Doppler Time-of-Flight Rendering

JUHYEON KIM, Dartmouth College, USA
WOJCIECH JAROSZ, Dartmouth College, USA
IOANNIS GKIOULEKAS, Carnegie Mellon University, USA
ADITHYA PEDIREDLA, Dartmouth College, USA

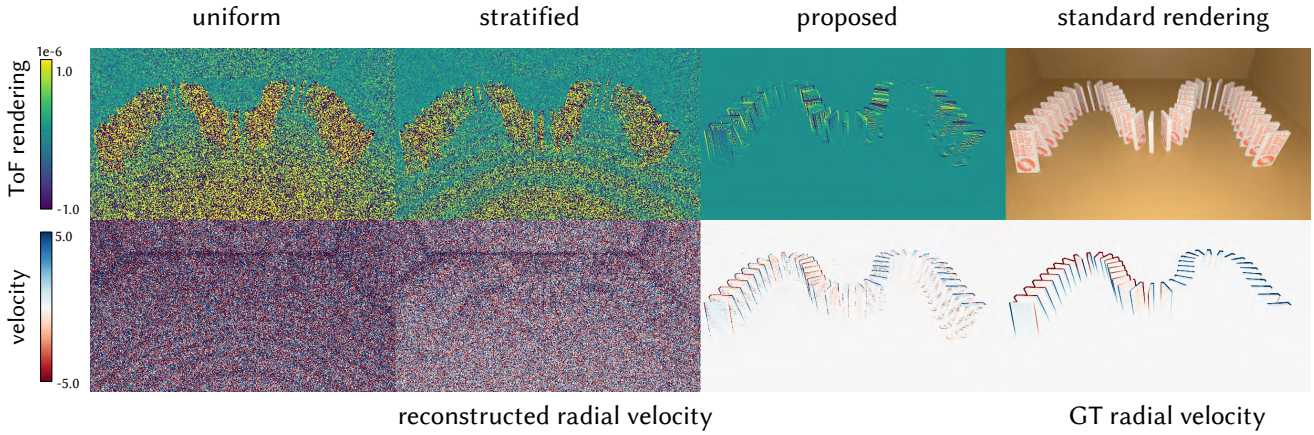

Fig. 1. We propose an unbiased and efficient rendering algorithm for Doppler time-of-flight cameras. Compared to naive sampling algorithms (left two columns), ours uses antithetic sampling with path correlation, and results in rendered images with orders of magnitude lower variance (top row). This makes it feasible to use rendering to evaluate radial velocity reconstruction algorithms (bottom row). (A teaser animation can be viewed in Adobe Acrobat Reader.)

We introduce Doppler time-of-flight (D-ToF) rendering, an extension of ToF rendering for dynamic scenes, with applications in simulating D-ToF cameras. D-ToF cameras use high-frequency modulation of illumination and exposure, and measure the Doppler frequency shift to compute the radial velocity of dynamic objects. The time-varying scene geometry and high-frequency modulation functions used in such cameras make it challenging to accurately and efficiently simulate their measurements with existing ToF rendering algorithms. We overcome these challenges in a twofold manner: To achieve accuracy, we derive path integral expressions for D-ToF measurements under global illumination and form unbiased Monte Carlo estimates of these integrals. To achieve efficiency, we develop a tailored time-path sampling technique that combines antithetic time sampling with correlated path sampling. We show experimentally that our sampling technique achieves up to two orders of magnitude lower variance compared to naive time-path sampling. We provide an open-source simulator that serves as a digital twin for D-ToF imaging systems, allowing imaging researchers, for the first time, to investigate the impact of modulation functions, material properties, and global illumination on D-ToF imaging performance.

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

Authors' addresses: Juhyeon Kim, Dartmouth College, USA, juhyeon.kim.gr@dartmouth.edu; Wojciech Jarosz, Dartmouth College, USA, wojciech.k.jarosz@dartmouth.edu; Ioannis Gkioulekas, Carnegie Mellon University, USA, igkioule@cs.cmu.edu; Adithya Pediredla, Dartmouth College, USA, adithya.k.pediredla@dartmouth.edu.

Additional Key Words and Phrases: time-of-flight imaging, Doppler effect, physically based rendering, computational imaging

ACM Reference Format:
Juhyeon Kim, Wojciech Jarosz, Ioannis Gkioulekas, and Adithya Pediredla. 2023. Doppler Time-of-Flight Rendering. *ACM Trans. Graph.* 42, 6, Article 271 (December 2023), 18 pages. https://doi.org/10.1145/3618335

1 INTRODUCTION

The last decade has witnessed a proliferation of time-of-flight (ToF) rendering algorithms, which simulate various ToF cameras in a physically accurate manner. These algorithms have facilitated several improvements in ToF-based imaging systems for applications such as depth sensing [Marco et al. 2017a; Po et al. 2022; Su et al. 2018], non-line-of-sight imaging [Iseringhausen and Hullin 2020; Pediredla et al. 2019a; Tsai et al. 2019], and imaging through scattering [Raghuram et al. 2019]. ToF rendering algorithms have also facilitated sensor design [Zhang et al. 2022] and large dataset generation for supervised learning [Chen et al. 2020; Gutierrez-Barragan et al. 2021].

Doppler time-of-flight (D-ToF) cameras are a class of ToF cameras that use the Doppler effect to estimate the radial velocity of moving objects [Heide et al. 2015; Hu et al. 2022]. In contrast to inter-frame methods [Whyte et al. 2015], D-ToF cameras can instantly evaluate radial velocity [Heide et al. 2015]. This capability makes them ideal for scenarios requiring high-speed operation, such as industrial

- Project page : https://juhyeonkim95.github.io/project-pages/dopplertof/

robotics, automobiles, and drones. Despite the practical importance of D-ToF cameras, efficient physically accurate rendering algorithms for them do not exist, hindering research and engineering efforts.

In particular, prior ToF rendering algorithms [Ament et al. 2014; Jarabo et al. 2014; Liu et al. 2022; Marco et al. 2019, 2017b; Pediredla et al. 2019b] assume static scenes, and therefore cannot handle the dynamic scenes that D-ToF cameras typically image. At first glance, it may appear that combining ToF and motion blur rendering algorithms could enable the simulation of D-ToF cameras. However, as we explain in Section 3, dynamic scenes break key assumptions underlying ToF rendering algorithms, making these algorithms inefficient or even incorrect. Moreover, due to the use of high-frequency modulation in D-ToF cameras, the naive time sampling of motion blur rendering algorithms results in significant noise, overwhelming the signal from subtle net-intensity differences between temporally adjacent light paths that D-ToF cameras measure.

In this paper, we introduce the first efficient, physically based rendering framework for D-ToF cameras. We start by deriving a ToF path integral for dynamic scenes (Sec. 3) that generalizes both ToF and motion blur rendering. We then simulate D-ToF imaging systems by estimating this integral, which we can do in an unbiased and efficient manner using Monte Carlo integration with importance sampling. However, importance sampling the D-ToF integrand is non-trivial: the integrand is the product of the path throughput and a sinusoidal function that depends on both time and each path's time of flight. Therefore, naive time-path sampling will result in large variance due to the random cancellation of contributions from the positive and negative lobes of the integrand as shown in Fig. 1. We overcome these challenges by introducing an efficient two-step time-path sampling strategy: time-domain antithetic sampling (Sec. 4), combined with correlated light path sampling (Sec. 5).

We implement both CPU and GPU versions of our method using Mitsuba 0.6 [Jakob 2013] and Mitsuba 3 [Jakob et al. 2022], and demonstrate its effectiveness on scenes with varying geometries and material parameters (Sec. 6). We show that our method can efficiently and accurately simulate different types of D-ToF cameras, including both homodyne and heterodyne at different frequency ranges. We replicate experimental results from prior D-ToF imaging work [Heide et al. 2015; Hu et al. 2022], and use our simulations to analyze failure cases (Sec. 7). We expect that our open-source implementation will facilitate research toward improving D-ToF cameras and velocity estimation algorithms.

2 RELATED WORK

Time-of-Flight Imaging. Time-of-flight (ToF) cameras measure time-resolved flux, and include transient, time-gated, and continuous-wave time-of-flight (CW-ToF) cameras. Transient cameras [O'Toole et al. 2017; Velten et al. 2011] use narrow pulsed laser sources and bin photons based on their time of travel. Time-gated cameras [Pediredla et al. 2019a; Walia et al. 2022] use pulsed lasers and fast shutters to capture photons that travel a fixed time range. Continuous-wave time-of-flight cameras use amplitude-modulated illumination and exposure, measuring transients in the Fourier domain [Lin et al. 2016; O'Toole et al. 2014; Peters et al. 2015]. All these ToF cameras have uses in applications such as depth sensing [Foix et al. 2011; Gokturk et al. 2004; Gupta et al. 2019; Lange and Seitz 2001; Po et al. 2022], robot navigation [Prusak et al. 2008; Yuan et al. 2009], non-line-of-sight (NLOS) imaging [Buttafava et al. 2015; Kadambi et al. 2016; Liu et al. 2019; O'Toole et al. 2018; Pediredla et al. 2017], and imaging through scattering media [Naik et al. 2014; Satat et al. 2016]. Some of these applications have been either enabled or enhanced due to advances in ToF rendering.

Time-of-Flight Rendering. ToF rendering is the physically accurate simulation of measurements of ToF cameras. ToF rendering algorithms have mostly focused on synthesizing a sequence of images that show the evolution of light (transient rendering) or a single image for a specific time gate (time-gated rendering). Both cases require handling the near-delta temporal manifold, which is a challenging problem. Inspired by their success in steady-state rendering [Jarosz et al. 2011, 2008; Jensen 2001; Jensen and Christensen 1998], several approaches have proposed using photon density estimation [Ament et al. 2014; Jarabo 2012; Jarabo et al. 2014] or photon beam methods [Marco et al. 2019, 2017b] for ToF rendering. Pediredla et al. [2019b] adapted ideas from ToF participating media rendering [Jarabo et al. 2014] and proposed ellipsoidal path connections to sample contributing paths for time-gated rendering. Liu et al. [2022] extended this approach to general photon primitives.

Other works focus on accelerated ToF rendering for specific applications. Tsai et al. [2019] and Iseringhausen and Hullin [2020] proposed simplified three-bounce rendering models for fast rendering in NLOS imaging. Pan et al. [2019] proposed a GPU-accelerated rasterization method using a transient version of instant radiosity. Recently, Yi et al. [2021], Wu et al. [2021], and Plack et al. [2023] developed differentiable ToF rendering algorithms to solve inverse imaging problems with analysis-by-synthesis.

In contrast to these prior works that largely focused on sampling the delta time-manifold, we focus on the unexplored problem of efficiently integrating a time-varying path space coupled with high-frequency illumination and sensor modulation signals.

Doppler Time-of-Flight Imaging. Doppler time-of-flight (D-ToF) imaging is a CW-ToF technique that estimates the radial velocity of a moving object [Heide et al. 2015; Shrestha et al. 2016]. If we illuminate the scene with a high-frequency temporal signal, the moving objects cause a frequency shift to the observed signal due to the Doppler effect. Heide et al. [2015] achieve D-ToF imaging using two ToF imaging modes, homodyne and heterodyne, which differ in sensor modulation frequency. Homodyne mode uses sensor modulation with the same frequency as the illumination modulation (10 MHz to 1000 MHz), whereas heterodyne mode uses a precisely shifted frequency—an integer multiple of the inverse exposure duration. A heterodyne image is proportional to the Doppler frequency shift, which in turn is proportional to the radial velocity. A homodyne image acts as a normalizing factor. Thus, the ratio of heterodyne and homodyne images provides the radial velocity. As a heterodyne mode image measures subtle frequency changes, it has low-intensity values, and thus a low signal-to-noise ratio (SNR). To overcome this problem, Hu et al. [2022] proposed heterodyne mode imaging with arbitrary sensor frequencies that maximize the SNR.

Our algorithms can simulate the cameras proposed by Heide et al. [2015] and Hu et al. [2022], as well as their generalized variants

that use arbitrary modulation waveforms. Whereas these papers use simple analytical models to analyze the D-ToF cameras they propose, we focus on how to efficiently and physically accurately simulate such cameras with rendering algorithms that account for both time-varying path throughput and global illumination effects.

Motion Blur. D-ToF rendering is related to motion blur rendering techniques [Navarro et al. 2011], which also handle dynamic scenes that change within exposure. These techniques work by distributing samples along both pixel (or path) and time spaces. Example sampling techniques for this problem include uniform [Cook et al. 1984], stratified [Mitchell 1996], adaptive [Whitted 1980], and multidimensional adaptive [Hachisuka et al. 2008], and frequency-aware [Egan et al. 2009]. However, D-ToF involves high-frequency modulation functions with negative path contributions, which lead to extreme variance using existing techniques. We devise tailored time-path space sampling techniques to tackle this problem.

Antithetic Sampling and Path Correlation. Antithetic sampling is a variance reduction technique that uses two correlated samples whose covariance is negative [Hammersley and Morton 1956]. Antithetic sampling significantly reduces variance if the integrand has regions that have negatively correlated parts.

Antithetic sampling has found uses in differential and gradient-domain rendering [Bangaru et al. 2020; Kettunen et al. 2015; Manzi et al. 2016; Zeltner et al. 2021; Zhang et al. 2021], leading to the development of *shift mapping* techniques for sampling highly correlated antithetic pairs of paths. Path correlation can be achieved in the primary sample space—the same random numbers generate two correlated paths [Hua et al. 2019; Manzi et al. 2016; Zeltner et al. 2021]—or in the path space—the primal path generates the correlated path by deterministic shifting [Kettunen et al. 2015; Zhang et al. 2021]. Prior work [Öztireli 2016; Singh et al. 2019; Subr et al. 2014] has additionally employed antithetic sampling in standard forward rendering for variance reduction.

Due to the use of illumination and sensor modulation, D-ToF rendering has a periodic integrand that lends itself to antithetic sampling. We show that antithetic sampling with appropriate path correlation reduces variance for a variety of modulation waveforms, including ones that do not have perfectly matching antithetic pairs.

3 DOPPLER TIME-OF-FLIGHT PATH INTEGRAL

We start by deriving a path integral expression for D-ToF rendering, and investigating its simplified form to devise an efficient sampling method. We summarize key notation in Tab. 1.

3.1 ToF Path Integral for Dynamic Scenes

The *ToF path integral* [Jarabo et al. 2014; Pediredla et al. 2019b],

$$m(T) = \int_0^T s(t) \int_{\mathcal{P}} f(\bar{\mathbf{x}}) g\left(t - \|\bar{\mathbf{x}}\|\right) \mathrm{d}\mu(\bar{\mathbf{x}}) \, \mathrm{d}t, \tag{1}$$

models the measurements $m(T)$ captured by a ToF imaging system for a scene that remains static during an exposure $[0, T]$. Here, $g(t)$ and $s(t)$ are the illumination and sensor modulation functions, respectively, and $\bar{\mathbf{x}} := \mathbf{x}_0\mathbf{x}_1 \ldots \mathbf{x}_K$ is a path consisting of $K+1$ vertices, with $\mathbf{x}_0$ on the sensor and $\mathbf{x}_K$ on the light source. The path space $\mathcal{P}$ is the set of all light paths of all lengths $K > 0$, and $\mathrm{d}\mu(\bar{\mathbf{x}})$ is the corresponding Lebesgue measure. The path throughput $f(\bar{\mathbf{x}})$ accounts for visibility, geometric attenuation, and reflectance at all path vertices and edges. We use $\|\bar{\mathbf{x}}\| := \sum_{j=0}^{K-1} \left\|\mathbf{x}_j - \mathbf{x}_{j+1}\right\|$ for the time of flight of a path, where $\left\|\mathbf{x}_j - \mathbf{x}_{j+1}\right\|$ is the time it takes light to travel from $\mathbf{x}_{j+1}$ to $\mathbf{x}_j$ considering the medium refractive index.

