariant under the shift, we can define a two-dimensional constraint function F as

$$F(\xi, \xi') = \begin{bmatrix} \ell(\xi') - \ell(\xi) - \delta\ell \\ G(\xi') - G(\xi) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \quad (14)$$

This formulation yields a well-posed system that can be solved efficiently using Newton iteration.

4.3 Selecting a Good Gauge Condition

A simple choice of gauge is a fixed linear axis, e.g., $G(\xi) = a^T \xi$ (Fig. 5-(A)). This choice is reversible and easy to implement. However, it often results in large displacements in surface parameter space, which can negatively impact path similarity. To reduce deformation, a natural alternative is to move along the gradient direction $\nabla \ell(\xi)$ which locally minimizes vertex displacement (Fig. 5-(B)). However, this approach is not reversible: starting from ξ' and reducing the path length by $\delta\ell$ does not recover ξ unless $\nabla \ell(\xi) \parallel \nabla \ell(\xi')$. This lack of bijectivity violates the requirements for unbiased shift mapping.

Instead, we propose a bijective average-gradient gauge (Fig. 5-(C)) which symmetrically depends on both the start and end points. Specifically, we define the constraint function

$$F(\xi, \xi') = \begin{bmatrix} \ell(\xi') - \ell(\xi) - \delta\ell \\ (Rm)^T(\xi' - \xi) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad (15)$$

where $m = (\nabla \ell(\xi) + \nabla \ell(\xi')) / 2$ and R denotes a 90 degree rotation matrix. Because the search direction m depends equally on ξ, ξ' , this formulation is symmetric and therefore reversible, ensuring bijectivity and unbiasedness of shift-mapping. Practically, in many cases, we could not easily see the biasedness of sample reuse using the initial gradient only, as the moving distance is typically small compared to the gradient variation. However, to maintain theoretical correctness, we choose the average gradient method throughout the manuscript. Meanwhile, we can also use mid-point gradient $m = \nabla \ell((\xi + \xi')/2)$, and this behaves similarly to using average gradient. However, since our objective is to solve for the endpoint ξ' , the midpoint method still requires evaluating a path at ξ' , which introduces additional overhead.

Newton's Iteration. Now we solve for ξ' using Newton's method, initialized at ξ and iteratively updated using gradients:

$$\xi'_0 = \xi, \quad \xi'_{k+1} \leftarrow \xi'_k - \left(\frac{\partial F}{\partial \xi'_k} \right)^{-1} F(\xi, \xi'_k). \quad (16)$$

We use backtracking line search as Jakob and Marschner [2012] for better step-size control. We also require the determinant of the Jacobian of the shift mapping, which can be computed via implicit differentiation as

$$\det \left(\frac{\partial \xi'}{\partial \xi} \right) = \det \left(\frac{\partial F}{\partial \xi} \right) \bigg/ \det \left(\frac{\partial F}{\partial \xi'} \right). \quad (17)$$

Note that we still need to consider an additional Jacobian term that transforms ξ to $\mathbf{p}$ to be applied to Eqs. (8) and (9):

$$\det \left(\frac{\partial \mathbf{p}'}{\partial \mathbf{p}} \right) = \frac{\sqrt{\det J'^T J'}}{\sqrt{\det J^T J}} \det \left(\frac{\partial \xi'}{\partial \xi} \right), \quad (18)$$

where $J = (\partial \mathbf{p} / \partial \xi) \in \mathbb{R}^{3 \times 2}$ is the *local tangent frame*.

Both Newton's iteration and Jacobian evaluation require the partial derivatives of F with respect to ξ and ξ' :

$$\begin{aligned} \frac{\partial F}{\partial \xi'} &= \begin{bmatrix} \nabla \ell(\xi') \\ (Rm + 0.5 \nabla^2 \ell(\xi') R(\xi' - \xi))^T \end{bmatrix}, \\ \frac{\partial F}{\partial \xi} &= \begin{bmatrix} -\nabla \ell(\xi) \\ -(Rm + 0.5 \nabla^2 \ell(\xi) R(\xi' - \xi))^T \end{bmatrix}, \end{aligned} \quad (19)$$

which involve both the gradient and the Hessian of ℓ .

Gradient and Hessian Evaluation. The gradient and Hessian of ℓ can be evaluated by transforming derivatives with respect to $\mathbf{p} \in \mathbb{R}^3$ into derivatives with respect to $\xi \in \mathbb{R}^2$. The path length is given by $\ell(\mathbf{p}) = \|\mathbf{p}_1 - \mathbf{p}\| + \|\mathbf{p}_2 - \mathbf{p}\|$, and we can evaluate its gradient as

$$\nabla_{\mathbf{p}} \ell = -\frac{\mathbf{p}_1 - \mathbf{p}}{\|\mathbf{p}_1 - \mathbf{p}\|} - \frac{\mathbf{p}_2 - \mathbf{p}}{\|\mathbf{p}_2 - \mathbf{p}\|} = -(\mathbf{d}_1 + \mathbf{d}_2) = -\mathbf{s}, \quad (20)$$

where $\mathbf{d}_1, \mathbf{d}_2$ are unit direction vectors and $\mathbf{s}$ is the unnormalized half-vector. The Hessian of ℓ with respect to $\mathbf{p}$ is

$$\nabla_{\mathbf{p}}^2 \ell = \frac{I - \mathbf{d}_1 \mathbf{d}_1^T}{\|\mathbf{p}_1 - \mathbf{p}\|} + \frac{I - \mathbf{d}_2 \mathbf{d}_2^T}{\|\mathbf{p}_2 - \mathbf{p}\|} = A. \quad (21)$$

We can then transform the gradient with respect to ξ , using the local tangent frame J ,

$$\nabla_{\xi} \ell = J^T (-\mathbf{s}). \quad (22)$$

Similarly, the Hessian with respect to ξ is given by

$$\nabla_{\xi}^2 \ell = J^T A J + Q, \quad (23)$$

where Q is defined as

$$Q \in \mathbb{R}^{2 \times 2}, \quad Q_{ij} = (-\mathbf{s})^T \left(\frac{\partial^2 \mathbf{p}}{\partial \xi_i \partial \xi_j} \right) \quad (24)$$

for $i, j \in \{1, 2\}$. The second-order term $\partial^2 \mathbf{p} / \partial \xi_i \partial \xi_j$, which captures surface curvature, can be evaluated for specific surface parameterizations. But it is practical to ignore this term ($Q = 0$), effectively treating the geometry as a collection of infinitesimal flat patches.

Comparison with Gradient Flow. Another intuitive alternative for handling the additional degrees of freedom is to follow the continuous *gradient flow* $\xi(s)$, defined by $d\xi/ds = \nabla L(\xi(s))$ (Fig. 5-(D)). This approach also guarantees bijectivity, but it is significantly slower than Newton’s method (linear vs. quadratic convergence). Further discussion can be found in the supplementary material.

Cost of Newton’s Iteration. In prior steady-state specular manifold methods, the cost of Newton’s iteration is a major concern, as it often requires tracing long specular chains via ray tracing. In contrast, for the path-length constraint in our setting, it suffices to perturb a single path vertex. This significantly reduces the cost of each Newton iteration, making it much less expensive than the specular-manifold case. We note that this strategy does not directly extend to mixed manifolds involving both specular and path-length constraints, which we do not consider in this work.

4.4 Initial Sample Generation

So far, we have focused on the efficient reuse of valid samples using path-length-aware shift mapping. However, time-gated rendering also suffers from a low probability of generating valid initial samples, as the time gate accepts only a very narrow range of path lengths. In this subsection, we discuss strategies to address this issue.

4.4.1 Initialization with Ellipsoidal Path Connection. One approach is to adopt existing techniques that explicitly target valid sample generation for time-gated rendering, such as ellipsoidal path connection [Pediredla et al. 2019b]. Our path-length-aware shift mapping is complementary to these techniques, and can therefore be combined with them to improve the quality of initial RIS samples. Some adjustments are required to make this approach suitable for GPU implementation, which we discuss in the implementation section.

4.4.2 Initial Sampling with Path-length Shrink Mapping. In some cases, ellipsoidal path connection is not suitable, either because it is too slow or because the geometry is not represented as a triangular mesh. To address this, we additionally use path-length-aware shift mapping for initial sample generation. The key idea is simple: we first sample paths using a wider time gate, and then shift (or *shrink*) these samples into a narrower time gate. In this setting, the path-length constraint in Eq. (10) becomes

$$\ell(\bar{x}') = \frac{1}{K} (\ell(\bar{x}) - \ell_0) + \ell_0, \quad (25)$$

where we use $\bar{x}'$ instead of $\bar{y}'$ since no baseline shift mapping is required. Here, ℓ_0 is the path length corresponding to the center of the time gate, and K denotes the scaling factor of the wider time gate. The difference between translation-based mapping (used for spatio-temporal reuse) and shrink mapping (used for initial sample refinement) is illustrated in Fig. 6. We can therefore reuse the previous Newton iteration by setting $\delta\ell = \frac{1}{K} (\ell(\bar{x}) - \ell_0) + \ell_0 - \ell(\bar{x})$. The only required modification is to scale the Jacobian determinant by $1/K$. Importantly, note that we do not apply this shrink mapping for all RIS samples, but only to the RIS winner, so the computational cost is minimal.

Initial sample generation with shrink mapping can introduce bias if used in isolation. This arises for two main reasons: (1) Newton’s method may become unstable when initial samples are far from the

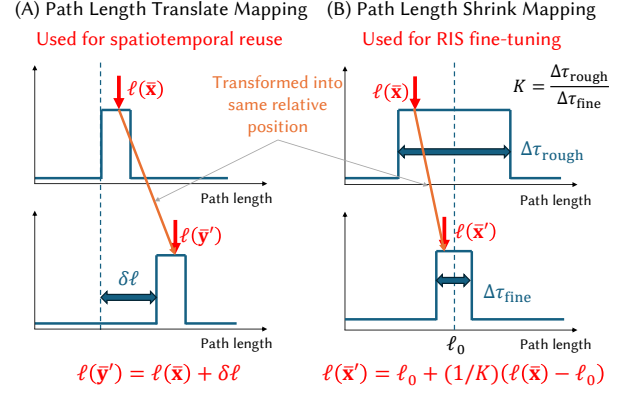


Fig. 6. Two types of path-length-aware shift mappings are used for different purposes. (A) Path-length translation mapping is used for spatio-temporal path reuse. (B) Path-length shrink mapping is used for efficient initial sample generation: samples are drawn from a wider time gate and then shrunk back to the original gate.

target manifold, and (2) the support of the transformed paths may be smaller than the valid time gate, causing the Jacobian correction (scaling by $1/K$) to be over-applied, which typically leads to darker estimates. This bias can be eliminated by combining shrink-mapped samples with canonical samples from the target (narrow) time gate using proper MIS weighting (Eq. (9)), which yields an unbiased estimator. In practice, however, we find that allocating all samples to the wider time gate often provides stronger variance reduction, which may outweigh the bias introduced by omitting canonical samples. Further discussion is provided in the results section.