The assumption of a static scene allowed previous techniques to change the integration order in the ToF path integral:

$$m(T) = \int_{\mathcal{P}} f(\bar{\mathbf{x}}) W_T(\|\bar{\mathbf{x}}\|) \, \mathrm{d}\mu(\bar{\mathbf{x}}), \tag{2}$$

where $W_T(\|\bar{\mathbf{x}}\|) := \int_0^T s(t) g\left(t - \|\bar{\mathbf{x}}\|\right) \mathrm{d}t$ is the *pathlength importance* [Pediredla et al. 2019b]. This form of the ToF path integral enabled the development of importance sampling strategies for path length [Pediredla et al. 2019b], and even techniques to analytically integrate the pathlength importance over time [Liu et al. 2022].

Table 1. Definition of variables used throughout the paper.

Notation	Description
$\bar{\mathbf{x}}_t$	Path with scene geometry at time t.
$\mathcal{P}(t)$	Path space with scene geometry at time t.
$\mathcal{M}(t, \cdot)$	Path mapping function at time t.
$\bar{\mathbf{x}}(t)$	Evolution of path $\bar{\mathbf{x}}(0) \in \mathcal{P}(0)$ to time t according to $\mathcal{M}$.
$\|\bar{\mathbf{x}}\|$	Time of flight of $\bar{\mathbf{x}}$.
$\phi(\bar{\mathbf{x}})$	Phase offset of path due to $\|\bar{\mathbf{x}}\|$.
$s(t), g(t)$	Sensor and illumination modulation function.
ω_s, ω_g	Sensor and illumination modulation function's frequency.
ψ	Programmable phase offset at sensor.
$[0, T]$	Sensor exposure time.
ω_d	Heterodyne frequency, $\omega_s - \omega_g$.
ω_r	Normalized heterodyne frequency, $\omega_\mathrm{d}/\omega_0$ ($\omega_0 : 2\pi/T$).
$\Delta\omega$	Observed Doppler frequency shift.
N_p	Number of sampled path evolutions.
N_t	Number of time samples along the evolution of a given $\bar{\mathbf{x}}(0)$.

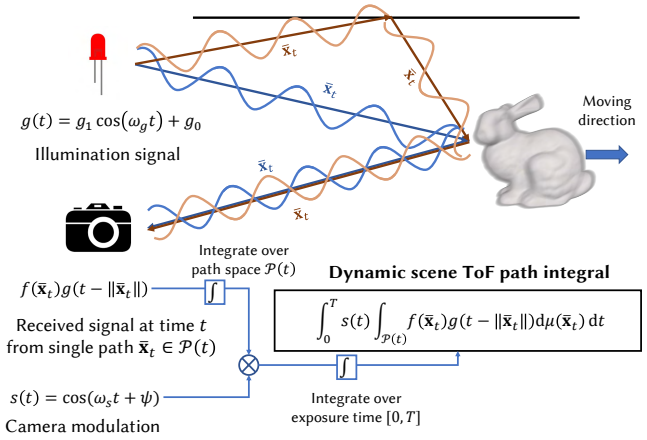

Fig. 2. Overview of ToF imaging and rendering of a dynamic scene. We show two paths $\bar{\mathbf{x}}_t$ that arrive at the same image pixel. In D-ToF imaging, $s(t)$ and $g(t)$ are high-frequency sinusoidal functions. The sinusoidal waves represent the illumination intensity at each point on the two paths.

However, in general, the scene geometry is not static within exposure, and the path space $\mathcal{P}(t)$ varies over time t as surfaces in the scene move. Then, the ToF path integral becomes

$$m(T) = \int_0^T s(t) \int_{\mathcal{P}(t)} f(\bar{\mathbf{x}}_t) g\left(t - \|\bar{\mathbf{x}}_t\|\right) \mathrm{d}\mu(\bar{\mathbf{x}}_t)\, \mathrm{d}t. \tag{3}$$

Here, $\bar{\mathbf{x}}_t \in \mathcal{P}(t)$ is a light path, similar to $\bar{\mathbf{x}}$, but using scene geometry at time t. We call Eq. (3) the *dynamic ToF path integral*. D-ToF imaging is a special case of this path integral where the modulation functions are high-frequency functions designed for radial velocity estimation [Heide et al. 2015]. Figure 2 visualizes the terms of Eq. (3) in a D-ToF imaging system. Equation (3) is, in fact, more general and can reproduce not only D-ToF imaging, but also other dynamic scene-related phenomena such as motion blur. However, whereas lighting may change slightly during exposure in a typical motion blur setting, D-ToF imaging includes extremely high-frequency modulation functions, necessitating new rendering approaches.

We ignore the subtle difference between the actual light path that arrives at the sensor at time t—and thus interact with the scene at times prior to t—and the path that is built using scene geometry at time t. We show in the supplement that this approximation introduces a negligible bias that is proportional to the square of the ratio of the radial velocity v to the speed of light c.

3.2 Path Evolutions over Time

The dynamic ToF path integral (3) does not assume any path mapping function over time, which means $\bar{\mathbf{x}}_t$ are all independent. However, imposing such a cross-time path correspondence makes the double integral more tractable. Therefore, we will introduce a path mapping function $\mathcal{M}(t, \cdot) : \mathcal{P}(0) \to \mathcal{P}(t)$ that is bijective for all $t \in [0, T]$ and maps a path $\bar{\mathbf{x}}_0$ to a path at arbitrary time t. We write a *path evolution* at time t under this mapping as $\bar{\mathbf{x}}(t) \coloneqq \mathcal{M}(t, \bar{\mathbf{x}}_0)$, which implies $\bar{\mathbf{x}}(t) = \mathcal{M}(t, \bar{\mathbf{x}}(0))$. Replacing $\bar{\mathbf{x}}_t$ with $\bar{\mathbf{x}}(t)$ in Eq. (3):

$$\begin{aligned}
&\int_0^T s(t) \int_{\mathcal{P}(t)} f(\bar{\mathbf{x}}(t)) g\left(t - \|\bar{\mathbf{x}}(t)\|\right) \mathrm{d}\mu(\bar{\mathbf{x}}(t))\, \mathrm{d}t \\
&= \int_0^T s(t) \int_{\mathcal{P}(0)} f(\bar{\mathbf{x}}(t)) J_{\mathcal{M}}(\bar{\mathbf{x}}(0)) g\left(t - \|\bar{\mathbf{x}}(t)\|\right) \mathrm{d}\mu(\bar{\mathbf{x}}(0))\, \mathrm{d}t \\
&= \int_{\mathcal{P}(0)} \int_0^T \widehat{f}(\bar{\mathbf{x}}(t)) s(t) g\left(t - \|\bar{\mathbf{x}}(t)\|\right) \mathrm{d}t\, \mathrm{d}\mu(\bar{\mathbf{x}}(0)),
\end{aligned} \tag{4}$$

where $J_{\mathcal{M}}(\bar{\mathbf{x}}(0)) \coloneqq \mathrm{d}\mu(\bar{\mathbf{x}}(t))/\mathrm{d}\mu(\bar{\mathbf{x}}(0))$ is the determinant of the Jacobian of the mapping, and $\widehat{f}(\bar{\mathbf{x}}(t)) \coloneqq f(\bar{\mathbf{x}}(t)) J_{\mathcal{M}}(\bar{\mathbf{x}}(0))$. This formulation is analogous to the material-form path space parameterization by Zhang et al. [2020]. To justify the assumption that there exists such a bijective mapping $\mathcal{M}$, we provide one intuitive example: We consider a path $\bar{\mathbf{x}}(0)$ with vertices attached to different points of the scene geometry at time $t = 0$. As the scene geometry evolves over time, these points will move to new locations, evolving at each time t into a new path $\bar{\mathbf{x}}(t)$ in bijective correspondence with the original path $\bar{\mathbf{x}}(0)$. The mapping $\mathcal{M}$ induced by this correspondence is not the only possible one, and we discuss other mappings $\mathcal{M}$ we use for our rendering algorithm in Sec. 5.

3.3 Sampling Strategy for D-ToF Rendering

To start our investigation of efficient sampling techniques, we first examine the shape of the integrand in Eq. (4) for the D-ToF case. Similar to the setup Heide et al. [2015] describe, we will consider sensor and illumination modulation functions that are high-frequency sinusoidal waves with frequencies ω_s and ω_g, respectively:

$$s(t) \coloneqq \cos\left(\omega_s t + \psi\right);\ \ g(t) \coloneqq g_1 \cos\left(\omega_g t\right) + g_0. \tag{5}$$

Here, ψ is a programmable phase offset at the sensor, and g_1, g_0 are constant values. We define the difference between the two frequencies as the heterodyne frequency $\omega_\mathrm{d} \coloneqq \omega_s - \omega_g$. Heide et al. [2015] used two imaging modes for radial velocity evaluation: *heterodyne* mode with $\omega_\mathrm{d} = 2\pi/T$, and *homodyne* mode with $\omega_\mathrm{d} = 0$. In subsequent work, Hu et al. [2022] proposed using a heterodyne mode where ω_d can take any value within the range of $[0, 2\pi/T]$, to improve signal-to-noise ratio. We aim to reproduce this more general setting and thus assume that $\omega_\mathrm{d} \in [0, 2\pi/T]$.

We define the path phase offset $\phi(\bar{\mathbf{x}}) \coloneqq -\omega_g \|\bar{\mathbf{x}}\|$, and use it to express the integrand in Eq. (4) as:

$$\widehat{f}(\bar{\mathbf{x}}(t)) \cos\left(\omega_s t + \psi\right) \left(g_1 \cos\left(\omega_g t + \phi(\bar{\mathbf{x}}(t))\right) + g_0\right). \tag{6}$$

Expanding the cosine terms in Eq. (6), we get the following terms:

$$\begin{aligned}
&\frac{g_1}{2} \cos\left((\omega_s + \omega_g)t + \psi + \phi(\bar{\mathbf{x}}(t))\right) \\
&+ \frac{g_1}{2} \cos\left((\omega_s - \omega_g)t + \psi - \phi(\bar{\mathbf{x}}(t))\right) + g_0 \cos\left(\omega_s t + \psi\right).
\end{aligned} \tag{7}$$

As $\widehat{f}(\bar{\mathbf{x}}(t))$ and $\phi(\bar{\mathbf{x}}(t))$ vary slowly relative to ω_s, and $T \gg 1/\omega_s$, the high-frequency terms ($\omega_s + \omega_g$ and ω_s) approximately sum to zero when we integrate over $[0, T]$. Empirically, we found that ignoring the high-frequency terms reduces the variance by several orders of magnitude. This low-pass filtering operation introduces bias of order $O(\omega_\mathrm{d} v/c)$ to the Doppler frequency shift, itself of order $O(\omega_g v/c)$. As ω_d is typically around a few kHz and ω_g ranges from 10 MHz to 1000 MHz, in practice the bias is less than 0.1% and thus negligible. Therefore, we will consider only the low-frequency ($\omega_s - \omega_g = \omega_\mathrm{d}$) term of $s(t)g\left(t - \|\bar{\mathbf{x}}(t)\|\right)$, which we will refer to as the *modulation term* for simplicity,

$$\frac{g_1}{2} \cos\left(\omega_\mathrm{d} t - \phi(\bar{\mathbf{x}}(t)) + \psi\right). \tag{8}$$

We can do similar low-pass filtering for arbitrary periodic signals $s(t)$, $g(t)$ using their Fourier series which results in correlation of two signals with frequency of ω_d, as we show in the supplement.

Substituting Eq. (8) in Eq. (4) gives a tractable D-ToF path integral:

$$\int_{\mathcal{P}(0)} \int_0^T \widehat{f}(\bar{\mathbf{x}}(t)) \frac{g_1}{2} \cos\left(\omega_\mathrm{d} t - \phi(\bar{\mathbf{x}}(t)) + \psi\right) \mathrm{d}t\, \mathrm{d}\mu(\bar{\mathbf{x}}(0)). \tag{9}$$

To understand the behavior of Eq. (9), in Fig. 3 we plot the integrand at three pixels assuming single-bounce paths $\bar{\mathbf{x}}(t)$. The illumination is a point light source collocated with the sensor. In heterodyne mode, the integrand is a sinusoid of exactly one period for static objects, making the integral zero in this case. For dynamic objects, ϕ is a function of time and contributes a Doppler frequency shift, making the integrand not a single-period sinusoid. This results in a non-zero integral and this non-zero value is important for computing the Doppler frequency shift, and thus the radial velocity. In homodyne mode, the integrand is close to a linear function as

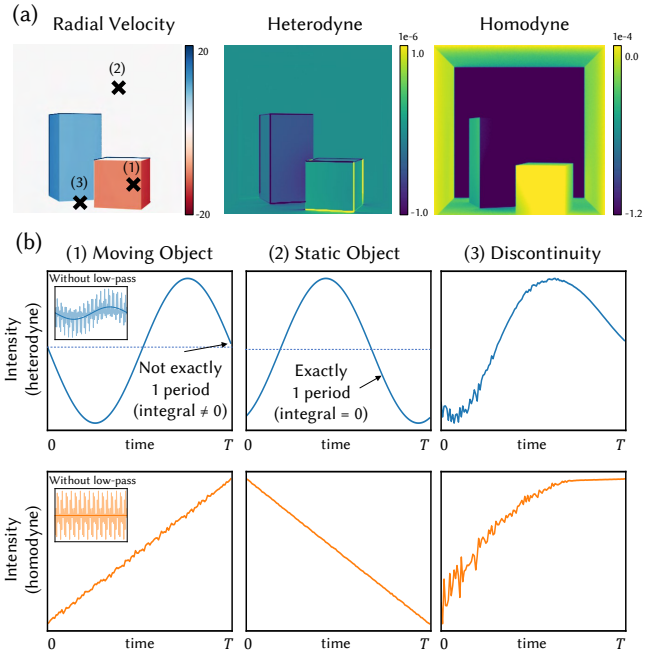

Fig. 3. Integrand of Eq. (9) for a single-bounce path over time. Integrating the signal over the time-path space generates the homodyne and heterodyne images (first row). The unfiltered version is also plotted at the corner.

$\omega_d = 0$. For paths near edges where scene discontinuities occur, we observe drastic changes in the integrand (third column in Fig. 3). In this case, $\|\bar{\mathbf{x}}(t)\|$ and $\widehat{f}(\bar{\mathbf{x}}(t))$ change significantly during exposure, and the integrand is neither a sinusoidal nor a linear function.