4.5 Extension to Transient Histogram Rendering

Our method is applicable not only to time-gated image rendering, which produces a two-dimensional image ($H \times W$), but also to transient histogram rendering, which outputs a three-dimensional transient histogram ($H \times W \times B$) per frame. By interpreting transient histogram as a sequence of time gates, we can directly apply path-length-aware shift mapping independently to each histogram bin. Compared to the image case, histogram rendering introduces two differences: (1) all B reservoirs must be updated for each pixel, increasing the reuse overhead, and (2) there is an additional reuse dimension along the bin axis B . An overview comparing path reuse for image and histogram rendering is shown in Fig. 7.

Cost-to-Performance Analysis. As mentioned earlier, a key challenge of path reuse in the histogram setting is the cost-to-performance trade-off, as reuse can be expensive. Here, we provide a simple analysis to illustrate this effect. Let the cost of tracing a path sample be C_{trace} , and the cost of performing a path reuse operation be C_{reuse} . For image rendering with path reuse, the total cost is $C_{\text{trace}} + C_{\text{reuse}}$. For transient histogram rendering, reuse must be applied independently to each bin, increasing the reuse cost by a factor of B , while the per-bin quality improvement remains unchanged. As a result,

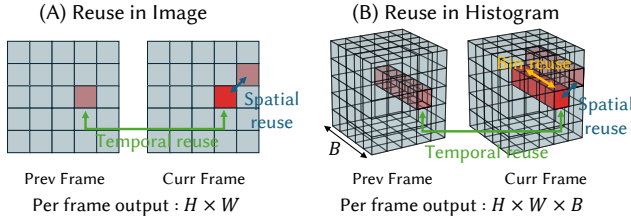


Fig. 7. (A) Time-gated image rendering whose per-frame output is $H \times W$, we can use temporal and spatial reuse. (B) Transient histogram rendering whose per-frame output is $H \times W \times B$, has two differences: (1) reservoir update is more expensive as we need to update all B reservoirs per pixel, and (2) we can use not only temporal and spatial reuse but also bin reuse.

the relative cost-to-performance efficiency degrades by a factor of

$$\frac{C_{\text{trace}} + C_{\text{reuse}}}{C_{\text{trace}} + B \times C_{\text{reuse}}}. \quad (26)$$

As B increases, the effectiveness of path reuse therefore decreases. This effect is particularly pronounced when C_{reuse} is large, for example, when path reuse requires expensive path throughput recomputation. Refer to the results section for a detailed comparison.

Comparison with Jarabo et al. [2014]’s ‘Path Reuse’. For initial sample generation, we adopt the “path reuse” in Jarabo et al. [2014] (which differs from ReSTIR-style path reuse), even though our overall framework uses ReSTIR. Specifically, we deposit each sample’s contribution into its corresponding time bin and use these as initial samples for subsequent ReSTIR-style reuse. We consider the transient histogram as a sequence of time-gated images and apply reuse independently for each bin *only* during the ReSTIR-style path reuse.

4.6 Extension to Doppler Frequency Rendering

The path-length constraints introduced in the previous sections can be naturally extended to the frequency domain induced by the Doppler effect. Analogous to the time-resolved path integral in Eq. (2), the frequency-resolved power spectral density (PSD) $S(\nu)$ resulting from Doppler shift [Kim et al. 2025] is given by

$$S(\nu) = \int_{\mathcal{P}} W_{\nu}(\omega(\bar{\mathbf{x}})) f(\bar{\mathbf{x}}) d\mu(\bar{\mathbf{x}}). \quad (27)$$

Here W_{ν} is a frequency weight similar to W_{τ} , and $\omega(\bar{\mathbf{x}})$ denotes the path frequency derived from the *path velocity* $u(\bar{\mathbf{x}}) = -\frac{d}{dt}\ell(\bar{\mathbf{x}})$. This can be expressed as

$$u(\bar{\mathbf{x}}) = \sum_{k=0}^{K-1} \eta_k (\mathbf{v}_k - \mathbf{v}_{k+1}) \cdot \hat{\mathbf{d}}_k, \quad (28)$$

where $\hat{\mathbf{d}}_k$ is the unit vector from $\mathbf{x}_k$ to $\mathbf{x}_{k+1}$ and $\mathbf{v}_k = \frac{d}{dt}\mathbf{x}_k$ denotes the instantaneous velocity at each vertex. The path velocity induces a Doppler frequency shift given by $\Delta f(\bar{\mathbf{x}}) := f_0 u(\bar{\mathbf{x}})/c$, where f_0 is the carrier frequency³.

Rather than constraining the optical path length $\ell(\mathbf{p})$, we impose a constraint on its temporal derivative $\frac{d}{dt}\ell(\mathbf{p})$ i.e., path velocity.

³For chirped lasers, ω depends on both path length and velocity. Here, we consider only a standard (constant-frequency) laser, for which $\omega(\bar{\mathbf{x}}) = 2\pi\Delta f(\bar{\mathbf{x}})$.

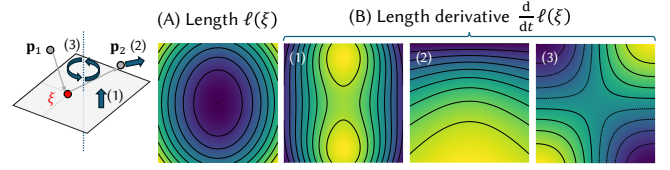


Fig. 8. (A) Path length $\ell(\xi)$ on the plane, forming ellipsoidal iso-value contours. (B) Temporal derivative of path length, $\frac{d}{dt}\ell(\xi)$, on the plane under three different scenarios: (1) the plane moves upward, (2) p_2 moves away, and (3) the plane rotates, each producing distinct iso-value contours.

This constraint defines a different manifold from the path-length manifold considered earlier. We visualize examples of manifolds induced by path-velocity constraints in Fig. 8 for three different cases. A comparison between time-gated and frequency-gated rendering is also summarized in Tab. 1. Overall, the same path reuse framework can be applied by substituting the path-length constraint with the path-velocity constraint. Full derivations of the corresponding gradients and Hessians are provided in the supplementary material.

Table 1. Comparison between time-domain and frequency-domain formulations and their corresponding constraints.

	Time Domain	Frequency Domain
Measurement	Transient signal $I(\tau)$	PSD $S(\nu)$
Depends on	Path Length	Path Velocity
Notation	$\ell(\bar{\mathbf{x}})$	$u(\bar{\mathbf{x}}) = -\frac{d}{dt}\ell(\bar{\mathbf{x}})$
Constraint	$\ell(\mathbf{p}') = \ell(\mathbf{p}') + \delta\ell$	$\frac{d}{dt}\ell(\mathbf{p}') = \frac{d}{dt}\ell(\mathbf{p}') + \delta u$

5 IMPLEMENTATION DETAIL

Reservoir Implementation. Our reservoir structure largely follows the design of Lin et al. [2022]. The main difference is that we additionally store the next vertex of the reconnection vertex, and keep the reconnection information (e.g., incident direction and radiance used to recompute the path throughput) at this next vertex rather than at the original reconnection vertex. This difference is illustrated in Fig. 9. Note that for temporal reuse under dynamic lighting or animated meshes, the reconnection information must be re-evaluated to remain valid. For further details, see Lin et al. [2022].

Light Source. Rather than considering general scene lighting, we restrict our setup to delta light sources, which are commonly used in ToF applications. Specifically, we model a spotlight under two regimes: (1) a collimated, laser-like case with a very small angular extent, and (2) a point light-like case with a large angular extent. In the collimated case, we trace additional rays from the light source to generate light samples prior to performing next event estimation (NEE), similar to bidirectional path tracing. In the point light case, we instead rely on direct NEE.

Ellipsoidal Path Connection. Several modifications are required to adapt the original ellipsoidal path connection method [Pediredla et al. 2019b] to our setting. In the original approach, ellipsoid–AABB and ellipsoid–triangle intersection tests are performed on the BVH

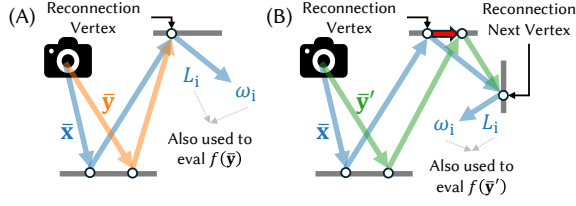


Fig. 9. (A) Lin et al. [2022] stores the hit record of the reconnection vertex along with additional information (incident direction ω_i and radiance L_i) to recompute the offset path throughput. (B) We additionally store the next vertex after the reconnection vertex and attach this information to that vertex instead of the original reconnection vertex. The red arrow indicates path-length-aware shift mapping.

to gather all triangles that may intersect a given ellipsoid. However, this strategy does not scale well on GPUs. For complex meshes, it can produce thousands of candidate triangles, leading to excessive memory usage and register spilling. Instead of explicitly collecting all intersecting triangles, we sample BVH nodes using heuristic weights. Each node is assigned a weight proportional to the total area of the triangles it contains, and set to zero weight if it does not intersect the ellipsoid. We also apply MIS between direct connection and ellipsoidal path connection to robustly handle specular materials. Finally, to avoid the complexity of a full BDPT implementation, we restrict ellipsoidal path connection to NEE only.

6 RESULTS

We implement our algorithm within the Falcor framework [Kallweit et al. 2022] and conduct experiments with an AMD Ryzen Pro 3955WX CPU and an NVIDIA RTX 3090 GPU. Most test scenes are taken from Bitterli [2016]; however, to better mimic laser-based ToF sensing, we replace the original scene light sources with a custom delta light source. We report and visualize only indirect illumination, as direct illumination under a delta light source is trivial.

6.1 Offline Time-Gated Rendering with Spatial Reuse

We first present offline time-gated rendering results for the CORNELL-Box scene with different material configurations for initial validation. A collimated laser illuminates the upper region of the back wall, with its direction parallel to the camera direction. The box size is $2 \times 2 \times 2$, and the time gate, which is a box function, is set to cover $0.5 - 2.5\%$ of the scene scale. For the ReSTIR configuration, following Lin et al. [2022], we use 32 path trees as initial candidates and perform spatial reuse three times, each time considering five neighboring pixels within a 10-pixel radius. We compare three methods: path tracing without any reuse, naive reuse based on ReSTIR-PT, and our proposed method. All methods are evaluated under the same time budget. We report the mean absolute percentage error (MAPE) as the evaluation metric, following Lin et al. [2022].

The results are shown in Fig. 10. Path tracing produces noisy images, while naive reuse moderately improves image quality but still fails to produce satisfactory results. Interestingly, naive reuse exhibits ellipsoidal reuse patterns (highlighted by the white curved arrow), as valid reuse without explicit path-length preservation can

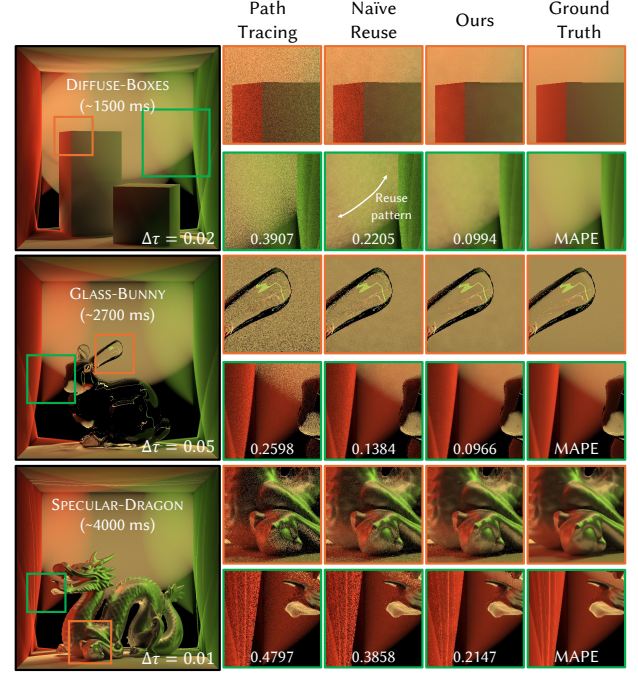


Fig. 10. Comparison of the CORNELL-Box scene with various objects and materials under an equal time budget. All images are rendered at a resolution of 1024×1024 , with the maximum path depth set to 6.

only occur along ellipsoidal loci that satisfy the path-length constraint. In contrast, our proposed path-length-aware shift mapping produces the best results across different scenarios, as it ensures that reused paths remain valid across neighboring pixels, even when they lie on different ellipsoids. Results for more scenes, including convergence plots, are shown in Fig. 13. Our method consistently achieves the best performance, while naive reuse can even perform worse than plain path tracing due to its low reuse validity rate.

Effect of Time Gate Width. Similar to Pediredla et al. [2019b], we compare results under different time-gate widths $\Delta\tau$ in Fig. 11. All images are normalized by the time-gate width to ensure comparable intensity levels. The error for different $\Delta\tau \in [0.005, 1.0]$ is plotted in Fig. 11-(A). We find that our method is most effective for narrow time gates. As the time gate widens, naive path reuse is less likely to violate the path-length constraint, making the additional computation for path-length correction less beneficial. In Fig. 11-(B), we also report the path reuse success rate for both naive reuse and our method. For very small time gates, naive reuse achieves a very low success rate, which gradually increases as the time gate widens. In contrast, our method maintains a nearly constant and high reuse success rate across different time-gate widths. Notably, for very large time gate, the success rate of our method becomes slightly lower than that of naive reuse. This reduction is primarily due to failures in Newton's iteration or additional occlusion introduced when moving the reconnection vertex. Overall, our method is most beneficial for narrow time gates, whereas naive reuse is sufficient for wider gates. The exact threshold depends on the scene, but we

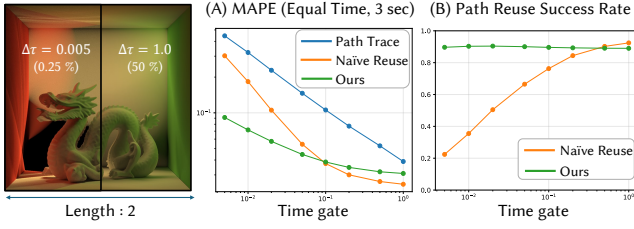


Fig. 11. (A) MAPE comparison under different time-gate widths within a 3-second time budget. (B) Comparison of path reuse success rates between naive reuse and our method. Overall, our method performs best when the time gate is sufficiently narrow, 5 – 10% of the scene size.

observe that $\Delta\tau$ below approximately 5 – 10% of the scene scale serves as a practical boundary.

Gauge Direction Comparison. We demonstrate the effectiveness of the proposed gauge in Fig. 12. We evaluate horizontal, vertical, and average-gradient directions under different coordinate systems, including a local plane and a unit hemisphere⁴. Each method is evaluated under an equal-time budget with $\Delta\tau = 0.01$. We observe that, for each region, one fixed axis typically performs better than the other in terms of error, depending on the scene geometry. However, regardless of the coordinate system, the gradient-based direction achieves the best performance on average. We also report the moving distance of the reconnection vertex ($\|p - S(p)\|$) induced by each method and find that our gradient-based approach consistently results in the shortest displacement. In fact, moving distance is not a perfect indicator of path-throughput similarity, since shorter displacements may break the similarity due to newly appeared occlusions. Still, we could observe a clear overall correlation between shorter moving distances and reduced variance. Regarding the performance, we could not find a noticeable difference when evaluating additional Hessians from using the gradient gauge direction.

Newton’s Iteration Statistics. We also report statistics for Newton’s iteration. We set the convergence threshold to 1% of the time-gate width and limit the maximum number of iterations to 5. Consistent with the observations of Jakob and Marschner [2012], we find that approximately two to three Newton iterations are sufficient to reach a valid solution, with a failure rate below 10 – 15%. The overall statistics are summarized in Tab. 2. Here, *Newton SR* denotes the Newton’s iteration success rate and *Actual SR* denotes the actual path reuse success rate, which is consistently lower, as it excludes samples that are occluded (from p_1) or rejected due to excessive Jacobian factors.

Initialization with Ellipsoidal Path Connection. So far, we have used direct connection for initial sample generation, but ellipsoidal path connection [Pediredla et al. 2019b] can also be employed for this purpose, producing higher-quality initial samples. The results are shown in Fig. 13. We present equal-time comparisons for different strategies, along with convergence plots with respect to both

⁴Each coordinate system has its own advantages and limitations, and we employ heuristics that leverage both. See the supplementary material for details.

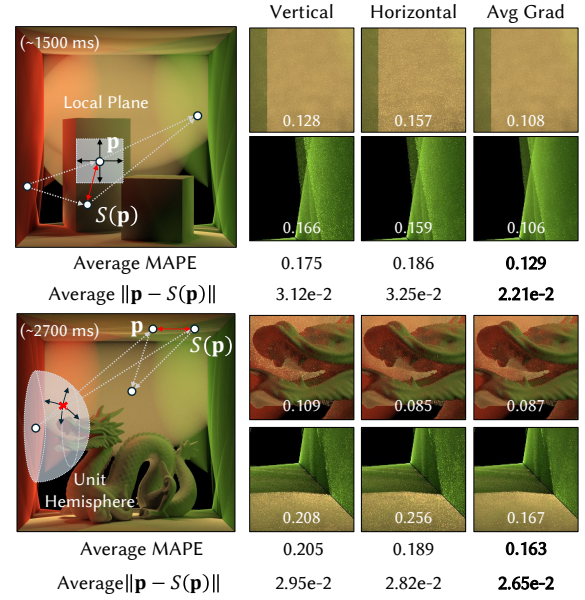


Fig. 12. Comparison of gauge directions using fixed horizontal and vertical axes, and the proposed average-gradient direction. Our method consistently produces the best results, largely due to the reduced displacement.

Table 2. Newton iteration counts and success rates across different scenes. The actual success rate accounts for occlusion and Jacobian factors and is always lower than the Newton success rate.

Scene	# of Iter.	Newton SR	Actual SR
CORNELL-BOX	1.5267	0.9652	0.9356
CORNELL-DRAGON	1.6499	0.9454	0.9027
VEACH-AJAR	1.5549	0.9578	0.9026
KITCHEN	1.9003	0.9197	0.8127
BEDROOM	1.9854	0.9199	0.8331
STAIRCASE	2.2179	0.8664	0.7238
BISTRO	2.3480	0.8455	0.6638

spp and time. Compared to direct connection, ellipsoidal connection yields better results, although the improvement depends on the scene. Importantly, we find that regardless of the initial sampling strategy, our method consistently outperforms the baseline methods. The computational cost of ellipsoidal path connection is significantly higher than that of direct connection—by a factor of 5× to 10×, depending on the scene—which often necessitates the use of narrower time gates to clearly observe its advantage. Our heuristic-based ellipsoidal sampling strategy may be suboptimal, as it is not the primary focus of this paper. Designing more efficient ellipsoidal connection algorithms that are well-suited for GPU execution is an interesting direction for future work.