Inspired by this pilot experiment, we estimate the double integral of Eq. (9) by sampling its two domains, time and path, in two steps that we visualize in Fig. 4:

1. Assuming $\widehat{f}(\bar{\mathbf{x}}(t))$ and $\|\bar{\mathbf{x}}(t)\|$ do not change significantly with time t for some, still unknown, $\bar{\mathbf{x}}(0)$ (i.e, assuming the behavior of the third column in Fig. 3-(b) is rare), we find two *antithetic* time samples for which Eq. (8) has approximately the same deviation from ground truth but opposite sign.
2. We use these time samples to create *correlated* path samples, equal to the evolutions $\bar{\mathbf{x}}(t)$ of an underlying $\bar{\mathbf{x}}(0)$ at the sampled times t, such that $\widehat{f}(\bar{\mathbf{x}}(t))$ and $\|\bar{\mathbf{x}}(t)\|$ do not vary significantly with t, as we required in the first step.

We use these samples to form a Monte Carlo estimator of Eq. (9):

$$\langle m(T)\rangle := \frac{1}{N}\sum_{i=0}^{N_p}\sum_{j=0}^{N_t}\frac{\hat{f}(\bar{\mathbf{x}}^i(t^j))\frac{g_1}{2}\cos\left(\omega_d t^j - \phi(\bar{\mathbf{x}}_i(t^j)) + \psi\right)}{p(t^j)p(\bar{\mathbf{x}}^i(0))}, \quad (10)$$

where N_p is the number of paths $\bar{\mathbf{x}}(0)$ used for evolutions, N_t is the number of antithetic time samples for the evolution of each $\bar{\mathbf{x}}(0)$ (which we set to 2 by default), $N := N_p \cdot N_t$ is the number of total path samples, and p is the sampling probability density function (pdf). We detail antithetic time sampling in Sec. 4 and correlated path sampling in Sec. 5. It is challenging to reverse the sampling order—find antithetic samples in the path domain and align them in the time domain—as this would require finding an antithetic path with a specific length, which is expensive [Pediredla 2019].

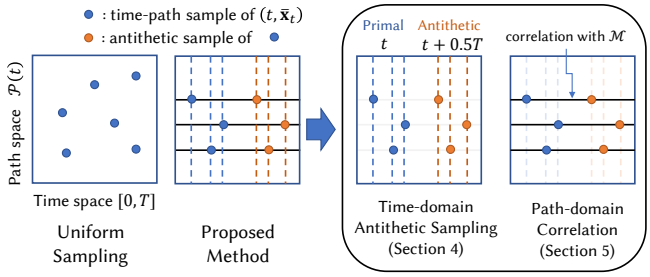

Fig. 4. Overview of our sampling strategy. We visualize path evolutions $\bar{\mathbf{x}}(t)$ for different paths $\bar{\mathbf{x}}(0)$ as horizontal lines in the time-path space. We first sample time using antithetic sampling, then create correlated path samples as evolutions of different $\bar{\mathbf{x}}(0)$ to the sampled antithetic time pairs.

4 TIME DOMAIN ANTITHETIC SAMPLING

Assuming a given path evolution where $\widehat{f}(\bar{\mathbf{x}}(t))$ and $\|\bar{\mathbf{x}}(t)\|$ do not change significantly with t, we aim to find an efficient time-sampling strategy using N_t samples for the modulation term time integral,

$$\int_0^T \cos\left(\omega_d t - \phi(\bar{\mathbf{x}}(t)) + \psi\right) \mathrm{d}t, \quad (11)$$

where $\omega_d \in [0, 2\pi/T]$. There is a trade-off between the number of independent path evolutions N_p and the number of time samples N_t, given a fixed budget N of total path samples. Increasing N_t decreases the variance of the modulation term; but it also reduces N_p which in turn increases the variance of the path throughput term. Empirically, we found that $N_t = 2$ results in lower overall variance, and thus fix it throughout the paper unless we state otherwise.

4.1 Antithetic Sampling for Modulation Term

To further narrow down our problem, we simplify Eq. (11). We assume that $\|\bar{\mathbf{x}}(t)\|$ is near constant over a path evolution, so we can approximate $\phi(\bar{\mathbf{x}}(t)) \approx \phi(\bar{\mathbf{x}}(0))$. We use this approximation *only* to derive time sampling techniques and not for the actual evaluation. We can represent $\phi(\bar{\mathbf{x}}(0)) + \psi$ as a random variable θ of unknown, scene-dependent distribution. Then, Eq. (11) simplifies to:

$$\int_0^T \cos\left(\omega_d t + \theta\right) \mathrm{d}t. \quad (12)$$

We know ω_d before sampling, but not θ. Our goal is to efficiently evaluate Eq. (12) for $\omega_d \in [0, 2\pi/T]$ and unknown θ.

For inspiration on how to sample time, we consider the heterodyne and homodyne cases (Fig. 5-(a,b)). In these cases, we can easily find a zero-variance estimator using two-sample *antithetic sampling*. If $\omega_d = 2\pi/T$ (perfect heterodyne), given *primal* sample t, we can select the *antithetic* sample as $t_a = \mathrm{mod}(t + 0.5T, T)$. Then $\cos(2\pi/T t + \theta)$ is exactly a cosine function with period $[0, T]$, and antithetic sampling gives exactly zero variance regardless of t and θ. We call this strategy *shifted antithetic sampling*.

If $\omega_d \approx 0$ (homodyne), the integrand becomes close to a linear function. Then, selecting a symmetric antithetic sample as $T - t$

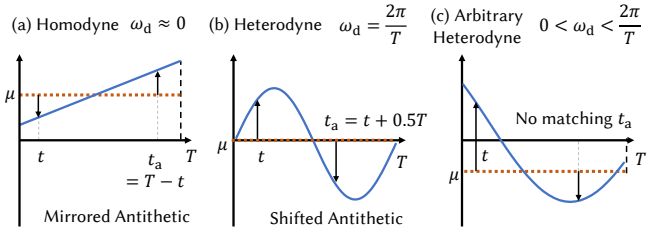

Fig. 5. Antithetic sampling for (a) homodyne, (b) heterodyne, and (c) arbitrary heterodyne mode. The ground truth integration value is μ. In (a) and (b), we can find an antithetic pair with mirroring and shifting, respectively, that makes the primal and antithetic samples sum to μ. But in (c), we cannot find such perfectly matching antithetic sample.

will make the sum of primal and antithetic values have expected value $\cos(\omega_d t + \theta)$ for $t \in [0, T]$. Thus we obtain a zero-variance estimator. We call this strategy *mirrored antithetic sampling*.

However, our goal is to handle arbitrary $\omega_d \in [0, 2\pi/T]$ as Hu et al. [2022] proposed, so we cannot expect to find a perfectly matching antithetic sample in general (Fig. 5-(c)). Instead, we aim to find an antithetic sample that minimizes the variance over t. We use shifted or mirrored antithetic sampling by setting the antithetic sample as $t + t_s$ or $-t + t_s$ for some constant t_s. (We omit the mod operation over T for simpler notation.) For perfect heterodyne or homodyne operation, t_s has an optimal value at $t_s = 0.5T$ and $t_s = 0$, respectively. In the following subsections, we will discuss the relationship of our antithetic estimator to *auto-correlation* and *auto-convolution* in signal processing, and show that $t_s = 0.5T$ and $t_s = 0$ is also optimal for arbitrary heterodyne frequencies.

4.2 Shifted Antithetic Sampling and Auto-correlation

We start with shifted antithetic sampling ($t_a = t + t_s$), which is optimal for the heterodyne case. Using a primal sample t drawn uniformly and an antithetic sample $t+t_s$, the variance of an antithetic estimator for an arbitrary integrand $x(t)$ is

$$\mathrm{Var}(t_s) = \int_0^T \left(\frac{x(t) + x\left(\mathrm{mod}(t + t_s, T)\right)}{2} - \mu_x \right)^2 \mathrm{d}t, \tag{13}$$

where μ_x is the mean of $x(t)$ in $[0, T]$. We see that minimizing the variance is equivalent to minimizing the *auto-correlation* :

$$R(t_s) = \int_0^T x(t) x\left(\mathrm{mod}(t + t_s, T)\right) \mathrm{d}t. \tag{14}$$

As auto-correlation is symmetric around $0.5T$, we can represent $R(t_s) = F(t_s) + F(T - t_s)$ for some F. If $x(t) = \cos(\omega_d t + \theta)$, we can analytically calculate $F(t_s)$ as

$$F(t_s) = \frac{1}{2\omega_d} \cos(\omega_d T + 2\theta) \sin(\omega_d t_s) + \frac{t_s}{2} \cos\left(\omega_d (T - t_s)\right). \tag{15}$$

To better understand this expression, we express R as a function of ω_d, θ, t_s, rather than just t_s. Then, we prove in the supplement that $R(\omega_d, \theta, t_s)$ has a global minimum at $t_s = 0.5T$ regardless of the values of ω_d, θ, if $\omega_d \in [0, 2\pi/T]$. Interestingly, optimality at $t_s = 0.5T$ also holds for other waveforms, such as triangular or trapezoidal, as Fig. 6 shows. The fact that the optimal shift is independent of θ provides a significant benefit when we consider multi-bounce paths, which contain many subpaths with different θ values.

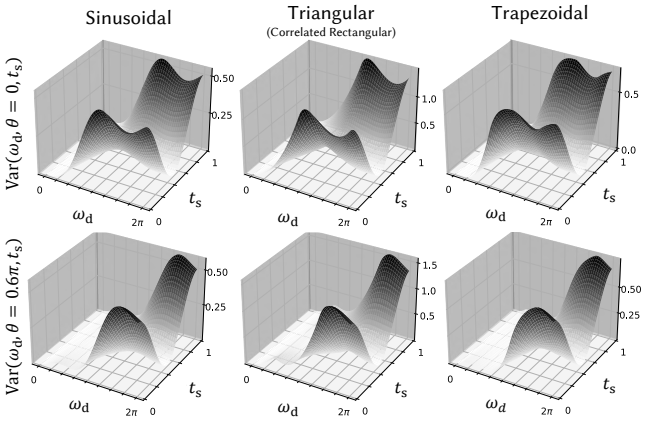

Fig. 6. $\mathrm{Var}(\omega_d, \theta, t_s)$ of shifted antithetic sampling for various waveforms $x(t)$. They all have optimal values at $t_s = 0.5T$ regardless of ω_d and θ. Without loss of generality, we set $T = 1$ and hence, $\omega_d \in [0, 2\pi]$. We only show results for two cases, but the tendency was similar for other θs.

4.3 Mirrored Antithetic Sampling and Auto-convolution

For mirrored antithetic sampling $t_a = -t + t_s$, which is optimal for the homodyne case, we can repeat our analysis analogously to the previous subsection. The only difference is that the auto-correlation becomes *auto-convolution* as the sign is inverted:

$$C(t_s) = \int_0^T x(t) x\left(\mathrm{mod}(-t + t_s, T)\right) \mathrm{d}t. \tag{16}$$

Unfortunately, in this case, we cannot find a globally minimal t_s that is independent of ω_d and θ for $x(t) = \cos(\omega_d t + \theta)$. Instead, we try to find an optimal value for the expectation over θ. As we do not know in advance the distribution of θ, which depends on path time-of-flight, we assume it to be uniform over $[0, 2\pi]$. Then, for the sinusoidal wave, we can analytically calculate the expectation:

$$\mathbb{E}_\theta[C(\omega_d, \theta, t_s)] = \frac{\sin(\omega_d t_s) + \sin\left(\omega_d (T - t_s)\right)}{2\omega_d}, \tag{17}$$

and this has a minimum at $t_s = 0, T$ regardless of $\omega_d \in [0, 2\pi/T]$.

4.4 Comparison with Uniform and Stratified Sampling

In Fig. 7, we compare the variance of antithetic sampling with other sampling methods—uniform sampling and stratified sampling, with $N_t = 2$. For both antithetic sampling techniques, we use our previously derived optimal shift t_s. In Fig. 7-(a) we plot $\mathrm{Var}(\omega_d, \theta)$ for different ω_d, θ, and in Fig. 7-(b) we plot the expectation of $\mathrm{Var}(\omega_d, \theta)$ over θ and its ratio to the uniform sampling case. Overall, antithetic sampling has better performance than both uniform and stratified sampling for different values of ω_d. Mirrored antithetic sampling is most effective for $\omega_d \in [0, \pi/T]$, and shifted antithetic for $\omega_d \in [\pi/T, 2\pi/T]$. We also observe that the advantages of different antithetic sampling strategies persist across different waveforms, including triangular and trapezoidal (Fig. 7-(b)).

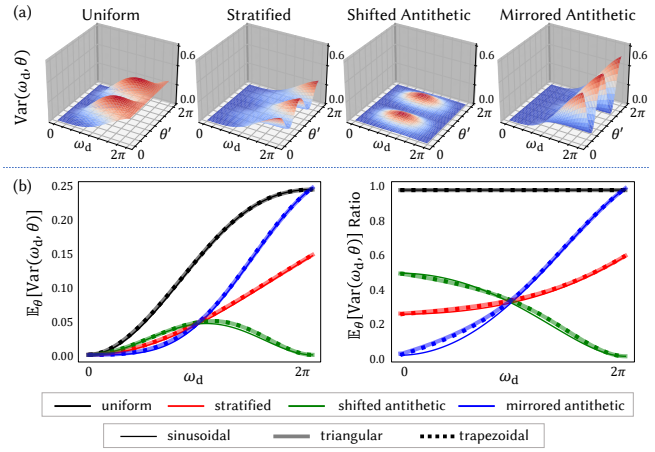

Fig. 7. Comparison of time sampling methods with $N_t = 2$. (a) We plot $\mathrm{Var}(\omega_\mathrm{d}, \theta)$ for a sinusoidal waveform using different sampling methods. Like Hu et al. [2022], we use $\theta' = \theta + 0.5\omega_\mathrm{d}T$ instead of θ for the x-axis to make the plot not skewed. We set $T = 1$. (b) For different sampling methods and waveforms, we plot (left) $\mathbb{E}_\theta[\mathrm{Var}(\omega_\mathrm{d}, \theta)]$, and (right) its ratio to the variance from uniform sampling. Mirrored and shifted antithetic sampling are most effective for $\omega_\mathrm{d} \in [0, \pi/T]$ and $\omega_\mathrm{d} \in [\pi/T, 2\pi/T]$, respectively.

4.5 Further Stratification in the Time Domain

Until now, we have only considered the simple 1D case where $N = N_t = 2$, as in Fig. 8-(a). For rendering, however, we will need to consider sampling over the path domain while additionally increasing N (for example, $N = 8$ in Fig. 8-(b) and (c)). The direct extension of the 1D case with $N_t = 2$ would correspond to Fig. 8-(b). Unfortunately, this extension does not stratify the collection of all N samples across the time domain, as the stratification is still fixed by $N_t = 2$. We could improve stratification by increasing $N_t > 2$, but doing so reduces path diversity for a fixed sample budget $N = N_t \cdot N_p$.

Instead, we propose to combine antithetic sampling with *further stratification* over the time domain (Fig. 8-(c)). For $N_t = 2$, our approach distributes the primal samples within the left half of the time domain, and places the antithetic samples at either a shifted (7a) or mirrored (8a) location in the right half of the domain, providing full stratification with N strata (in the actual implementation, we may swap the primal and antithetic samples). In order to evaluate the impact of antithetic sampling, we also consider in Fig. 8-(6a) a fully stratified extension of Fig. 8-(6), but which does not exploit any antithetic properties. Instead, this approach enforces N strata in time and correlates the paths of random strata with each other (stratum i is correlated with a random stratum j on the other half).

In Fig. 9, we compare the effect of using further stratification in the Cornell-box scene. (We describe the experiment details in Sec. 6.) When we do not use further stratification, the relative performance of different techniques resembles the 1D case with $N_t = 2$ in Fig. 7. Applying further stratification keeps relative performance similar, but reduces overall variance for all time-domain sampling techniques. Thus, to ensure optimal performance, we use in our experiments further stratification in the time domain for all sampling methods except uniform sampling.

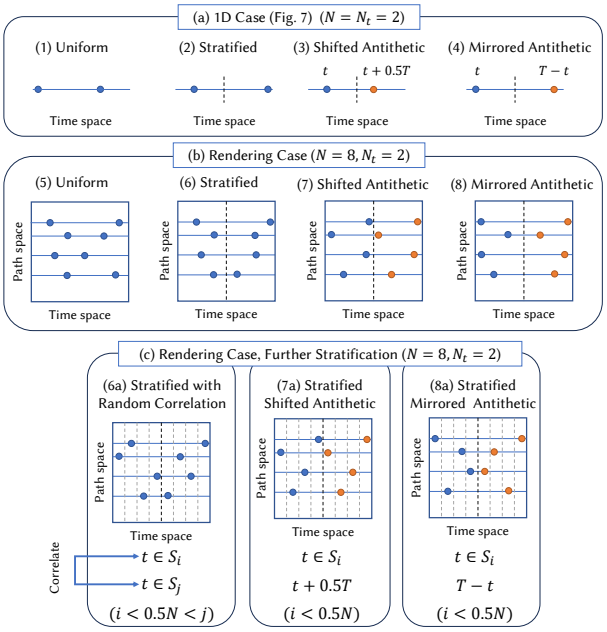

Fig. 8. Comparison of time-domain stratification techniques. We visualize only cases where the primal samples are in $[0, 0.5T]$. Horizontal lines are path evolutions, as in Fig. 4. The 1D case in Fig. 7 considers only the time domain and corresponds to (a) with $N = N_t = 2$. The rendering case jointly considers both time and path domains. The direct expansion of the 1D to the rendering case is (b), with $N_t = 2$. In (c) we further improve (b) with $N_t = 2$, by stratifying the time domain. S_i is a stratum of $[i/NT, (i+1)/NT]$.

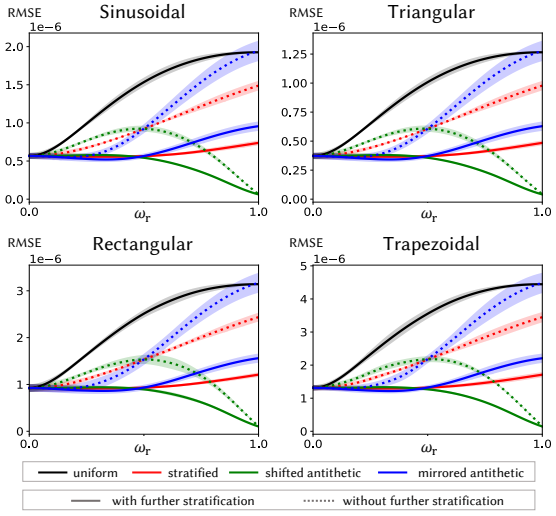

Fig. 9. RMSE versus ω_r (normalized ω_d) for various sampling configurations and waveforms for the Cornell-Box scene. The relative performance when not using further stratification (dotted line) is similar to the 1D case in Fig. 7. Enforcing further stratification (solid line) improves it in all cases.

4.6 Analytic Approximation

So far, we have discussed Monte Carlo integration techniques to estimate Eq. (9). We can alternatively derive an analytic approximation to this equation, which results in a fast but biased estimate. The first-order Taylor series approximations of $\widehat{f}(\bar{\mathbf{x}}(t))$ and $\|\bar{\mathbf{x}}(t)\|$ are:

$$\widehat{f}(\bar{\mathbf{x}}(0)) + t\frac{\partial}{\partial t}\widehat{f}(\bar{\mathbf{x}}(t)), \|\bar{\mathbf{x}}(0)\| + t\frac{\partial}{\partial t}\|\bar{\mathbf{x}}(t)\|. \tag{18}$$

We evaluate the derivatives with the finite-difference method, using values at $t = 0, T$. With this approximation, we can integrate Eq. (9) analytically. This analytic approximation is similar to the method of Heide et al. [2015], except that we use first-order approximations for both $\|\bar{\mathbf{x}}(t)\|$ and $\widehat{f}(\bar{\mathbf{x}}(t))$, whereas Heide et al. use zeroth-order approximation for $\widehat{f}(\bar{\mathbf{x}}(t))$. We found that our analytic approximation, while still biased, significantly improves rendering performance compared to Heide et al.'s approximation (Fig. 19).

5 PATH CORRELATION BY TEMPORAL SHIFT MAPPING

We have discussed an efficient antithetic time domain sampling assuming a path mapping $\mathcal{M}$ such that $\widehat{f}(\bar{\mathbf{x}}(t))$ and $\|\bar{\mathbf{x}}(t)\|$ remain approximately constant over time t. In this section, we will focus on finding such a mapping $\mathcal{M}$. As a $\mathcal{M}$ that results in near-constant $\widehat{f}(\bar{\mathbf{x}}(t))$ also results in near-constant $\|\bar{\mathbf{x}}(t)\|$, we will focus on only $\widehat{f}(\bar{\mathbf{x}}(t))$. We will consider mappings $\mathcal{M}$ that we can represent as a composition of bijective path vertex mapping functions $\mathcal{T}_i(t,\cdot) : \mathbb{R}^3 \to \mathbb{R}^3$ for each vertex $\mathbf{x}_i(0)$ until path depth K. Therefore:

$$\mathcal{M}(t, \bar{\mathbf{x}}(0)) \coloneqq \mathcal{T}_0(t, \mathbf{x}_0(0))\mathcal{T}_1(t, \mathbf{x}_1(0)) \dots \mathcal{T}_K(t, \mathbf{x}_K(0)). \tag{19}$$

We assume that pixel coordinate of the camera ray $\mathcal{T}_0(t, \mathbf{x}_0(0)) \to \mathcal{T}_1(t, \mathbf{x}_1(0))$ is unaltered by $\mathcal{M}$. Such a constraint makes sensor importance remain the same, decreasing variance significantly. Prior work has studied finding good mappings $\mathcal{M}$ in the context of *shift mappings* between adjacent pixels [Hua et al. 2019; Kettunen et al. 2015; Lehtinen et al. 2013; Manzi et al. 2016]. Our $\mathcal{M}$ is a time-domain shift, not an image-domain shift as most of the works, so we call it *temporal shift mapping* following Manzi et al. [2016]. However, our $\mathcal{M}$ can be defined on arbitrary shift values, not a unit frame shift as Manzi et al. [2016]. We will adopt some simple shift mapping strategies, as we discuss in the following subsections. Previously, we defined the reference path space at $t = 0$, but we now equivalently use the primal time sample as a reference for convenience.

5.1 Temporal Random Replay

In temporal random replay $\mathcal{T}_\text{s}$, we use the same sequence of random numbers to generate both primal and antithetic paths [Hua et al. 2019; Manzi et al. 2016]. Temporal random replay is a mapping on *primary sample space* (PSS) [Kelemen et al. 2002]. Therefore, to be accurate, $\mathcal{T}_i$ should be defined on random numbers required to sample x_i. However, exploiting the theoretical bijectivity between random numbers and paths [Bitterli et al. 2018], we will keep using the path vertex representation with a bit of abuse. We found that temporal random replay works well when path vertex locality is preserved—the transformed points still intersect the same object, with roughly the same normal, texture coordinates, etc.—but fails otherwise—near silhouette edges or high-frequency textures.

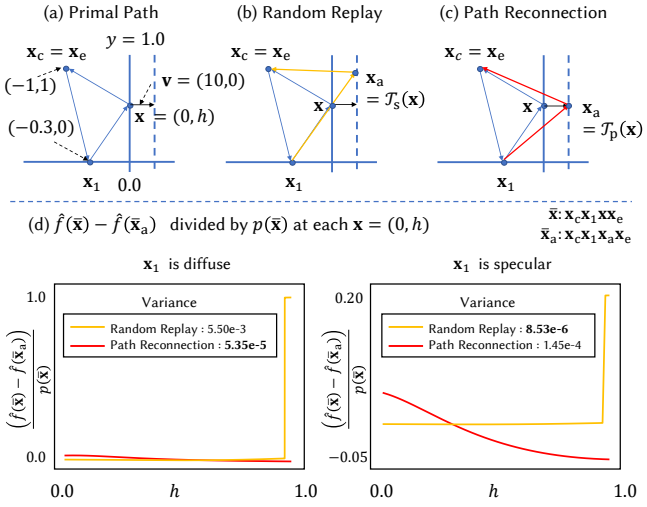

Fig. 10. We exploit two temporal shift mapping techniques to generate an antithetic path for a given (a) primal path. Random replay (b) uses the same random numbers as the primal path to generate the antithetic path. Path reconnection (c) keeps vertices attached to the moving geometry to preserve locality. We show (d) the difference of $\hat{f}$ between the primal and the antithetic path, divided by the sampling pdf, for vertical wall points. Random replay has better correlation everywhere except near the edge, which results in significant variance for the diffuse case, but not the specular case.

5.2 Temporal Path Reconnection

In temporal path reconnection $\mathcal{T}_\text{p}$, we reconnect antithetic path with transformed primal path vertex along with the geometry it resides on. Temporal path reconnection is conceptually equivalent to *path reconnection* in Kettunen et al. [2015], but transforms the target vertex according to time. Temporal path reconnection has the advantage of preserving locality at the transformed point. However, it does not consider BSDF importance sampling, thus it is less advantageous for making $\widehat{f}(\bar{\mathbf{x}}(t))$ constant over t. This is especially problematic for specular materials where a small change in ray direction causes significant BSDF change.

For shorter notation, we will omit the word *temporal* from both temporal random replay and temporal path reconnection from now.

5.3 Analysis of Random Replay and Path Reconnection

To better understand the two types of temporal shift mappings, we consider the simple 2D scene in Fig. 10. We denote a primal path $\bar{\mathbf{x}}$. and its antithetic path $\bar{\mathbf{x}}_\text{a}$. The floor is fixed and the wall that runs vertically between $y \in [0,1]$ moves in the positive x direction with speed of 10. The point light source ($\mathbf{x}_\text{e}$) and sensor ($\mathbf{x}_\text{c}$) are collocated. We use BSDF sampling to sample the primal path at $t = 0$, which has the form $\bar{\mathbf{x}} = \mathbf{x}_\text{c}\mathbf{x}_1\mathbf{x}\mathbf{x}_\text{e}$ with vertices $\mathbf{x}_\text{c} = (-1,1), \mathbf{x}_1 = (-0.3,0), \mathbf{x} = (0,h)$, and thus only depends on $h \in [0,1]$. To map $\bar{\mathbf{x}}$ to its antithetic path $\bar{\mathbf{x}}_\text{a}$ at $t_\text{a} = 0.5T$, where $T = 0.001$, our only degree of freedom is changing the intersection point $\mathbf{x}_\text{a}$ on the moved wall at t_a, which we can do using either $\mathcal{T}_\text{s}$ or $\mathcal{T}_\text{p}$. To compare the two mappings, in Fig. 10 we plot the difference of $\hat{f}(\bar{\mathbf{x}}) - \hat{f}(\bar{\mathbf{x}}_\text{a})$ divided by the primal path pdf $p(\bar{\mathbf{x}})$, which is an unbiased estimate for

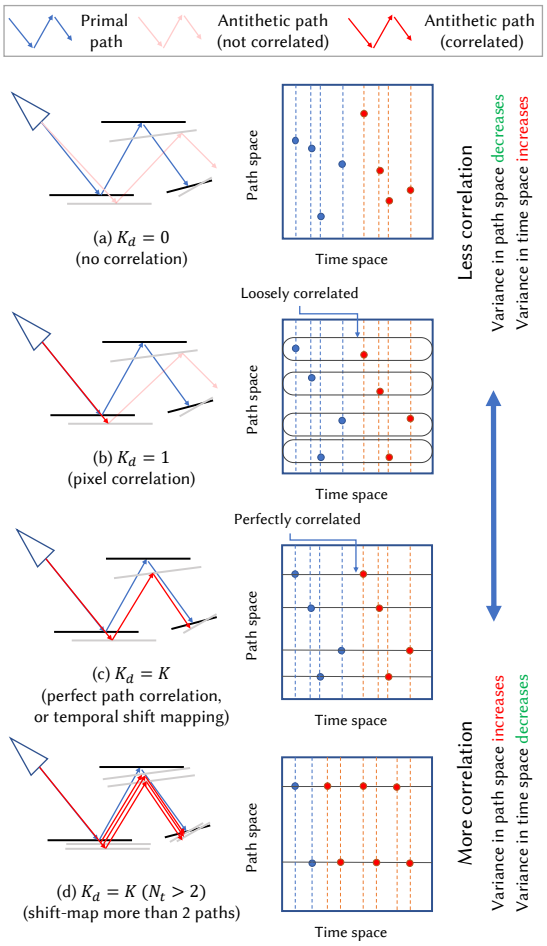

Fig. 11. Effects of varying path correlation strength. By increasing the shift mapping depth or mapping more paths, and thus making paths more correlated, we can decrease variance of the time sub-integral, but increase variance of the path sub-integral. The optimal shift mapping depth depends on the scene configuration and heterodyne frequency.

Eq. (9) with perfectly matching modulation term. We observe that random replay gives near-zero difference for most of the h values. However, for primal points near the edge of the wall, random replay gives an invalid antithetic point, and $\hat{f}$ drastically changes. This results in a large spike near the edge if $\mathbf{x}_1$ is diffuse. On the other hand, path reconnection does not suffer from this discontinuity problem. For specular materials, random replay still suffers from the discontinuity, but is better than path reconnection. This is because path reconnection causes a dramatic change in BSDF, which makes $\widehat{f}$ differ significantly between primal and antithetic paths.

Primal and Antithetic Multiple Importance Sampling. As any path could be sampled as either primal or antithetic, we use multiple importance sampling (MIS) [Veach and Guibas 1995] between primal and antithetic samples. MIS is known to reduce variance in path reconnection [Kettunen et al. 2015].