6.2 Evaluation of Path Length Shrink Mapping

In this subsection, we evaluate the proposed path-length shrink mapping as an efficient strategy for initial sample generation under

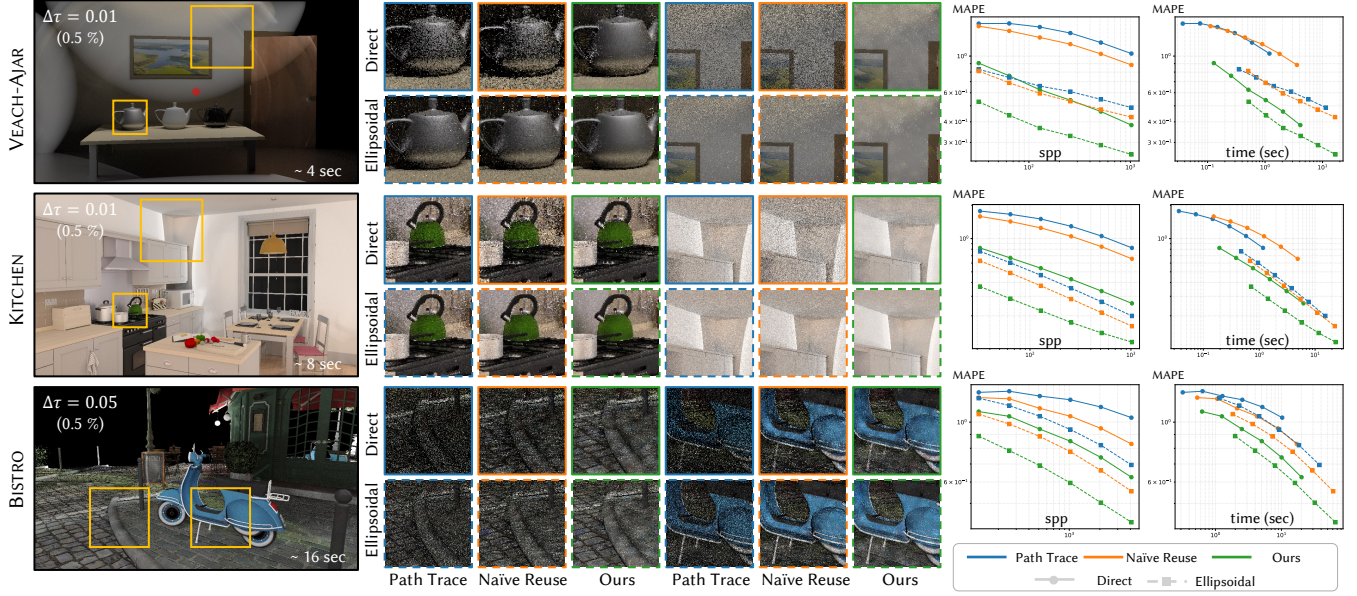


Fig. 13. We implement ellipsoidal path connection [Pediredla et al. 2019b], adapted to a GPU environment, and use it as an initial candidate generation. Compared to a direct connection, an ellipsoidal path connection produces better initial samples at the cost of increased computation time. Our method improves the result regardless of the initial candidate generation method. The light source is collocated with the camera. For the VEACH-AJAR scene, a collimated laser is used, while a point light is used for the others. Image size is 960×540 , and max bounce is set to 6.

narrow time gates. To clearly show the variance-bias behavior, we disable spatial reuse in this experiment. We focus on two parameters: the relative width of the rough time gate, defined as $K = \frac{\Delta\tau_{\text{rough}}}{\Delta\tau_{\text{fine}}}$, and the fraction of samples allocated to the rough time gate, defined as $R = \frac{M_{\text{rough}}}{M_{\text{rough}} + M_{\text{fine}}}$. Fig. 14-(A) reports the MAPE for different values of $K \in [1.0, 500]$ and $R \in [0.0, 1.0]$, evaluated at a fixed fine time gate of $\Delta\tau = 0.01$. Overall, we observe that using a moderately larger rough time gate and allocating all samples to the rough gate yields the best metric across different scenes. In Fig. 14-(B), we fix $R = 1.0$ and vary the value of K . When the rough time gate is too small, the benefit of using a wider gate is limited, and variance dominates the result. On the other hand, when the rough time gate becomes excessively large, the result becomes biased and typically darker than the ground truth image, since the region mapped from the rough time gate becomes smaller than the original fine gate. The MAPE plot in Fig. 14-(A) suggests that an acceptable bias-variance trade-off occurs for K values between approximately 10 and 100. This range is scene dependent and also varies with the chosen time-gate width.

We further examine the effect of sample allocation in Fig. 14-(C) by varying R while keeping K fixed. As discussed earlier, setting $R < 1$ and applying MIS between canonical samples from the fine time gate and candidate samples from the rough time gate produces an unbiased estimator. Setting $R = 1$ completely ignores canonical samples, resulting in a biased estimate that is typically darker than the reference image, but with substantially reduced variance. Despite being biased, this approach does not introduce the blurring artifacts observed when simply increasing the time-gate width alone

(rightmost column of Fig. 14-(C)). This improvement arises because shrink mapping explicitly attempts to match path lengths, whereas merely widening the time gate does not.

Performance Comparison with Ellipsoidal Path Connection. Since shrink mapping is designed to address sampling difficulties under narrow time gates without expensive scene-ellipsoid intersections, we compare it with ellipsoidal path connection in Fig. 15. We use $R = 1$ and $K = 10$. Under an equal-spp constraint, ellipsoidal path connection achieves the best performance, as it guarantees valid time-gated path samples. However, when compared under equal-time budgets, our shrink mapping demonstrates an advantage, especially in complex scenes where ellipsoidal sampling incurs huge overhead due to scene-ellipsoid intersection tests. We also show shrink mapping with spatial reuse (red curve), which can further improve performance, although it remains biased due to $R = 1$.

In summary, while shrink mapping does not guarantee exact path-length correctness in the same way as analytic ellipsoidal intersection, it offers a substantial performance advantage, particularly at $R = 1$, at the cost of bias, but without introducing blurring.

6.3 Real-time Time-Gated Rendering with Temporal Reuse

Path-length-aware reuse can also be applied to temporal reuse across frames. Following Lin et al. [2022], we use a single iteration of spatial reuse with three neighbor selections and set $M_{\text{cap}} = 5-20$ (maximum confidence in Eq. (9)), while generating multiple initial paths per frame. As in other ReSTIR-based methods, we exploit motion vector information to determine the corresponding pixel from which to fetch the reservoir in the previous frame. For simplicity, we use

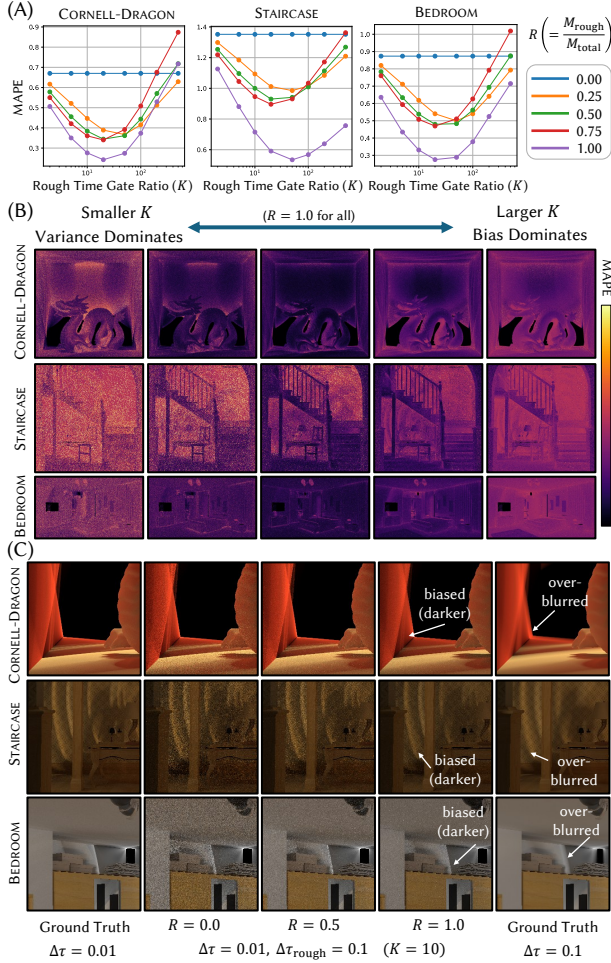


Fig. 14. Results of path-length shrink mapping using a rough time gate. (A) When the rough time gate is too small, little benefit is gained from widening the gate, and variance dominates, whereas excessively large rough time gates introduce bias. (B) Visually acceptable results are typically obtained when the rough time gate is approximately 10–20 times wider than the fine time gate. (C) Simply increasing the time-gate width leads to over-blurring of scene details.

direct connection for initial samples here, but it can be combined with ellipsoidal connection or shrink mapping described earlier.

Real-time Transient Rendering. We can exploit temporal reuse for real-time transient simulation in static scenes, producing transient images in a streaming fashion. This is less common in the ToF community than in gaming or real-time rendering, but it is still useful for rapid inspection, as constructing a full histogram requires long accumulation times. We visualize a real-time transient rendering result for the CLASSROOM scene in Fig. 16-(A).

Dynamic Scene. We further demonstrate time-gated rendering in dynamic scenes, which is relevant for simulating proximity cameras or transient sensors that form image sequences by sweeping the time gate. In Fig. 16-(B), we simulate a proximity-sensing drone equipped

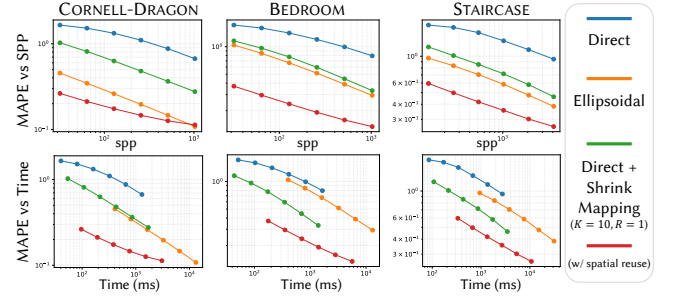


Fig. 15. Comparison between ellipsoidal path connection, direct connection, and direct connection combined with shrink mapping. Under an equal-spp constraint, an ellipsoidal path connection achieves the best quality. However, for complex scenes where ellipsoidal sampling becomes expensive, our method provides a more favorable trade-off under an equal-time budget. We also show shrink mapping with spatial reuse as a red line.

with a fixed time-gated sensor and a collocated point light hovering within the BEDROOM scene. Our method produces substantially improved results compared to path tracing and naive reuse in this dynamic setting, and can even faithfully reproduce mirror-reflected regions. We also evaluate the NLOS-BUNNY scene in Fig. 16-(C). In this scene, the camera remains fixed while a collimated laser beam scans across the back wall, indicated by the red dot in the ground-truth image. The time gate also shifts as the laser moves. Again, our method yields superior results compared to baseline approaches. We also demonstrate a downstream application for object shape reconstruction under a similar scene setting, which is described in detail in Sec. 7.1.

6.4 Evaluation of Transient Histogram Rendering

We also evaluate our algorithm on transient histogram rendering, as discussed in Sec. 4.5. In this setting, the goal is to render a sequence of transient images simultaneously, producing an output histogram of size (H, W, B) per frame. An example of transient histogram rendering is shown in Fig. 16-(D), where the camera moves forward in the VEACH-AJAR scene. We use a histogram bin count of 64, and to limit reuse cost, we apply temporal reuse only. We find that our method produces higher-quality results for transient histogram rendering compared to standard path tracing and naive reuse.

Cost-to-Performance Evaluation with Different B . We evaluate the effectiveness of path reuse while accounting for its computational cost under different values of B . Specifically, we measure rendering error, execution time, and their product as a simple indicator of cost-to-performance efficiency. For the metric error $\times$ time, lower values indicate better cost-to-performance. Since second-order error decreases linearly over time, we use relative MSE as the error metric. We evaluate several path reuse configurations, including temporal reuse, bin-wise reuse, and spatial reuse. All experiments are conducted with $B \in [4, 64]$, while keeping the temporal width of each histogram bin fixed. Each configuration is rendered for 100 frames with an animated camera, and the same spp is used across all experiments.