Ensuring Unbiasedness. MIS between primal and antithetic samples also helps ensure unbiasedness—it effectively rejects a shift mapping if bijectivity fails because the antithetic path has zero throughput (e.g., due to a missing ray or path blocking) and thus could not have been sampled as the primal path. In this case, the integration falls back from Eq. (4) to Eq. (3), but is still unbiased. One thing we need to be careful about is to ensure the primal sample covers whole time domain so that there exists no unsampled area even though shift mapping fails. This is why we randomly swapped the stratum of primal and antithetic samples in Sec. 4.5.

Implementation. Random replay is preferable to path reconnection in terms of implementation complexity. We can implement random replay implicitly by simply repeating the path tracing process with the same sampler and random seed, but replacing the primal with the antithetic time sample. On the other hand, to implement path reconnection, we need to either trace primal and antithetic paths at the same time, or store the primal path and update its vertices to form the antithetic path.[1] Both approaches have considerable implementation overhead.

5.4 Adaptive Temporal Shift Mapping

As neither random replay nor path reconnection is universally better, we follow Kettunen et al. [2015] and combine the two mapping strategies adaptively based on the vertex material: If current, next vertex of primal path $(\mathbf{x}_i, \mathbf{x}_{i+1})$, and current vertex of antithetic path $(\mathbf{x}_{\mathrm{a},i})$ are identified as diffuse material, we use path reconnection, otherwise, we use random replay. We found this adaptive approach to be effective for most scenes. An alternative approach would be to use MIS between the two shift mapping strategies. Unfortunately, we found this approach to be prohibitively expensive, as it requires storing 2^K paths created with all possible per-vertex combinations of random replay versus path reconnection. Furthermore, we show in the supplement that such an MIS approach is not guaranteed to perform well under antithetic sampling.

5.5 Depth-limited Temporal Shift Mapping

In practice, instead of mapping the entire path until vertex K, we can limit mapping only to the first $K_d < K$ vertices,

$$\mathcal{M}(t, \overline{\mathbf{x}}(0)) \coloneqq \mathcal{T}_0(t, \mathbf{x}_0(0))\mathcal{T}_1(t, \mathbf{x}_1(0)) \dots \mathcal{T}_{K_d}(t, \mathbf{x}_{K_d}(0)). \quad (20)$$

After $K_d{}^{\text{th}}$ vertex, we continue the path using independent path tracing.[2] Note that this is no longer deterministic shift mapping, but correlation with some stochasticity. This *depth-limited* approach makes the primal and antithetic paths less correlated, and thus makes antithetic time sampling less effective. However, it is advantageous in terms of increasing diversity in path space. Figure 11 shows this trade-off between path correlation and path diversity. Empirically, we found that perfect path correlation, (or temporal shift mapping), works best at near-heterodyne modes, whereas limited correlation works best at near-homodyne modes.

[1] Random replay can be also implemented this way, but the implicit method is preferable.
[2] Our choice to stop shift mapping at vertex K_d means that our rendering algorithm uses a hybrid between the path integrals of Eq. (3) and Eq. (4), with partial Jacobian.

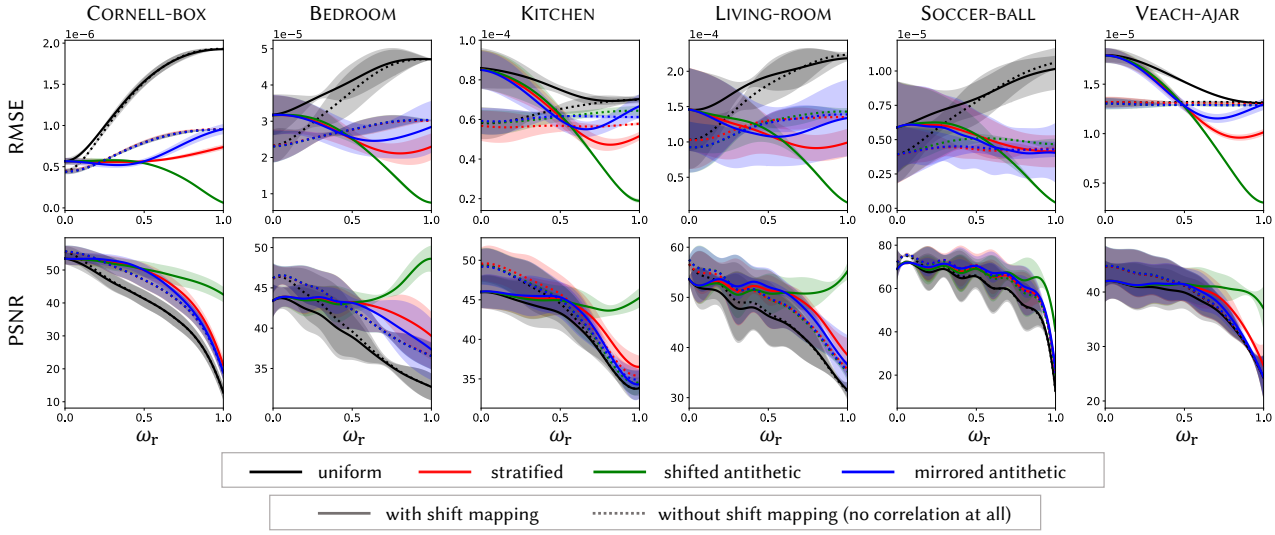

Fig. 12. RMSE and PSNR of different sampling methods for $\omega_r \in [0, 1]$ (x-axis for all of the plots). We plotted mean and variance for uniformly spaced values of ψ. Shifted antithetic sampling with shift mapping had the best performance for $\omega_r \in [0.5, 1.0]$. For $\omega_r \in [0.0, 0.5]$, among techniques with shift mapping, mirrored antithetic sampling performed the best; but shift mapping did not always improve performance.

6 EXPERIMENTS

In this section, we evaluate the time-path sampling techniques we proposed in Secs. 4 and 5 for various scene geometries, modulation functions, and modulation frequencies.

Implementation and Experimental Settings. We implement CPU and GPU versions of our algorithm using Mitsuba 0.6 [Jakob 2013] and Mitsuba 3 [Jakob et al. 2022]. We use six scenes for evaluation, with a collocated point light source and camera. Across all scenes, we set $\omega_g = 30$ MHz and $T = 1.5$ ms, same as Heide et al. [2015]. We use 1024 samples per pixel (spp) for all experiments. We also set the maximum bounce depth to 4, or 8 if the scene contains refractive objects. Unless we state otherwise, we use $N_t = 2$ and random replay with full depth by default. To simplify notation, we use a normalized version $\omega_r \in [0, 1]$ instead of $\omega_d \in [0, 2\pi/T]$ in this section.

6.1 Effectiveness of Antithetic Sampling

To demonstrate the effectiveness of our proposed antithetic sampling methods with path correlation, we compare them against standard sampling techniques with and without shift mapping for various scenes under different heterodyne frequencies ω_r and sensor phase offsets ψ (11×11 configurations of $\omega_r \in [0, 1]$, $\psi \in [0, 2\pi]$ with uniformly-spaced intervals). In each method except uniform sampling, we use further stratification as we described in Sec. 4.5.

Figure 12 shows averages and standard deviations of RMSE and PSNR computed across ψ values. For $\omega_r \in [0.5, 1.0]$, shifted antithetic sampling with shift mapping clearly shows the best performance, especially at perfect heterodyne mode ($\omega_r = 1$), where it is around two orders of magnitude better in terms of squared error (variance) compared to the worst case of uniform sampling.

On the other hand, for $\omega_r \in [0.0, 0.5]$, there is no clear winner. If we consider only sampling methods with shift mapping (solid line),

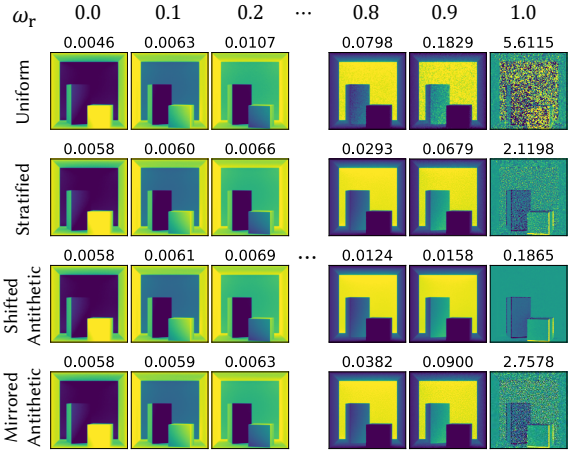

Fig. 13. Qualitative results in CORNELL-BOX for different values of ω_r at $\psi = 0$. Above each image we denote relative RMSE. We use shift mapping for all methods except uniform sampling. Performance differences are very noticeable at $\omega_r = 1.0$, but become essentially indistinguishable for $\omega_r < 0.8$.

their relative performance is similar to the 1D case in Fig. 7, where mirrored antithetic sampling works best for $\omega_r \in [0.0, 0.5]$, shifted antithetic sampling works best for $\omega_r \in [0.5, 1.0]$, and stratified sampling falls somewhere in between. However, such perfect path correlation turned out to be not always helpful for $\omega_r \in [0.0, 0.5]$.

This difference in performance is due to the trade-off we explained in Fig. 11 between variance in path space and time space. For the homodyne case ($\omega_r = 0$), the modulation term is near constant, so the variance in path space is the main bottleneck and we should dedicate more samples for path diversity, which makes shift mapping harmful.

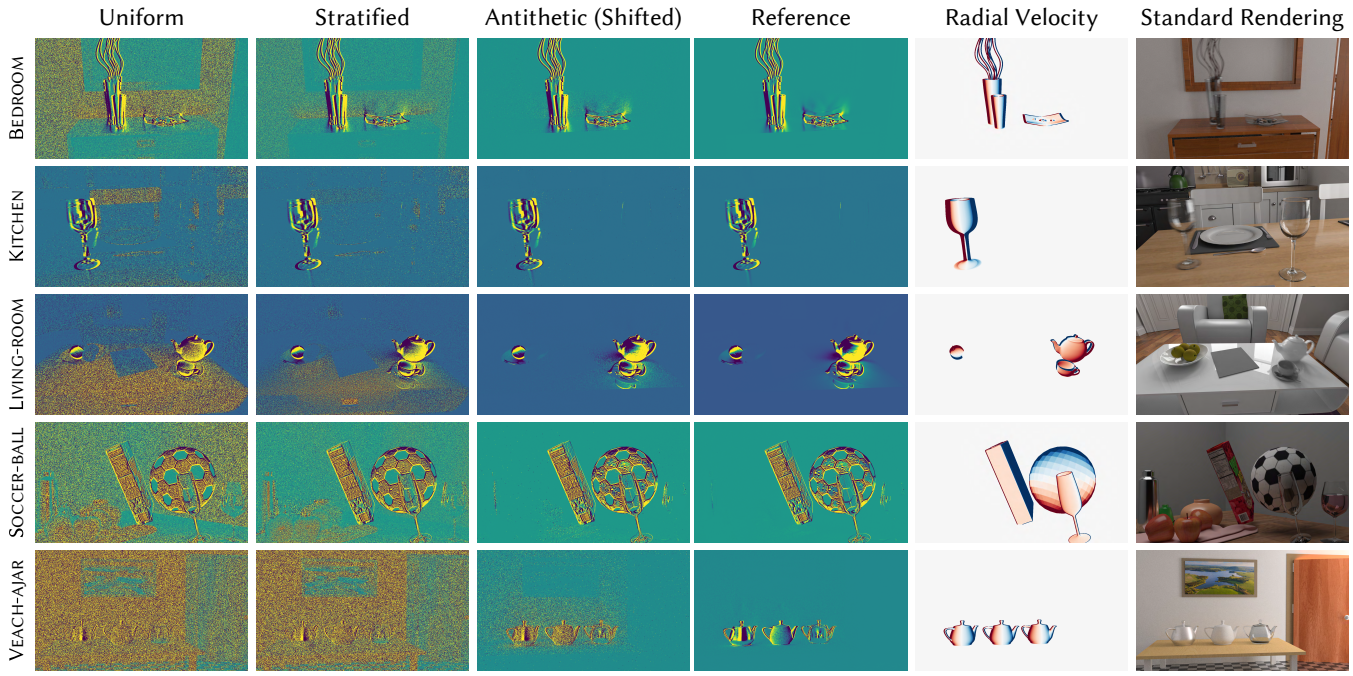

Fig. 14. Qualitative results using different time sampling methods for perfect heterodyne mode ($\omega_r = 1.0$) with $\psi = 0$. We used shift mapping in all results, as it performs the best at $\omega_r = 1.0$ (Fig. 12). We omit mirrored antithetic sampling for easier visualization. For each scene, we show D-ToF renderings, standard intensity renderings, and the ground truth radial velocity maps computed from depth differences at $t = 0, T$. Our proposed shifted antithetic sampling method has significantly lower variance than the uniform or stratified sampling methods.

As ω_r increases, the variance in time space due to the modulation term increases and eventually outweighs the variance in path space. Then, we should favor better time sampling with shift mapping at the cost of reduced path diversity. The exact value of ω_r where this change occurs is scene-dependent. For example, for a simple scene like the Cornell-box, path-space variance is relatively low, so improving time sampling using mirrored antithetic with shift mapping starts to become helpful from $\omega_r \approx 0.2$. This value is 0.3 for the Living-room, while does not appear for other scenes. We have more discussion on this correlation-diversity trade-off in Sec. 6.2.

Fortunately, when we consider PSNR, performance differences between methods are relatively small for $\omega_r \in [0.0, 0.5]$ (Fig. 13). However, for $\omega_r \in [0.5, 1.0]$, the differences become more pronounced, especially at $\omega_r = 1.0$. This implies that as ω_r becomes larger, it becomes more important to use carefully designed sampling techniques, in which case the proposed method can be particularly helpful. The qualitative evaluation for perfect heterodyne mode in Fig. 14 shows clear visual differences between sampling methods.

Using Other Waveforms. In Fig. 9, we show that our method performs well with other widely used non-sinusoidal modulation waveforms (rectangular, triangular and trapezoidal). Overall, relative performances remain consistent with the sinusoidal case.

Verification of Optimal Antithetic Shift. To verify our claim for the optimal shift t_s for antithetic sampling (Secs. 4.2 and 4.3), in Fig. 15 we show experiments using varying shift t_s in $[0, T]$. The first two columns show results for specific values of ψ, and the last column shows averages across all values of ψ. Shifted antithetic sampling performs the best results when $t_s = 0.5T$ for all combinations of ω_r and ψ. For mirrored antithetic sampling, the optimal shift value varies depending on ψ. However, when we consider the averaged results across all values of ψ, optimal performance occurs at $t_s = 0$ and $t_s = T$. These observations are consistent with our theory.

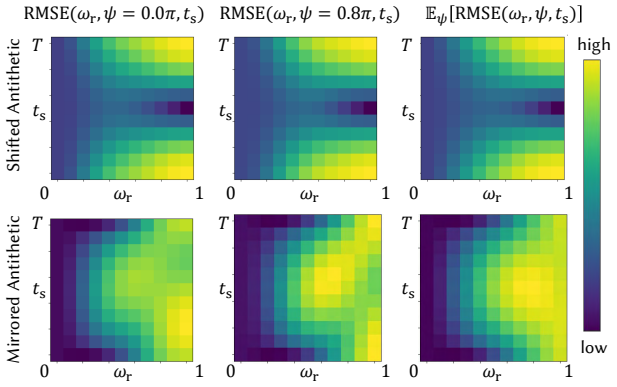

Fig. 15. RMSE by using different antithetic shifts (t_s) for Cornell-Box. For shifted antithetic, the optimal value occurs at $0.5T$ independent of ω_r, ψ. For mirrored antithetic, the optimal value depends on ψ. However, the optimal expected RMSE over uniform ψ occurs at $t_s = 0$, independent of ω_r.