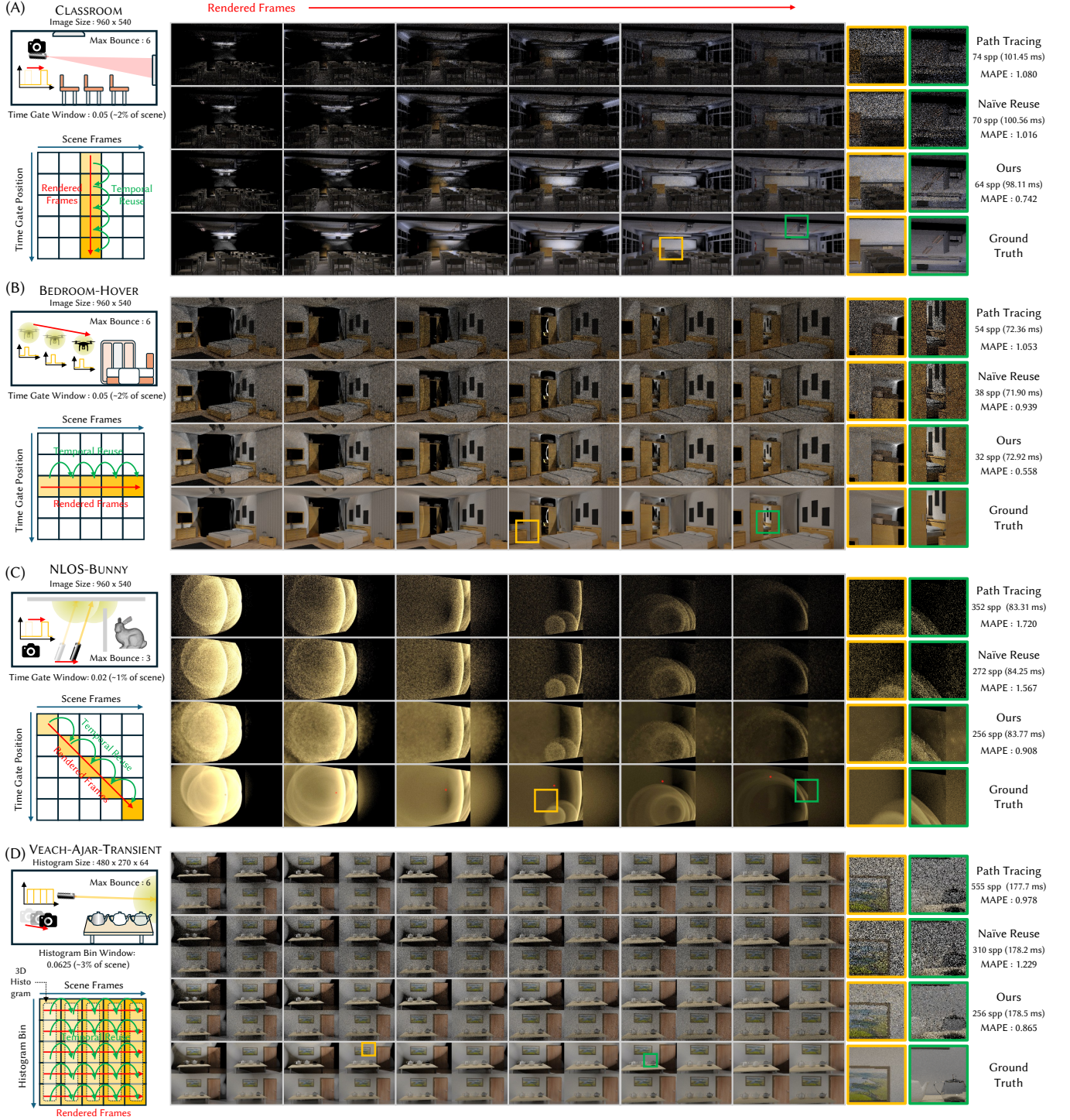


Fig. 16. Four scenarios demonstrating the proposed path temporal reuse. (A) Streamed transient rendering in a static CLASSROOM scene. (B) Dynamic proximity camera simulation in a BEDROOM scene. (C) Transient NLOS imaging with a time-gated camera and a moving collimated laser; this setup is also used in later sections. (D) Transient histogram rendering with the number of bins set to $B = 64$. Across all scenarios, our method consistently produces the highest-quality results.

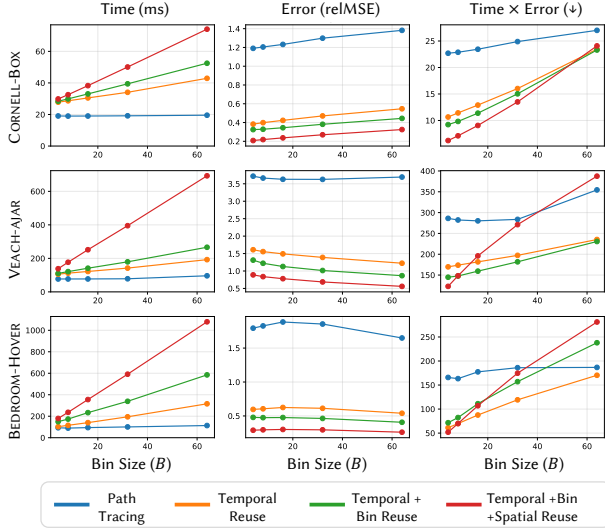


Fig. 17. Comparison of path reuse in transient histogram rendering with different B s. We could find that the cost-to-performance efficiency of path reuse diminishes as B becomes larger.

The results are shown in Fig. 17. The first column reports rendering time. As expected, the cost of naive path tracing remains nearly constant even B increases, whereas the cost of path reuse grows linearly with B . Configurations with more aggressive reuse exhibit steeper slopes due to higher reuse overhead. The second column shows the relative MSE averaged over all bins. The error remains nearly constant across different values of B , with minor variations depending on the histogram bin range. Notably, more aggressive path reuse generally yields lower error. The third column presents the cost-to-performance metric computed as time multiplied by error. For small values of B , aggressive path reuse is clearly beneficial, as the reuse overhead is relatively small compared to the resulting error reduction. As B increases, the benefit of path reuse diminishes, and beyond a certain point, disabling path reuse becomes more advantageous. We further observe that this trend is strongly influenced by the path reuse cost C_{reuse} . The CORNELL-BOX scene exhibits the lowest reuse cost, while VEACH-AJAR incurs a higher cost due to its textured materials. The BEDROOM scene shows the highest reuse cost, as the collocated moving light source requires updating the path throughput beyond the reconnection vertex.

In summary, increasing B reduces the benefit of complex path reuse strategies, and in practice, a limited amount of reuse (e.g., temporal only) is often preferable.

Comparison with Biased Transient Rendering. We compare our method with two previous transient rendering approaches: progressive kernel density estimation (KDE) [Jarabo et al. 2014] and the radiosity-based linear triangle approximation [Iseringhausen and Hullin 2020]. For progressive KDE, we use the Epanechnikov kernel with a kernel-width reduction rule $\Delta\tau_{j+1}/\Delta\tau_j = (j+\alpha)/(j+1)$, with $\alpha = 0.8$, following Jarabo et al. [2014]. For the triangle approximation, we subdivide the scene geometry to obtain a sufficiently fine

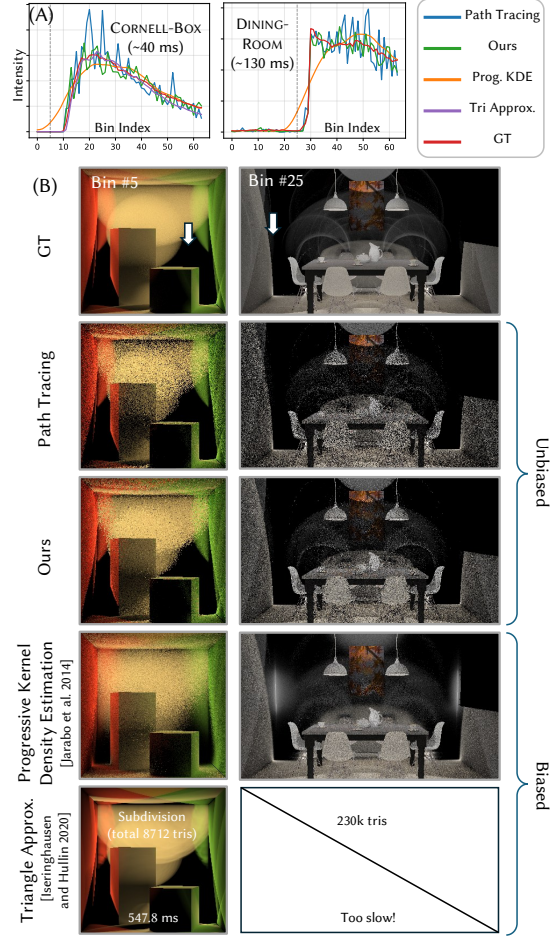


Fig. 18. Comparison of our method with progressive KDE [Jarabo et al. 2014] and the linear triangle approximation [Iseringhausen and Hullin 2020], both of which are biased. Progressive KDE produces less noisy results but exhibits noticeable temporal blurring. The triangle approximation works only for simple scenes and does not scale to complex geometry. The transient profile at a specific point is shown in (A), and the full image is shown in (B).

mesh (8,712 triangles for CORNELL-BOX). For our method, we apply temporal reuse only. All methods are evaluated over 100 frames using $B = 64$ and the same time budgets, except for the triangle approximation. We visualize the transient profile at a specific point and the full image in Fig. 18, at a specific frame. Compared to unbiased methods, progressive KDE produces less noisy images at the cost of introducing bias, primarily in the form of temporal blurring. As shown in Fig. 18-(A), this bias manifests as non-zero values in regions where the ground truth is zero. The triangle approximation requires heavy tessellation and takes 547.8 ms even for the simple CORNELL-BOX scene, making it impractical for more complex scenes such as DINING-ROOM, which contains approximately 230k triangles. Moreover, it does not support multi-bounce light transport.