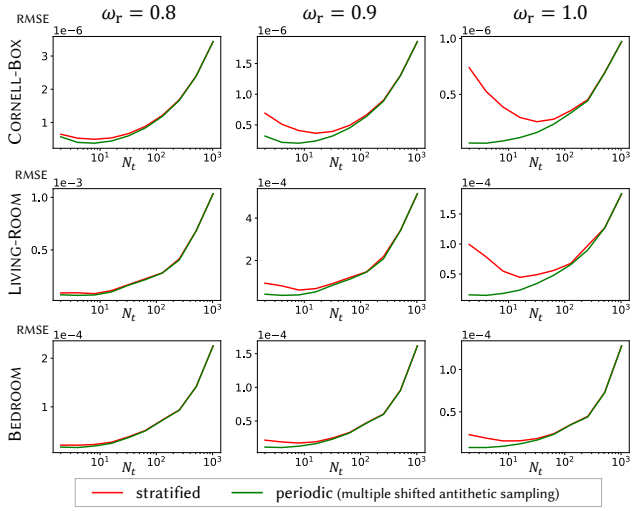

Fig. 16. RSME versus number of correlated paths N_t for the Cornell-Box, Living-Room, and Bedroom scenes. Even though the minimum RMSE for our method (green) sometimes occurs at $N_t > 2$, using $N_t = 2$ is close to the optimum and consistently outperforms stratified sampling (red).

6.2 Effect of Path Correlation Strength

We investigate the effect of varying degrees of correlation between sampled paths, as illustrated in Fig. 11. First, we examine the impact of increasing correlation by using more than two shift-mapped paths ($N_t > 2$) (Fig. 11-(d)). For stratified sampling, we increase N_t from 2 to 1024 while keeping the number of time strata fixed at 1024. For antithetic sampling, we only consider the shifted method, as extending the mirrored method to $N_t > 2$ is not straightforward. There are different ways to implement antithetic sampling with multiple samples. We use a periodic approach analogous to uniform jittered sampling [Pauly et al. 2000; Ramamoorthi et al. 2012]: instead of a single antithetic shift with $0.5T$, we use $[T/N_t, 2T/N_t, \ldots, (N_t-1)T/N_t]$ where $N_t \in [2, 1024]$, and correlate these paths with shift mapping.

From Fig. 16, we observe that increasing N_t above 2 generally degrades performance, but can provide some improvement at high ω_r values for stratified sampling. This improvement is because the variance of the modulation term dominates performance, thus dedicating more samples to the time domain is helpful. Too many correlated paths, however, result in higher variance due to limited path diversity. Modulation variance increases with larger ω_r, so stratified sampling achieves its minimum RMSE for $N_t > 2$ as $\omega_r \to 1$. Periodic sampling shows a similar trend, but it reaches its lowest RMSE at a much lower N_t (usually $N_t = 2$ or $N_t = 4$). Furthermore, performance at that point is better than stratified sampling. The performance improvement is largest for perfect heterodyne mode, where periodic sampling with just two samples already gives near zero-variance estimation, thus making it better to allocate the remaining samples towards increasing path diversity. Within the above general trends, there are some notable scene-dependent differences. As scene complexity increases (Cornell-Box < Living-Room < Bedroom), the penalty from reduced path diversity increases, and using more correlated paths becomes more harmful. In summary,

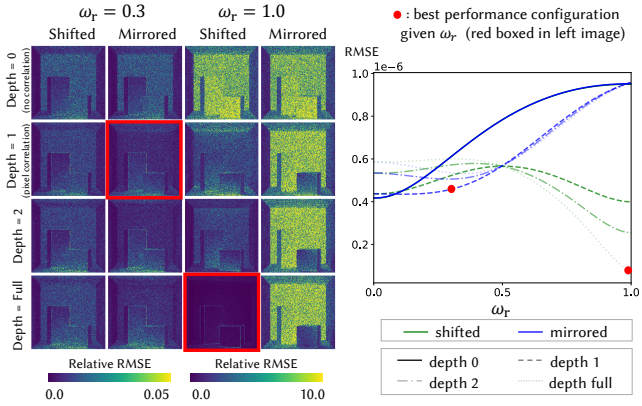

Fig. 17. The effect of using different maximum shift mapping depths. Large correlation depths work best for higher ω_r, and worst for lower ω_r.

antithetic sampling with just $N_t = 2$ correlated paths was best for many cases and especially for perfect heterodyne mode.

Second, we examine the impact of varying the maximum shift mapping depth K_d (Fig. 11-(a-c)). Maximum shift mapping depth $K_d = 0$ implies using no correlation at all, 1 implies camera ray (pixel) correlation, and so on until full path correlation (shift mapping). Figure 17 shows qualitative results. For $\omega_r \in [0.5, 1.0]$, using larger K_d gives better results, but the opposite is generally true for $\omega_r \in [0.0, 0.5]$. This change occurs because the variance of the modulation term becomes less dominant as $\omega_r \to 0$. Interestingly, completely abandoning variance from the modulation term ($K_d = 0$) gives the worst performance except over a small region near $\omega_r = 0$. This indicates that there should be the optimal balance between path correlation and diversity for $\omega_r \in [0.0, 0.5]$, but this seems to be highly scene-dependent, making it hard to find an absolute rule.

6.3 Shift Mapping Strategy Comparison

In Fig. 19, we compare the different shift mapping strategies we introduced in Sec. 5. In general, if specular materials are dominant in the scene, random replay works better; conversely, path reconnection works better for scenes with mostly diffuse materials. However, exceptions can arise: In Fig. 18, even though the scene is diffuse, there is a region where path reconnection fails. In this region, the

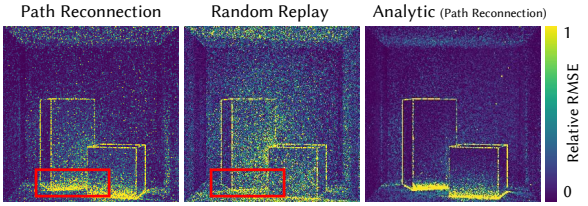

Fig. 18. Comparison of different shift mapping methods. The scene is made up of diffuse materials, so path reconnection works better than random replay. However, it fails near the region highlighted in red. There, the lengths of interreflected rays are short, and thus path reconnection causes a large change in $\widehat{f}(\bar{\mathbf{x}}(t))$, which increases variance.

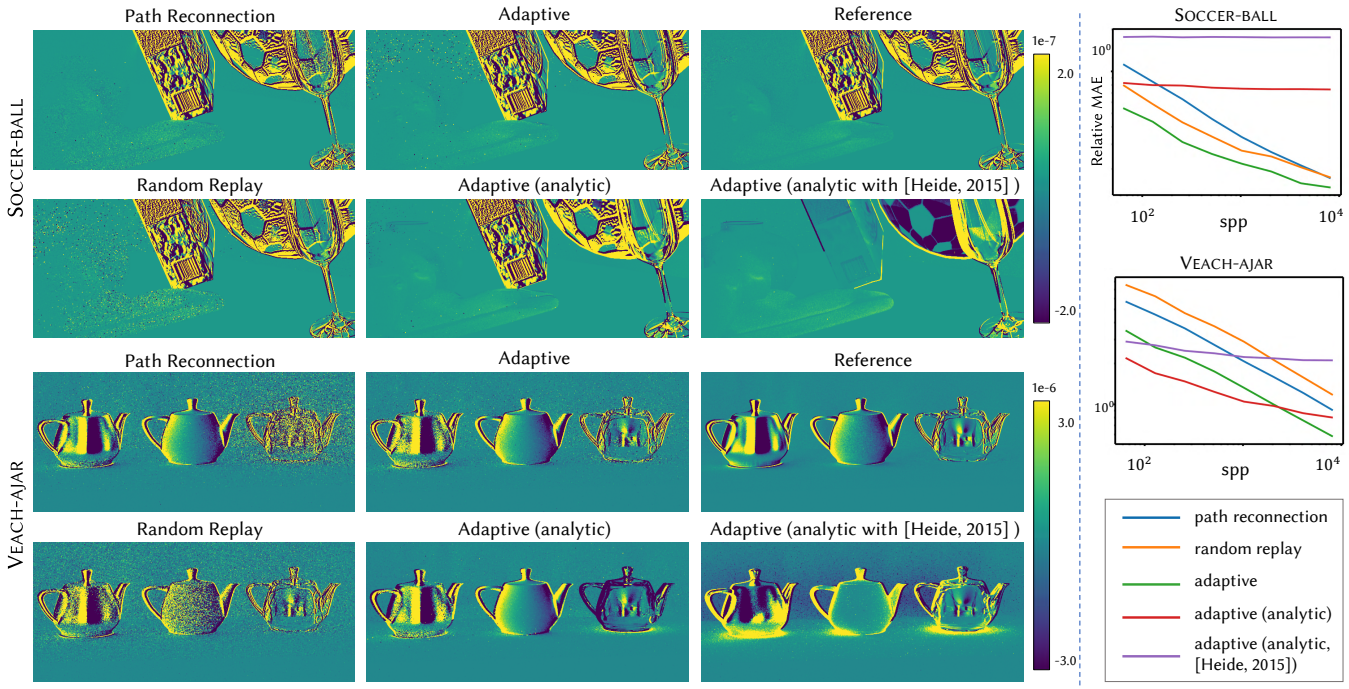

Fig. 19. Qualitative result and relative MAE using different shift mapping strategies. We use MAE instead of RMSE because RMSE overestimated the failure case depicted in Fig. 18. Among unbiased techniques, adaptive correlation works best in most cases. The integration-based method (first-order approximation) shows bias similar to Heide et al. [2015], but has significantly lower error than their zero-order approximation of integrand.

length of interreflected rays is small, and thus geometric attenuation varies significantly for path reconnection. Adaptive strategy generally works better than either path reconnection or random replay. For all three methods, MAE (Fig. 19, right column plots) decreases at the rate of $1/\sqrt{\text{spp}}$, showing that they are unbiased.

6.4 Analytic Approximation

In Fig. 19, we include results to show the speed-up and bias of our analytic approximation. This approximation results in a less noisy image compared to Monte Carlo methods but shows noticeable bias in both the images and the MAE plot over spp. Our first-order approximation still gives better results compared to the zeroth-order approximation from Heide et al. [2015], which demonstrates the importance of considering non-constant $\widetilde{f}(\overline{\mathbf{x}}(t))$ over exposure time.

6.5 High Frequency Terms and Precision Issue

As D-ToF rendering handles very small numerical values, such as v/c, it is vulnerable to numerical precision issues. We run our algorithm in both 32 bit (Mitsuba 3, CUDA) and 64 bit (Mitsuba 0.6, CPU) floating-point precision, and compare rendered results in Fig. 20. We also compare rendered results without ignoring the high-frequency terms in Eq. (7), which are most sensitive to such precision issues.

In the full signal rendering without low-pass filtering, we notice strong ringing artifacts when using 32 bit precision, which persists as we increase the number of samples. We found that scaling up the scene helps resolve these artifacts—generally, a 10 – 100× scaling

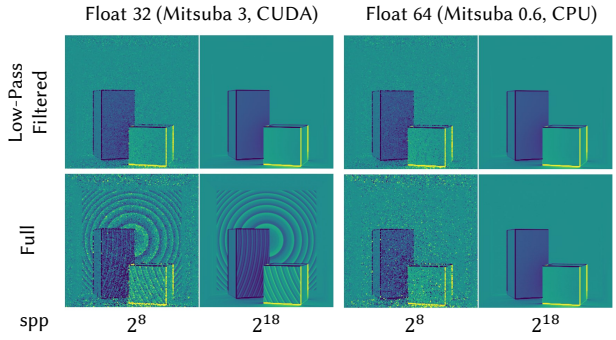

Fig. 20. We compare renderings at `float32` and `float64` precision, for the low-pass filtered and full D-ToF integral. The full rendering at `float32` has ringing artifacts, which we can resolve by increasing precision or removing high frequencies, at the cost of longer runtimes and bias, respectively.

was effective, but we could not increase the scale arbitrarily due to intersection precision problems. For 64 bit precision, such artifacts do not occur, but noise is increased universally. Low-pass filtering the modulation term as Eq. (8) resolves ringing artifact and noise with the cost of negligible bias (< 1%) for both precisions.

6.6 Area Light Sources

To demonstrate that our algorithm can handle a variety of light sources and materials, in Fig. 21 we render the CORNELL-BOX and

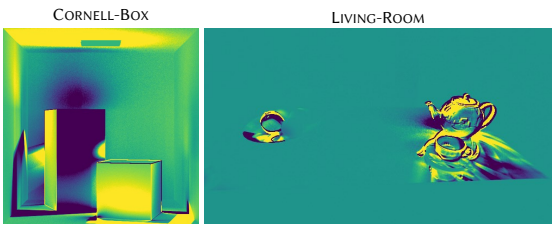

Fig. 21. An example scene using area light source demonstrating that the renderer can handle a variety of light sources along with material properties.

LIVING-ROOM scenes with area light sources. The figure illustrates the complex interplay of diverse light paths. Overall, it took more time (16–64×) to converge compared to a point light.

7 APPLICATION TO RADIAL VELOCITY ESTIMATION

In this section, we reproduce the radial velocity estimation algorithms from previous D-ToF imaging systems. We identify conditions under which these algorithms fail and explain the failure cases. With the help of our D-ToF rendering engine, we identify new imaging system parameters that, if realized experimentally, can make the velocity estimation more accurate.

7.1 Reproducing Existing D-ToF Imaging System

We first use simulation to reproduce the systems and algorithms of Heide et al. [2015] and Hu et al. [2022] in virtual experiments, using the same experimental setup as Hu et al. ($\omega_g = 74\,\text{MHz}$, $T = 2\,\text{ms}$) with collocated sensor and point light source. For Heide et al., we use the ratio of homodyne ($\omega_r = 0.0$) to heterodyne ($\omega_r = 1.0$) measurements to estimate radial velocity; for Hu et al. we use the ratio of $\omega_r = 0.6625$ (optimal for sinusoidal) measurements with four different ψ offsets. We tested five different $\Delta\omega$s, $-10\,\text{Hz}$, $-5\,\text{Hz}$, $0\,\text{Hz}$, $5\,\text{Hz}$ and $10\,\text{Hz}$ over $\psi \in [0, \pi]$. Figure 22 shows the simulated result, which is similar to Fig. 8 from Hu et al. [2022]. The measurement ratio of Hu et al.'s (r_2) shows a larger margin compared to Heide et al.'s ratio (r_{OF}), which implies that Hu et al.'s technique can better separate different Doppler frequencies. For more details, we refer to Hu et al. [2022]. We note that we did not model the sensor noise profile. The noise in Fig. 22 is Monte Carlo noise, not simulated sensor shot noise.

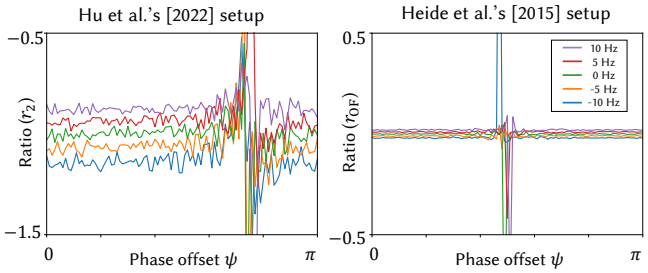

Fig. 22. We reproduce the hardware results from Hu et al. [2022] with our renderer. This figure illustrates that Hu et al.'s technique can separate Doppler frequencies better than Heide et al.'s [2015] technique.

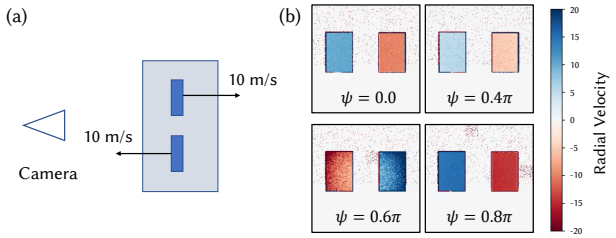

Fig. 23. We observe that for the same scene configuration shown in (a), the reconstructed radial velocity in (b) varies significantly for different phase offset ψ values. Only the left top corner ($\psi = 0$) gives the correct velocity. We only show result of Heide et al. [2015], but similar for Hu et al. [2022].

For both methods, however, we found that velocity estimation works well only for certain offsets (Fig. 23). For others, the reconstructed velocity direction can even be reversed. One interpretation is that such deviations are simply variance, but next we provide a better interpretation in terms of the dynamic ToF path integral.

7.2 Limitations of Velocity Estimation Algorithms

Existing velocity estimation algorithms for D-ToF imaging systems [Heide et al. 2015; Hu et al. 2022] assume that the object normal is parallel to the viewing direction, the light sources are directional, and global illumination is absent. When these assumptions fail, the velocity estimation algorithms become inaccurate. In this section, we use our rendering algorithm to investigate each factor in turn.

Effective Radial Velocity. First, we clarify the exact definition of the velocity that Doppler imaging estimates. We consider Fig. 24-(a), where the camera $\mathbf{x}_c$ and point light source $\mathbf{x}_e$ are collocated, and locality is preserved over single-bounce paths $\mathbf{x}_c\mathbf{x}(t)\mathbf{x}_e$. Then:

$$\mathbf{x}(t) = \mathbf{x}(0) + \mathbf{d}ut \;\text{ and }\; (\mathbf{x}(0) + \mathbf{v}t - \mathbf{x}(t)) \cdot \mathbf{n} = 0, \tag{21}$$

where $\mathbf{d}$ is the ray direction, $\mathbf{n}$ is the surface normal, $\mathbf{v}$ is the object velocity, and u is the *effective radial speed* in direction $\mathbf{d}$ which equals

$$u = \frac{(\mathbf{n} \cdot \mathbf{v})}{(\mathbf{n} \cdot \mathbf{d})}. \tag{22}$$

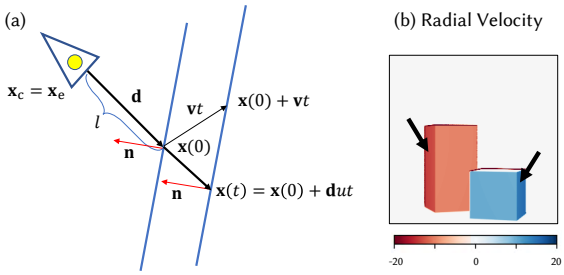

Fig. 24. Effective radial velocity for a planar patch. The Doppler imaging algorithms assume that the surface normal of the object aligns with the viewing direction. However, for a general planar patch shown in (a), the effective radial speed $u = (\mathbf{n} \cdot \mathbf{v})/(\mathbf{n} \cdot \mathbf{d})$. This results in large radial velocity estimates for objects viewed at grazing angle as shown in (b).

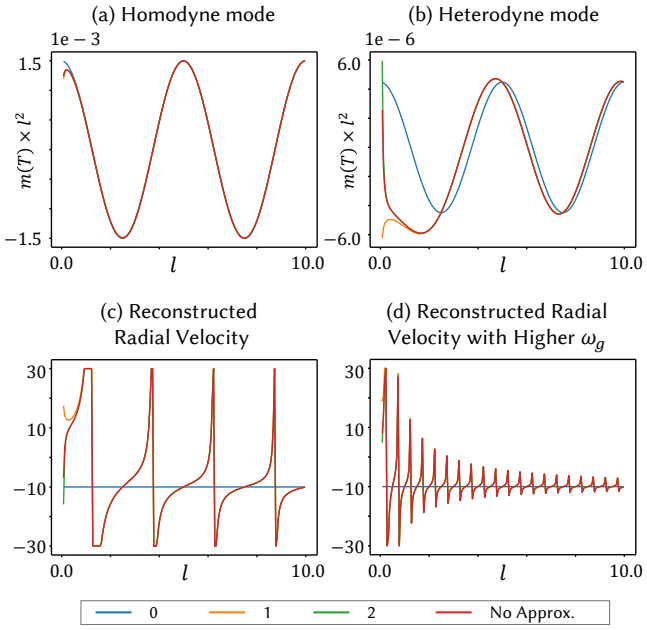

Fig. 25. Effect of geometric attenuation term. We multiplied $m(T)$ by l^2 to aid visualization. We show the integrated radiance for homodyne and heterodyne mode for 0, 1, 2-order Taylor series approximations of the geometric attenuation term. Previous velocity computation techniques derived velocity expressions assuming 0-order approximation for the forward model. These algorithms fail when the approximation is not valid.

This is different from the actual *radial speed*, $(\mathbf{v} \cdot \mathbf{d})$. We refer to the velocity vectors associated with these speeds as the effective radial velocity and radial velocity, respectively. Therefore, the effective radial speed is equal to the radial speed only if the object normal aligns with the viewing direction. At grazing angles (Fig. 24 (b)), the effective radial speed previous techniques report could be larger than the object's radial speed (side of the box in Cornell-box).

Time-varying Geometric Attenuation. We proceed to analyze $\widehat{f}$ in Eq. (9) for single-bounce paths in Fig. 24-(a).The distance toward the object at $t = 0$ is l and the object is moving away from the camera with effective radial speed u. Then we can write $\widehat{f}$ in Eq. (9) as

$$\widehat{f}(\overline{\mathbf{x}}(t)) = \frac{f_r(\mathbf{x}(t), -\mathbf{d}, -\mathbf{d})(\mathbf{n} \cdot -\mathbf{d})}{(l + ut)^2} \tag{23}$$

where f_r is the BRDF. We assume f_r and $\mathbf{n}$ are constant over t, so that only the geometric attenuation term $(l + ut)^{-2}$ affects $\widehat{f}$. Both Heide et al. [2015] and Hu et al. [2022] assume a constant geometric attenuation term and integrate cosine modulation terms over time, which is accurate if either the distance l is large, the light source is directional, or the cosine frequency is small (homodyne). In terms of Taylor series expansion, this is a zeroth-order approximation of $(l + ut)^{-2}$. On the other hand, our approximate analytic expression uses first-order approximation. To validate our approximation, we calculated $m(T)$ with different approximation orders.

Figure 25-(a,b) show analytically integrated $m(T)$ for homodyne and heterodyne modes. The red line is the ground truth value, while lines with labels 0, 1, 2 represent different order Taylor-series approximations. For homodyne mode, even the zeroth-order Taylor approximation is accurate and hence, CW-ToF cameras that use homodyne for depth measurement do not suffer from this problem. However, there is a lot of deviation in the heterodyne mode, especially when we use zeroth-order approximation. The first-order approximation also fails at close distances, but overall, it is significantly more accurate than the zeroth-order, which explains why our analytic integration outperforms Heide et al. [2015].

Figure 25-(c) shows the result for radial velocity calculation. The physically accurate $m(T)$ gives significantly deviated velocity with many discontinuities, implying that velocity computation techniques fail when geometric attenuation (or in general, path throughput $\widehat{f}$) varies over exposure time. This is why the existing ratio-based techniques fail for some ψs in Fig. 23.

Global Illumination. Another scenario where the velocity estimation fails is in the presence of strong global illumination. In Fig. 26, we show how global illumination results in inaccurate velocity estimates. Quantifying the impact of global illumination (usually termed *multi-path interference* (MPI) in ToF literature) is crucial for ToF applications, as evinced by the extensive prior work on suppressing MPI in CW-ToF imaging [Kadambi et al. 2013; Whyte et al. 2015].

7.3 Improving Radial Velocity Calculation

To mitigate the challenges imposed by time-varying geometric attenuation and global illumination, we propose a few solutions that require operating the D-ToF camera at high frequency, which is feasible with recent advances in CW-ToF cameras [Baek et al. 2023].

Mitigating Effects due to Time-varying Geometric Attenuation. One way to mitigate the effects of time-varying geometric attenuation on velocity estimation is to increase the working distance or use a directional light, both of which make geometric attenuation (approximately) constant. Another approach is to increase the modulation frequency ω_g. From Fig. 25 (d), we observe that increasing this frequency makes the velocity estimation error small over a larger working distance range. However, increasing modulation frequency makes homodyne mode become zero more frequently, which causes

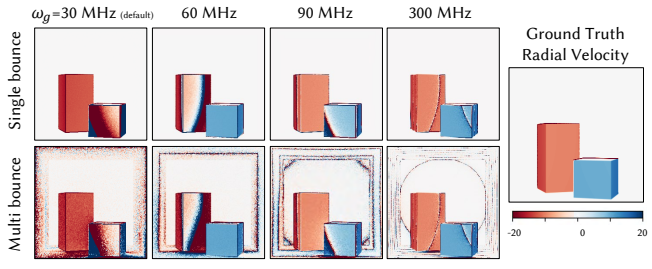

Fig. 26. In the presence of global illumination (multi-bounce), velocity estimation algorithms fail. We observed, however, that using higher modulation frequency illumination diminishes the global illumination effects.

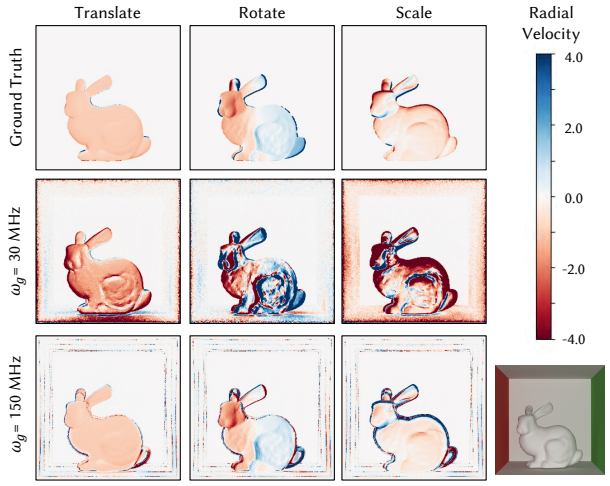

Fig. 27. For various object motions (translation, rotation, scaling), we observed that global illumination significantly affects velocity computation at low illumination frequency but not at high illumination frequency.

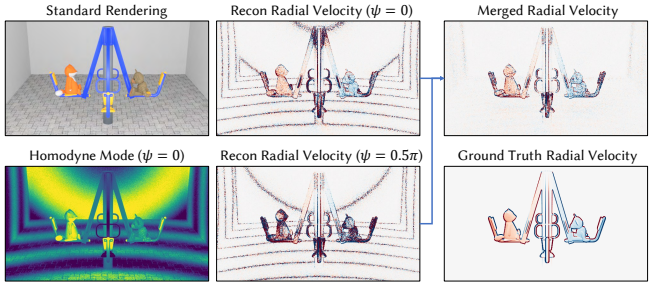

Fig. 28. Homodyne mode has zero values at certain positions which causes ringing artifact in velocity reconstruction. This could be improved by using two different phases and mixing them. The rendering time for this scene with 22 k triangles and $N = 1024$ is 4 sec per frame.

numerical precision problems resulting in noticeable ringing artifacts. We can resolve these problems by using two different phases which have zero homodyne values at different positions (Fig. 28) and combining them as in the *six-sample method* in Hu et al. [2022].

Mitigating Global Illumination. We found that using higher frequencies also mitigates global illumination effects. In Fig. 26, we show that increasing the modulation frequency reduces the inaccuracy of velocity calculation due to global illumination. Figure 27 shows another example under various scene motions. This behavior is because contributions from multi-bounce paths tend to cancel out as modulation frequency increases. The same effect is already known in structured light [Nayar et al. 2006] and CW-ToF [Gupta et al. 2015] imaging systems and it is exciting to see that it also holds true for D-ToF imaging systems.

Finally, we run our rendering algorithm and velocity estimation techniques on more complex scene geometries under several frames and animate the results (Fig. 1, Fig. 28). Our algorithm reliably and efficiently reproduces both D-ToF images and radial velocity. We provide more videos in the supplement.

8 CONCLUSION

We developed a D-ToF Monte Carlo rendering framework and tailored sampling techniques for efficiently simulating D-ToF cameras. Using our open source implementation, we showed that our techniques provide orders of magnitude improved performance compared to naive sampling techniques, under various illumination and sensor modulation functions. We additionally reproduced in simulation previously-reported results from real D-ToF hardware systems [Heide et al. 2015; Hu et al. 2022], and investigated the accuracy of their velocity estimation on a variety of scenes.

Our work suggests several directions for future research. Recent advances in path reuse and shift mapping techniques, for both static and animated scenes [Bitterli et al. 2020; Sawhney et al. 2022], could be adapted into temporal mapping techniques for D-ToF rendering, to improve rendering performance for challenging scenes or more general modulation waveforms. Additionally, our D-ToF path integral framework invites the application to D-ToF rendering of other Monte Carlo algorithms, such as Markov chain Monte Carlo algorithms that have proven effective also for the related motion blur rendering problem [Li et al. 2010; Luan et al. 2020].

Differentiable D-ToF rendering is another intriguing future research direction, which could facilitate the design of D-ToF imaging systems and related inverse rendering applications. The D-ToF integral in Eq. (3) is differentiable when the illumination and exposure codes are differentiable. However, as the integrand can become negative, building efficient path sampling techniques is non-trivial [Chang et al. 2023; Zhang et al. 2021]. Additional challenges arise for discontinuous illumination and exposure waveforms (e.g., square): differentiating Eq. (3) then results in singularities in the time-domain, which require sampling on delta manifolds similar to those studied by Pediredla et al. [2019b] and Wu et al. [2021].

As light-based velocity sensing becomes commonplace in critical applications (autonomous vehicles, robotic navigation, remote sensing), we expect that our work will inspire rendering research for other common or emerging technologies for this sensing modality. An example is rendering frequency-modulated continuous-wave (FMCW) ToF sensors [Qian et al. 2022]: even though their operation is also based on the Doppler effect and correlation measurements, these sensors are interferometric [Fercher et al. 2003; Kotwal et al. 2020], thus simulating them requires rendering challenging wave effects [Bar et al. 2019; Steinberg and Yan 2021].