Our method is unbiased and produces higher-quality results than naive path tracing under the same time budget. While the results

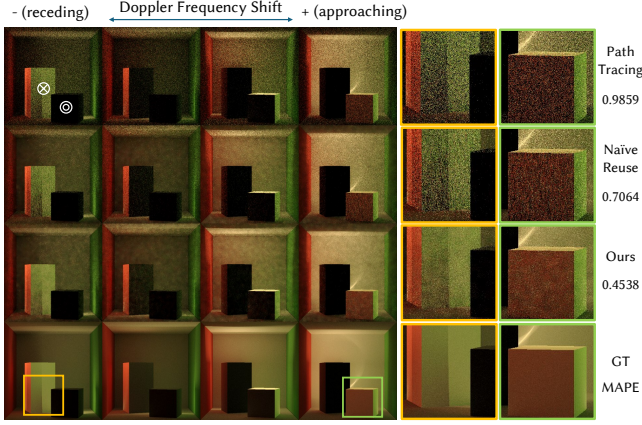


Fig. 19. Doppler frequency shift rendering under different frequencies. The large box is receding from the camera, resulting in a negative frequency shift, while the small box is approaching the camera, producing a positive frequency shift. Overall, our proposed algorithm also performs well under path-velocity constraints.

remain noisier than those of biased methods, this gap highlights opportunities for future improvement.

6.5 Evaluation of Doppler Frequency Rendering

Finally, we demonstrate a simple example of Doppler frequency rendering in Fig. 19. In this scene, the large box is receding from the camera, while the small box is approaching it. Each image is rendered with a 2-second budget in an offline style, with only spatial reuse performed. Across different Doppler frequency shifts, our method produces improved results both qualitatively and quantitatively, exhibiting performance gains similar to those observed for the path-length constraint.

7 DOWNSTREAM APPLICATIONS

Our renderer can be easily deployed to test downstream applications. This capability has the potential to reduce the replication crisis in computational imaging while also accelerating the software development of imaging systems. Below, we demonstrate two applications of the renderer for downstream tasks.

7.1 NLOS Reconstruction with Temporal Focusing

We first simulate NLOS detection with temporal focusing, similar to Pediredla et al. [2019a], but with a small change to allow for better signal processing. The scene configuration is illustrated in Fig. 20–(A). In this setup, the target object is hidden behind an occluder and is not directly visible to the imaging system, which can directly see only a diffuse relay wall. The objective is to recover the voxel reflectance (albedo) of the hidden object by exploiting indirect multi-bounce reflections and ToF measurements. We will first describe the imaging setup and then the rendering strategy.

Imaging setup. In Pediredla et al. [2019a]’s setup, the imaging system consists of a single-pixel time-gated sensor (D) and a laser source (S) that are colocated. Instead of a single-pixel detector, we

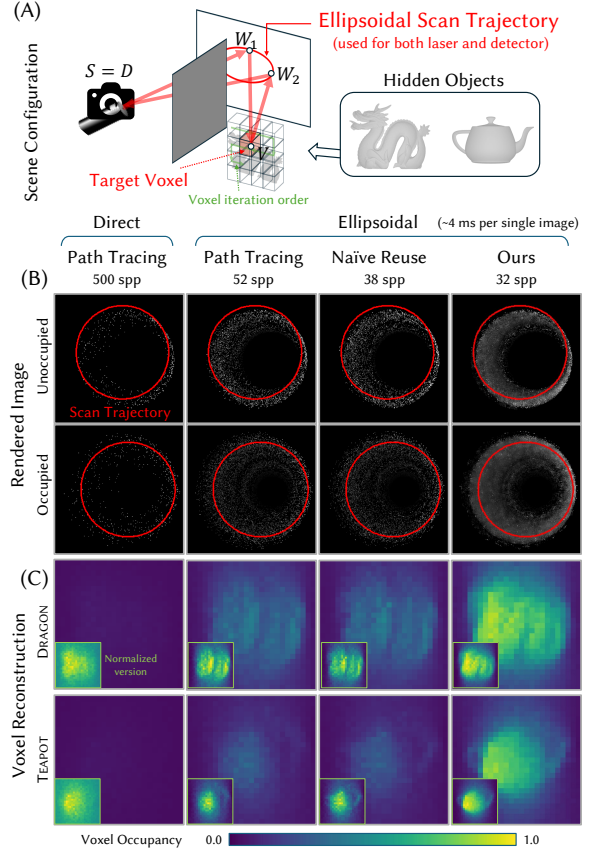


Fig. 20. Simulation of NLOS imaging with temporal focusing [Pediredla et al. 2019a] using our rendering pipeline. (A) Scene configuration for detecting a hidden object behind a wall. Both the laser and the sensor scan ellipsoidal trajectories on the visible wall to determine whether a target voxel is occupied. (B) Example rendered image, and (C) reconstructed voxel grids (the version normalized by the maximum value is shown in the bottom-left), obtained using different methods under a time budget of 4 ms per image, demonstrating the advantage of our proposed method.

assume an array time-gated sensor such as an ICCD. The key idea of Pediredla et al. [2019a] is to temporally focus the illumination and backprojected imaging rays onto a single voxel V and image the NLOS scene by scanning one voxel at a time.

First, consider all light paths that temporally focus on a single voxel V at time gate τ_1 . Light emitted from the laser first reflects off the visible relay wall and then arrives at V . Therefore, points W_1 on the relay wall that lead to temporal focusing at V satisfy

$$\|S - W_1\| + \|W_1 - V\| = \tau_1. \quad (29)$$

Similarly, light reflected from V will focus on the detector at time τ_2 after reflecting off wall points W_2 that satisfy

$$\|V - W_2\| + \|W_2 - D\| = \tau_2. \quad (30)$$

Both Eq. (29) and Eq. (30) are equations of an ellipse; hence, by illuminating and imaging an ellipse on the relay wall, we can compute the voxel reflectance. In Pediredla et al. [2019a], $\tau_1 = \tau_2 = \tau$.

This technique only sums imaging rays along the ellipse. Instead, by using an array detector, we can treat all points along the imaging ellipse independently and develop improved signal-processing techniques for accurate NLOS scene estimation. Specifically, we exploit the observation that the entire imaging ellipse must be visible at the specific time gate τ , and thus we take the geometric mean of the ellipsoidal points. The geometric mean yields a better signal-to-background ratio than the arithmetic mean.

Rendering setup. To simulate this process using our renderer, we render multiple images while steering the laser toward different positions along the ellipsoidal trajectory. In each rendered image, we take the geometric mean along the imaging ellipse to estimate the voxel reflectance. The target time range is set to 2τ , and τ is chosen to be slightly larger than $\|S - C\| + \|C - V\|$, where C denotes the center of the visible wall. An example of the rendered image and the corresponding sensor ellipsoidal trajectory for unoccupied and occupied voxels is shown in Fig. 20–(B).

We compare four different methods: path tracing with direct connection, path tracing with ellipsoidal connection, ellipsoidal connection with naive path reuse, and ellipsoidal connection with our proposed method. All images are rendered at a resolution of 256×256 under a time budget of 4 ms per image. The laser is scanned over 16 positions, and we scan 25×25 NLOS voxels, resulting in a total of $16 \times 25 \times 25$ rendered images. The time-gate width is set to 0.01, corresponding to less than 1% of the scene scale. We apply both temporal and spatial reuse for reuse-based methods. To ensure frame-to-frame similarity between rendered images (which makes temporal reuse useful), we scan voxels in a continuous order, as indicated by the bright green trajectory in Fig. 20–(A).

The voxel reconstruction results are shown in Fig. 20–(C). As observed by Pediredla et al. [2019a], direct connection performs poorly in this scenario because most rays do not contribute to the target voxels. Ellipsoidal path connection improves reconstruction quality, but the results remain noisy. Naive path reuse provides some improvement, such as partially recovering the teapot spout. In contrast, our method produces the most accurate reconstruction, owing to the significantly reduced noise in the rendered images, as shown in Fig. 20–(B). We use a simple scene configuration here, but more challenging scenarios, such as scenes with multiple objects, non-diffuse materials, and higher-order bounces, could be explored in future evaluations.

7.2 NLOS Navigation

We also simulate an inference-based NLOS navigation scenario similar to Young et al. [2025]. We consider an L-shaped road with an obstacle located in an occluded region. The goal is to detect the hidden object from global transient illumination and enable safe, fast, and robust path planning. The overall scene configuration is illustrated in Fig. 21–(A).

The camera is initially positioned at $(x, z) = (0, 1)$ and moves toward the $(0, -1)$ direction, starting from a location outside the field of view. As it advances, the system continuously infers the location of the hidden object and, at a predefined point ($z = 0.5$), performs path planning based on the accumulated estimates. For simplicity, we infer only the x -coordinate of the object, assuming

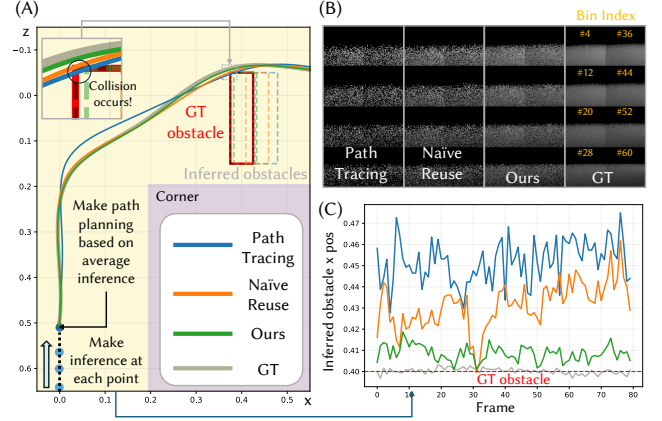


Fig. 21. We simulate NLOS navigation [Young et al. 2025] using our rendering pipeline. (A) Overall scene configuration, where a camera with a collocated light source moves forward and captures transient histograms. (B) Rendered transient image used for inference. (C) Inferred hidden object position. When the camera reaches $z = 0.5$, path planning is performed based on the averaged object inference. Compared to path tracing and naive reuse, which result in collisions with the true obstacle, our method achieves more accurate object localization and consequently more robust path planning.

that the z -coordinate and object size are known. The camera and laser are collocated, with the laser having a small opening angle and a slight downward tilt toward the ground.

We train a lightweight convolutional neural network (CNN) for object location inference. The network takes as input the camera position, viewing direction, and transient histogram data, and outputs the estimated x -coordinate of the object. Training is performed using ground-truth rendered data with randomly sampled object positions, while inference uses noisy data generated by different rendering methods. An equal time budget of 10 ms per frame is enforced for all methods. We use temporal reuse only. The per-frame histogram resolution is 64×64 with $B = 64$. The CNN consists of two convolutional layers followed by a fully connected layer.

The rendered transient histograms and the corresponding inference results are shown in Fig. 21–(B) and (C), respectively. In this experiment, the object is located at $x = 0.4$. Path tracing and naive reuse produce inaccurate estimates due to noise in the rendered histograms. In contrast, our method yields more accurate inference results by reducing noise through temporal path reuse.

The accuracy of object inference directly affects path planning quality. We visualize the ground-truth obstacle position and the planned paths produced by each method using the A* algorithm with spline smoothing. Because path tracing and naive reuse predict the object to be much farther than its true position, the resulting paths collide with the actual obstacle. In contrast, our method produces a more accurate estimate of the object location, enabling safer and more robust navigation. While we focus on a simple scenario here, more advanced navigation strategies and more complex environments could be explored using our simulation framework in future work.

8 DISCUSSION AND CONCLUSION

In this manuscript, we introduce a path-length-aware shift mapping framework that enables effective spatio-temporal sample reuse for time-of-flight (ToF) rendering. By explicitly enforcing optical path-length constraints during reuse, our method overcomes a key limitation of steady-state ReSTIR when applied to time-resolved imaging, where naive reuse often produces temporally invalid contributions. The proposed approach supports a wide range of ToF rendering scenarios, including offline time-gated rendering, interactive transient rendering, transient histogram rendering in dynamic scenes, and extensions to Doppler frequency simulation. We further demonstrate two downstream applications—shape reconstruction and navigation—highlighting the potential of our renderer for interactive and decision-driven ToF sensing systems.

Limitations and Future Works. We discuss the limitations of our approach and outline potential directions for future research.

First, we use the reconnection vertex in the ReSTIR-PT framework as the adjustment vertex for path-length-aware shift mapping. More sophisticated heuristics for selecting this vertex—such as incorporating geometric or material properties of adjacent vertices p_1 and p_2 —could improve both robustness and efficiency. Alternatively, instead of perturbing a single vertex, one could distribute the path-length adjustment across the entire path to minimize per-segment changes, although this would incur additional cost due to updating the full path. Moreover, our method primarily enforces path-length constraints and does not explicitly account for specular manifolds. In scenes where both temporal and specular constraints are important, a joint manifold formulation would be required, and designing efficient strategies for such mixed constraints remains an open challenge.

We currently employ a basic Newton-based solver for manifold traversal, but more advanced variants from steady-state caustics rendering [Fan et al. 2024; Hong et al. 2025] could be adapted to improve convergence and stability. Also, while we select the update direction based on the gradient to minimize vertex displacement, this heuristic does not always guarantee valid reuse due to visibility changes (e.g., the discrepancy between Newton success rate and actual reuse success rate in Tab. 2). Developing more robust direction-selection strategies that incorporate visibility and geometric constraints is an important direction for future work.

Our current implementation also assumes delta light sources, which precludes searching over extended light source domains. For finite-area light sources, the constraint simplifies to the distance between the previous vertex and the light source, allowing sampling directly over the emitter surface. Also in this setting, spherical connections are more appropriate than ellipsoidal ones, as discussed by Pediredla et al. [2019b].

Finally, there is substantial room for improvement in transient histogram rendering. In this work, we directly apply path-length-aware shift mapping to each histogram bin. However, since transient histograms involve continuous time bins rather than strict delta manifolds, explicitly enforcing path-length constraints via root finding may be unnecessary. Time-bin splatting strategies, similar to those proposed by Liu et al. [2025], could provide a more efficient alternative. A remaining challenge is the design of effective backup

sampling strategies for time bins that receive no prior samples. Developing more scalable ReSTIR-based algorithms for large numbers of time bins B is also an important direction for future work.

ACKNOWLEDGMENTS

We acknowledge the support of the Visual Computing Seminar and colleagues from the RISC and VCL labs for their helpful feedback on this project. This work was supported in part by the National Science Foundation under Awards 2403122 and 2326904.

REFERENCES

- A Adam, C Dann, O Yair, S Mazor, and S Nowozin. 2016. Bayesian Time-of-Flight for Realtime Shape, Illumination and Albedo. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2016).
- Marco Ament, Christoph Bergmann, and Daniel Weiskopf. 2014. Refractive Radiative Transfer Equation. *ACM Transactions on Graphics* 33, 2 (April 2014), 17:1–17:22. <https://doi.org/10/gbf323>
- Ian M Baker, Stuart S Duncan, and Jeremy W Copley. 2004. A low-noise laser-gated imaging system for long-range target identification. In *Infrared Technology and Applications XXX*, Vol. 5406. SPIE, 133–144.
- Benedikt Bitterli. 2016. Rendering Resources. <https://benedikt-bitterli.me/resources/>
- Benedikt Bitterli, Chris Wyman, Matt Pharr, Peter Shirley, Aaron Lefohn, and Wojciech Jarosz. 2020. Spatiotemporal Reservoir Resampling for Real-Time Ray Tracing with Dynamic Direct Lighting. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 39, 4 (July 2020). <https://doi.org/10/gg8xc7>
- Mauro Buttafava, Jessica Zeman, Alberto Tosi, Kevin Eliceiri, and Andreas Velten. 2015. Non-Line-of-Sight Imaging Using a Time-Gated Single Photon Avalanche Diode. *Optics Express* 23, 16 (2015). <https://doi.org/10/gfz5k4>
- Lucrezia Cester, Ashley Lyons, Maria Chiara Braidotti, and Daniele Faccio. 2019. Time-of-Flight Imaging at 10 ps Resolution with an ICCD Camera. *Sensors* 19, 1 (2019), 180.
- Min-Te Chao. 1982. A General Purpose Unequal Probability Sampling Plan. *Biometrika* 69, 3 (Dec. 1982), 653–656. <https://doi.org/10/fd87zs>
- Zhimin Fan, Jie Guo, Yiming Wang, Tianyu Xiao, Hao Zhang, Chenxi Zhou, Zhenyu Chen, Pengpei Hong, Yanwen Guo, and Ling-Qi Yan. 2024. Specular Polynomials. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–13.
- Zhimin Fan, Pengpei Hong, Jie Guo, Changqing Zou, Yanwen Guo, and Ling-Qi Yan. 2023. Manifold path guiding for importance sampling specular chains. *ACM Transactions on Graphics (TOG)* 42, 6 (2023), 1–14.
- Genevieve Gariepy, Nikola Krstajić, Robert Henderson, Chunyong Li, Robert R Thomson, Gerald S Buller, Barmak Heshmat, Ramesh Raskar, Jonathan Leach, and Daniele Faccio. 2015. Single-Photon Sensitive Light-in-Fight Imaging. *Nature Communications* 6 (2015).
- Ioannis Gkioulekas, Anat Levin, Frédo Durand, and Todd Zickler. 2015. Micron-Scale Light Transport Decomposition Using Interferometry. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 34, 4 (2015). <https://doi.org/10/f7mzkt>
- Yoav Grauer and Ezri Sonn. 2015. Active gated imaging for automotive safety applications. In *Video Surveillance and Transportation Imaging Applications 2015*, Vol. 9407. SPIE, 112–129.
- Johannes Hanika, Anton Kaplanyan, and Carsten Dachsbacher. 2015. Improved Half Vector Space Light Transport. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 34, 4 (July 2015), 65–74. <https://doi.org/10/gfzv83>
- Trevor Hedstrom, Markus Kettunen, Daqi Lin, Chris Wyman, and Tzu-Mao Li. 2025. ReSTIR BDPT: Bidirectional ReSTIR Path Tracing with Caustics. *ACM Transactions on Graphics* (2025).
- Felix Heide, Matthias B. Hullin, James Gregson, and Wolfgang Heidrich. 2013. Low-Budget Transient Imaging Using Photonic Mixer Devices. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 32, 4 (July 2013), 45:1–45:10. <https://doi.org/10/gbdg7s>
- Robert K Henderson, Nick Johnston, Francescopaolo Mattioli Della Rocca, Haochang Chen, David Day-Uei Li, Graham Hungerford, Richard Hirsch, David Mcloskey, Philip Yip, and David JS Birch. 2019. A 192x128 time correlated SPAD image sensor in 40-nm CMOS technology. *IEEE Journal of Solid-State Circuits* 54, 7 (2019), 1907–1916.
- Pengpei Hong, Meng Duan, Beibei Wang, Cem Yuksel, Tizian Zeltner, and Daqi Lin. 2025. Sample Space Partitioning and Spatiotemporal Resampling for Specular Manifold Sampling. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*. 1–10.
- Binh-Son Hua, Adrien Gruson, Victor Petitjean, Matthias Zwicker, Derek Nowrouzezahrai, Elmar Eisemann, and Toshiya Hachisuka. 2019. A Survey on Gradient-Domain Rendering. *Computer Graphics Forum (Proceedings of Eurographics State of the Art Reports)* 38, 2 (2019), 455–472. <https://doi.org/10/ggd8m5>
- Julian Iseringhausen and Matthias B. Hullin. 2020. Non-Line-of-Sight Reconstruction Using Efficient Transient Rendering. *ACM Transactions on Graphics* 39, 1 (Jan. 2020),

- 8:1–8:14. <https://doi.org/10/gspq3m>
- Wenzel Jakob and Steve Marschner. 2012. Manifold Exploration: A Markov Chain Monte Carlo Technique for Rendering Scenes with Difficult Specular Transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 31, 4 (July 2012), 58:1–58:13. <https://doi.org/10/gfzq4p>
- Wenzel Alban Jakob. 2013. *Light Transport on Path-Space Manifolds*. Ph. D. Dissertation. Cornell University.
- Adrian Jarabo. 2012. *Femto-Photography: Visualizing Light in Motion*. M.Sc. Thesis. Universidad de Zaragoza.
- Adrian Jarabo, Julio Marco, Adolfo Munoz, Raul Buisan, Wojciech Jarosz, and Diego Gutierrez. 2014. A Framework for Transient Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 33, 6 (Nov. 2014), 177:1–177:10. <https://doi.org/10/gfznb8>
- Adrian Jarabo, Belen Masia, Julio Marco, and Diego Gutierrez. 2017. Recent Advances in Transient Imaging: A Computer Graphics and Vision Perspective. *Visual Informatics* 1, 1 (2017), 65–79. <https://doi.org/10/gfz5mh>
- Wojciech Jarosz, Derek Nowrouzezahrai, Robert Thomas, Peter-Pike Sloan, and Matthias Zwicker. 2011. Progressive Photon Beams. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 30, 6 (Dec. 2011), 181:1–181:12. <https://doi.org/10/fn5x2j>
- Achuta Kadambi, Jamie Schiel, and Ramesh Raskar. 2016. Macroscopic Interferometry: Rethinking Depth Estimation with Frequency-Domain Time-of-Flight. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Simon Kallweit, Petrik Clarberg, Craig Kolb, Tomáš Davidović, Kai-Hwa Yao, Theresa Foley, Yong He, Lifan Wu, Lucy Chen, Tomas Akenine-Möller, Chris Wyman, Cyril Crassin, and Nir Benty. 2022. The Falcor Rendering Framework. <https://github.com/NVIDIAGameWorks/Falcor>
- Anton S. Kaplanyan, Johannes Hanika, and Carsten Dachsbacher. 2014. The Natural-Constraint Representation of the Path Space for Efficient Light Transport Simulation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 33, 4 (July 2014), 102:1–102:13. <https://doi.org/10/f6cz85>
- Markus Kettunen, Marco Manzi, Miika Aittala, Jaakko Lehtinen, Frédo Durand, and Matthias Zwicker. 2015. Gradient-domain path tracing. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–13.
- Juhyeon Kim, Craig Benko, Magnus Wrenninge, Ryusuke Villemin, Zeb Barber, Wojciech Jarosz, and Adithya Pediredla. 2025. A Monte Carlo Rendering Framework for Simulating Optical Heterodyne Detection. *ACM Transactions on Graphics (TOG)* 44, 4 (2025), 1–19.
- Juhyeon Kim, Wojciech Jarosz, Ioannis Gkioulekas, and Adithya Pediredla. 2023. Doppler Time-of-Flight Rendering. *ACM Transactions on Graphics (TOG)* 42, 6 (2023), 1–18.
- Ahmed Kirmani, Tyler Hutchison, James Davis, and Ramesh Raskar. 2011. Looking around the Corner Using Ultrafast Transient Imaging. *International Journal of Computer Vision (IJCV)* 95, 1 (2011). <https://doi.org/10/dsk2dt>
- Jonathan Klein, Christoph Peters, Jaime Martin, Martin Laurenzis, and Matthias B. Hullin. 2016. Tracking Objects Outside the Line of Sight Using 2D Intensity Images. *Scientific Reports* 6 (2016). <https://doi.org/10/f82649>
- Martin Laurenzis and Andreas Velten. 2014. Nonline-of-Sight Laser Gated Viewing of Scattered Photons. *Optical Engineering* 53, 2 (2014). <https://doi.org/10/gfz5mr>
- Jaakko Lehtinen, Tero Karras, Samuli Laine, Miika Aittala, Frédo Durand, and Timo Aila. 2013. Gradient-Domain Metropolis Light Transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 32, 4 (July 2013), 95:1–95:12. <https://doi.org/10/gbdghd>
- He Li, Beibei Wang, Changhe Tu, Kun Xu, Nicolas Holzschuch, and Ling-Qi Yan. 2022. Unbiased Caustics Rendering Guided by Representative Specular Paths. In *ACM SIGGRAPH Asia Conference Papers*. Association for Computing Machinery, New York, NY, USA, 1–8. <https://doi.org/10/grqvds>
- Daqi Lin, Markus Kettunen, Benedikt Bitterli, Jacopo Pantaleoni, Cem Yuksel, and Chris Wyman. 2022. Generalized Resampled Importance Sampling: Foundations of ReSTIR. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 41, 4 (July 2022), 75:1–75:23. <https://doi.org/10/gqjn7b>
- Daqi Lin, Chris Wyman, and Cem Yuksel. 2021. Fast Volume Rendering with Spatiotemporal Reservoir Resampling. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 40, 6 (Dec. 2021). <https://doi.org/10/grrjd6>
- Jeffrey Liu, Daqi Lin, Markus Kettunen, Chris Wyman, and Ravi Ramamoorthi. 2025. Reservoir Splatting for Temporal Path Resampling and Motion Blur. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Papers*. 1–11.
- Yang Liu, Shaojie Jiao, and Wojciech Jarosz. 2022. Temporally Sliced Photon Primitives for Time-of-Flight Rendering. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 41, 4 (2022). <https://doi.org/10.1111/cgf.14584>
- Julio Marco. 2013. *Transient Light Transport in Participating Media*. Ph. D. Dissertation. Universidad de Zaragoza.
- Julio Marco, Ibón Guillén, Wojciech Jarosz, Diego Gutierrez, and Adrian Jarabo. 2019. Progressive Transient Photon Beams. *Computer Graphics Forum* 38, 1 (March 2019). <https://doi.org/10/gfvr9w>
- Matthew O'Toole, Felix Heide, Lei Xiao, Matthias B. Hullin, Wolfgang Heidrich, and Kiriakos N. Kutulakos. 2014. Temporal Frequency Probing for 5D Transient Analysis of Global Light Transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 33, 4 (2014), 87:1–87:11. <https://doi.org/10/gfz5m2>
- Yaobin Ouyang, Shiqiu Liu, Markus Kettunen, Matt Pharr, and Jacopo Pantaleoni. 2021. ReSTIR GI: Path Resampling for Real-Time Path Tracing. *Computer Graphics Forum* 40, 8 (2021), 17–29. <https://doi.org/10/gqwmxd>
- Adithya Pediredla, Akshat Dave, and Ashok Veeraraghavan. 2019a. SNLOS: Non-line-of-sight Scanning through Temporal Focusing. In *IEEE International Conference on Computational Photography (ICCP)*. 1–13. <https://doi.org/10/gspq34>
- Adithya Pediredla, Ashok Veeraraghavan, and Ioannis Gkioulekas. 2019b. Ellipsoidal Path Connections for Time-Gated Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 38, 4 (July 2019), 38:1–38:12. <https://doi.org/10/gf5jbm>
- Christoph Peters, Jonathan Klein, Matthias B. Hullin, and Reinhard Klein. 2015. Solving Trigonometric Moment Problems for Fast Transient Imaging. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 34, 6 (2015). <https://doi.org/10/f7wqfq>
- Diego Royo, Jorge García, Adolfo Muñoz, and Adrian Jarabo. 2022. Non-Line-of-Sight Transient Rendering. *Computers & Graphics* 107 (Oct. 2022), 84–92. <https://doi.org/10/gspq3r>
- Diego Royo, Jorge Garcia-Pueyo, Miguel Crespo, Óscar Pueyo-Ciudad, Guillermo Enguita, and Diego Bielsa. 2025. mitransient: Transient light transport in Mitsuba 3. arXiv:2510.25660 [cs.GR]. <https://arxiv.org/abs/2510.25660>
- Rohan Sawhney, Daqi Lin, Markus Kettunen, Benedikt Bitterli, Ravi Ramamoorthi, Chris Wyman, and Matt Pharr. 2022. Decorrelating ReSTIR Samplers via MCMC Mutations. <https://doi.org/10.48550/arXiv.2211.00166> arXiv:2211.00166 [cs]
- Siddharth Somasundaram, Akshat Dave, Connor Henley, Ashok Veeraraghavan, and Ramesh Raskar. 2023. Role of transients in two-bounce non-line-of-sight imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9192–9201.
- Justin F. Talbot. 2005. *Importance Resampling for Global Illumination*. Master's thesis. Brigham Young University. <https://scholarsarchive.byu.edu/etd/663>
- Justin F. Talbot, David Cline, and Parris Egbert. 2005. Importance Resampling for Global Illumination. In *Rendering Techniques (Proceedings of the Eurographics Symposium on Rendering)*. Eurographics Association, 139–146. <https://doi.org/10/gfzsm2>
- Chia-Yin Tsai, Kiriakos N. Kutulakos, Srinivasa G. Narasimhan, and Aswin C. Sankaranarayanan. 2017. The geometry of first-returning photons for non-line-of-sight imaging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7216–7224.
- Eric Veach. 1997. *Robust Monte Carlo Methods for Light Transport Simulation*. Ph. D. Dissertation. Stanford University.
- Andreas Velten, Thomas Willwacher, Otkrist Gupta, Ashok Veeraraghavan, Mouni G. Bavendi, and Ramesh Raskar. 2012. Recovering Three-Dimensional Shape around a Corner Using Ultrafast Time-of-Flight Imaging. *Nature Communications* 3 (2012).
- Andreas Velten, Di Wu, Adrian Jarabo, Belen Masia, Christopher Barsi, Chinmaya Joshi, Everett Lawson, Mouni G. Bavendi, Diego Gutierrez, and Ramesh Raskar. 2013. Femto-Photography: Capturing and Visualizing the Propagation of Light. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 32, 4, Article 44 (July 2013), 8 pages. <https://doi.org/10/gfpxkt>
- Di Wu, Andreas Velten, Matthew O'Toole, Belen Masia, Amit Agrawal, Qionghai Dai, and Ramesh Raskar. 2014. Decomposing Global Light Transport Using Time of Flight Imaging. *International Journal of Computer Vision (IJCV)* 107 (2014). <https://doi.org/10/f5v589>
- Lifan Wu, Guangyan Cai, Ravi Ramamoorthi, and Shuang Zhao. 2021. Differentiable Time-Gated Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 40, 6 (Dec. 2021). <https://doi.org/10/h2c9>
- Xiaofeng Xu, Lu Wang, and Beibei Wang. 2023. Efficient caustics rendering via spatial and temporal path reuse. In *Computer Graphics Forum*, Vol. 42. Wiley Online Library, e14975.
- Shinyoung Yi, Donggun Kim, Kiseok Choi, Adrian Jarabo, Diego Gutierrez, and Min H. Kim. 2021. Differentiable Transient Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 40, 6 (Dec. 2021). <https://doi.org/10/h2db>
- Aaron Young, Nevindu M. Batagoda, Harry Zhang, Akshat Dave, Adithya Pediredla, Dan Negrut, and Ramesh Raskar. 2025. Enhancing autonomous navigation by imaging hidden objects using single-photon lidar. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 4907–4914.
- Franco Zappa, Simone Tisa, Alberto Tosi, and Sergio Cova. 2007. Principles and features of single-photon avalanche diode arrays. *Sensors and Actuators A: Physical* 140, 1 (2007), 103–112.
- Tizian Zeltner, Iliyan Georgiev, and Wenzel Jakob. 2020. Specular Manifold Sampling for Rendering High-Frequency Caustics and Glints. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 39, 4 (July 2020). <https://doi.org/10/gg8xc8>
- Zheng Zeng, Markus Kettunen, Chris Wyman, Lifan Wu, Ravi Ramamoorthi, Ling-Qi Yan, and Daqi Lin. 2025. ReSTIR PG: Path Guiding with Spatiotemporally Resampled Paths. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*. 1–11.
- Song Zhang, Daqi Lin, Markus Kettunen, Cem Yuksel, and Chris Wyman. 2024. Area ReSTIR: Resampling for Real-Time Defocus and Antialiasing. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–13.