Last but not least, our open-source simulator can facilitate research and engineering efforts toward designing and optimizing all aspects of future D-ToF computational imaging systems. Examples include the design of new sensor architectures, D-ToF modulation functions, and velocity estimation algorithms that are robust to noise and global illumination. In this context, our simulator can act as a digital twin that enables quick prototyping, supervised data generation, quantitative evaluation, and even end-to-end optimization of both hardware and software components of a real D-ToF computational imaging pipeline. Towards realizing these applications, it will be important to research and incorporate into our renderer realistic sensor noise models for D-ToF imaging.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their feedback and especially Anonymous Reviewer 4 for the idea to use scene scaling in Sec. 6.5 to mitigate the ringing artifacts due to limited floating-point precision. We also thank the authors of Hu et al. [2022] for helpful discussions on Sec. 7, and in particular correcting an error in Fig. 25. Wojciech Jarosz was supported by NSF award 1844538, Ioannis Gkioulekas by NSF awards 1730147, 1900849, and a Sloan Research Fellowship, and Adithya Pediredla by a Burke research initiation award.

REFERENCES

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

Seung-Hwan Baek, Noah Walsh, Ilya Chugunov, Zheng Shi, and Felix Heide. 2023. Centimeter-Wave Free-Space Neural Time-of-Flight Imaging. *ACM Transactions on Graphics* 42, 1 (March 2023), 3:1–3:18. https://doi.org/10/gspq49

Sai Praveen Bangaru, Tzu-Mao Li, and Frédo Durand. 2020. Unbiased Warped-Area Sampling for Differentiable Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 39, 6 (Nov. 2020), 245:1–245:18. https://doi.org/10/gj32sk

Chen Bar, Marina Alterman, Ioannis Gkioulekas, and Anat Levin. 2019. A Monte Carlo Framework for Rendering Speckle Statistics in Scattering Media. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 38, 4 (July 2019). https://doi.org/10/gf5jbk

Benedikt Bitterli, Wenzel Jakob, Jan Novák, and Wojciech Jarosz. 2018. Reversible Jump Metropolis Light Transport Using Inverse Mappings. *ACM Transactions on Graphics* 37, 1 (Jan. 2018), 1:1–1:12. https://doi.org/10/gd52ph

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

Wesley Chang, Venkataram Sivaram, Derek Nowrouzezahrai, Toshiya Hachisuka, Ravi Ramamoorthi, and Tzu-Mao Li. 2023. Parameter-Space ReSTIR for Differentiable and Inverse Rendering. In *ACM SIGGRAPH Conference Papers*. ACM Press, Los Angeles, CA. https://doi.org/10/kn67

Wenzheng Chen, Fangyin Wei, Kiriakos N. Kutulakos, Szymon Rusinkiewicz, and Felix Heide. 2020. Learned Feature Embeddings for Non-Line-of-Sight Imaging and Recognition. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 39, 6 (Nov. 2020). https://doi.org/10/ghsr98

Robert L. Cook, Thomas Porter, and Loren Carpenter. 1984. Distributed Ray Tracing. *Computer Graphics (Proceedings of SIGGRAPH)* 18, 3 (July 1984), 137–145. https://doi.org/10/c9thc3

Kevin Egan, Yu-Ting Tseng, Nicolas Holzschuch, Frédo Durand, and Ravi Ramamoorthi. 2009. Frequency Analysis and Sheared Reconstruction for Rendering Motion Blur. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 28, 3 (July 2009), 93:1–93:13. https://doi.org/10/dn2gcq

Adolf F Fercher, Wolfgang Drexler, Christoph K Hitzenberger, and Theo Lasser. 2003. Optical Coherence Tomography-Principles and Applications. *Reports on Progress in Physics* 66, 2 (2003), 239. https://doi.org/10/d5pmdp

Sergi Foix, Guillem Alenya, and Carme Torras. 2011. Lock-in Time-of-Flight (ToF) Cameras: A Survey. *IEEE Sensors Journal* 11, 9 (2011), 1917–1926. https://doi.org/10/dgfc2k

Salih Burak Gokturk, Hakan Yalcin, and Cyrus Bamji. 2004. A Time-of-Flight Depth Sensor - System Description, Issues and Solutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Washington, DC, USA, 35–35. https://doi.org/10/fcqz6m

Anant Gupta, Atul Ingle, and Mohit Gupta. 2019. Asynchronous Single-Photon 3D Imaging. In *Proceedings of the International Conference on Computer Vision (ICCV)*. IEEE Computer Society, 7909–7918. https://doi.org/10/ghfh3t

Mohit Gupta, Shree K Nayar, Matthias B Hullin, and Jaime Martin. 2015. Phasor Imaging: A Generalization of Correlation-Based Time-of-Flight Imaging. *ACM Transactions on Graphics* 34, 5 (2015). https://doi.org/10/gfz5k7

Felipe Gutierrez-Barragan, Huaijin Chen, Mohit Gupta, Andreas Velten, and Jinwei Gu. 2021. iToF2dToF: A Robust and Flexible Representation for Data-Driven Time-of-Flight Imaging. *IEEE Transactions on Computational Imaging* 7 (2021), 1205–1214. https://doi.org/10/gspq33

Toshiya Hachisuka, Wojciech Jarosz, Richard Peter Weistroffer, Kevin Dale, Greg Humphreys, Matthias Zwicker, and Henrik Wann Jensen. 2008. Multidimensional Adaptive Sampling and Reconstruction for Ray Tracing. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 27, 3 (Aug. 2008). https://doi.org/10/fm6c2w

John Michael Hammersley and Keith William Morton. 1956. A New Monte Carlo Technique: Antithetic Variates. *Mathematical Proceedings of the Cambridge Philosophical Society* 52, 03 (July 1956), 449–475. https://doi.org/10/dshxdn

Felix Heide, Wolfgang Heidrich, Matthias Hullin, and Gordon Wetzstein. 2015. Doppler Time-of-Flight Imaging. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 34, 4 (2015). https://doi.org/10/gfz5k9

Yunpu Hu, Leo Miyashita, and Masatoshi Ishikawa. 2022. Differential Frequency Heterodyne Time-of-Flight Imaging for Instantaneous Depth and Velocity Estimation. *ACM Transactions on Graphics* 42, 1 (Sept. 2022), 9:1–9:13. https://doi.org/10/gspq2z

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). https://doi.org/10/gspq3m

Wenzel Jakob. 2013. Mitsuba Renderer. http://www.mitsuba-renderer.org

Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. 2022. Mitsuba 3 Renderer. https://mitsuba-renderer.org

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

Wojciech Jarosz, Derek Nowrouzezahrai, Iman Sadeghi, and Henrik Wann Jensen. 2011. A Comprehensive Theory of Volumetric Radiance Estimation Using Photon Points and Beams. *ACM Transactions on Graphics* 30, 1 (Jan. 2011). https://doi.org/10/fcdh2f

Wojciech Jarosz, Matthias Zwicker, and Henrik Wann Jensen. 2008. The Beam Radiance Estimate for Volumetric Photon Mapping. *Computer Graphics Forum (Proceedings of Eurographics)* 27, 2 (April 2008), 557–566. https://doi.org/10/bjsfsx

Henrik Wann Jensen. 2001. *Realistic Image Synthesis Using Photon Mapping*. AK Peters, Ltd., Natick, MA, USA.

Henrik Wann Jensen and Per H. Christensen. 1998. Efficient Simulation of Light Transport in Scenes with Participating Media Using Photon Maps. In *Annual Conference Series (Proceedings of SIGGRAPH)*. ACM Press, 311–320. https://doi.org/10/b64p36

Achuta Kadambi, Refael Whyte, Ayush Bhandari, Lee Streeter, Christopher Barsi, Adrian Dorrington, and Ramesh Raskar. 2013. Coded Time of Flight Cameras: Sparse Deconvolution to Address Multipath Interference and Recover Time Profiles. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 32, 6 (2013). https://doi.org/10/gfz5mk

Achuta Kadambi, Hang Zhao, Boxin Shi, and Ramesh Raskar. 2016. Occluded Imaging with Time-of-Flight Sensors. *ACM Transactions on Graphics* 35, 2 (March 2016). https://doi.org/10/f8q5jg

Csaba Kelemen, László Szirmay-Kalos, György Antal, and Ferenc Csonka. 2002. A Simple and Robust Mutation Strategy for the Metropolis Light Transport Algorithm. *Computer Graphics Forum* 21, 3 (Sept. 2002), 531–540. https://doi.org/10/bfrsqn

Markus Kettunen, Marco Manzi, Miika Aittala, Jaakko Lehtinen, Frédo Durand, and Matthias Zwicker. 2015. Gradient-Domain Path Tracing. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 34, 4 (July 2015), 123. https://doi.org/10/gfzrhn

Alankar Kotwal, Anat Levin, and Ioannis Gkioulekas. 2020. Interferometric Transmission Probing with Coded Mutual Intensity. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 39, 4 (July 2020). https://doi.org/10/gg8xcj

Robert Lange and Peter Seitz. 2001. Solid-State Time-of-Flight Range Camera. *IEEE Journal of quantum electronics* 37, 3 (2001), 390–397. https://doi.org/10/bd236w

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

Hongwei Li, Li-Yi Wei, Pedro V. Sander, and Chi-Wing Fu. 2010. Anisotropic Blue Noise Sampling. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 29, 6 (Dec. 2010), 167:1–167:12. https://doi.org/10/c8npd7

Jingyu Lin, Yebin Liu, Jinli Suo, and Qionghai Dai. 2016. Frequency-Domain Transient Imaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP, 99 (2016). https://doi.org/10/f95dvd

Xiaochun Liu, Ibón Guillén, Marco La Manna, Ji Hyun Nam, Syed Azer Reza, Toan Huu Le, Adrian Jarabo, Diego Gutierrez, and Andreas Velten. 2019. Non-Line-of-Sight Imaging Using Phasor-Field Virtual Wave Optics. *Nature* 572, 7771 (2019), 620–623. https://doi.org/10/gj7bhd

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

Fujun Luan, Shuang Zhao, Kavita Bala, and Ioannis Gkioulekas. 2020. Langevin Monte Carlo Rendering with Gradient-Based Adaptation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 39, 4 (July 2020). https://doi.org/10/gg8xcw

Marco Manzi, Markus Kettunen, Frédo Durand, Matthias Zwicker, and Jaakko Lehtinen. 2016. Temporal Gradient-Domain Path Tracing. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 35, 6 (Dec. 2016). https://doi.org/10/f9cpsw

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

Julio Marco, Quercus Hernandez, Adolfo Muñoz, Yue Dong, Adrian Jarabo, Min H. Kim, Xin Tong, and Diego Gutierrez. 2017a. DeepToF: Off-the-shelf Real-Time Correction of Multipath Interference in Time-of-Flight Imaging. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 36, 6 (Nov. 2017). https://doi.org/10/ggfg4v

Julio Marco, Wojciech Jarosz, Diego Gutierrez, and Adrian Jarabo. 2017b. Transient Photon Beams. In *Congreso Espanol de Informatica Grafica*. Eurographics Association. https://doi.org/10/gfzncc

Don P. Mitchell. 1996. Consequences of Stratified Sampling in Graphics. In *Annual Conference Series (Proceedings of SIGGRAPH)*, Holly Rushmeier (Ed.). Addison-Wesley, 277–280. https://doi.org/10/dkw86x

Nikhil Naik, Christopher Barsi, Andreas Velten, and Ramesh Raskar. 2014. Estimating Wide-Angle, Spatially Varying Reflectance Using Time-Resolved Inversion of Backscattered Light. *Journal of the Optical Society of America A* 31, 5 (2014). https://doi.org/10/gfz5mw

Fernando Navarro, Francisco J. Seron, and Diego Gutierrez. 2011. Motion Blur Rendering: State of the Art. *Computer Graphics Forum* 30, 1 (2011). https://doi.org/10/b574px

Shree K Nayar, Gurunandan Krishnan, Michael D Grossberg, and Ramesh Raskar. 2006. Fast Separation of Direct and Global Components of a Scene Using High Frequency Illumination. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 25, 3 (2006), 935–944. https://doi.org/10/fw3cpp

Matthew O'Toole, Felix Heide, David B. Lindell, Kai Zang, Steven Diamond, and Gordon Wetzstein. 2017. Reconstructing Transient Images from Single-Photon Sensors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, 2289–2297. https://doi.org/10/gspq4f

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). https://doi.org/10/gfz5m2

Matthew O'Toole, David B. Lindell, and Gordon Wetzstein. 2018. Confocal Non-Line-of-Sight Imaging Based on the Light-Cone Transform. *Nature* 555, 7696 (March 2018), 338–341. https://doi.org/10/gc3gs5

A. Cengiz Öztireli. 2016. Integration with Stochastic Point Processes. *ACM Transactions on Graphics* 35, 5 (Aug. 2016), 160:1–160:16. https://doi.org/10/f85k3g

Xian Pan, Victor Arellano, and Adrian Jarabo. 2019. Transient Instant Radiosity for Efficient Time-Resolved Global Illumination. *Computers & Graphics* 83 (Oct. 2019), 107–113. https://doi.org/10/gspq3t

Mark Pauly, Thomas Kollig, and Alexander Keller. 2000. Metropolis Light Transport for Participating Media. In *Rendering Techniques (Proceedings of the Eurographics Workshop on Rendering)*. Springer-Verlag, Vienna, 11–22. https://doi.org/10/gfzm93

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)*. IEEE Computer Society, 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). https://doi.org/10/gf5jbm

Adithya K. Pediredla. 2019. *Rendering and Imaging with Transients: A Study of Computational Photography with Scattered Photons.* Ph. D. Dissertation. Rice University. https://hdl.handle.net/1911/105936

Adithya Kumar Pediredla, Mauro Buttafava, Alberto Tosi, Oliver Cossairt, and Ashok Veeraraghavan. 2017. Reconstructing Rooms Using Photon Echoes: A Plane Based Model and Reconstruction Algorithm for Looking around the Corner. In *IEEE International Conference on Computational Photography (ICCP)*. IEEE Computer Society. https://doi.org/10/gsscxg

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

Markus Plack, Clara Callenberg, Monika Schneider, and Matthias B. Hullin. 2023. Fast Differentiable Transient Rendering for Non-Line-of-Sight Reconstruction. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 3066–3075. https://doi.org/10/gspq3s

Ryan Po, Adithya Pediredla, and Ioannis Gkioulekas. 2022. Adaptive Gating for Single-Photon 3D Imaging. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 16333–16342. https://doi.org/10/gspq4b

A Prusak, O Melnychuk, H Roth, Ingo Schiller, and Reinhard Koch. 2008. Pose Estimation and Map Building with a Time-of-Flight-Camera for Robot Navigation. *International Journal of Intelligent Systems Technologies and Applications* 5, 3-4 (2008), 355–364. https://doi.org/10/ftcf67

Ruobing Qian, Kevin C Zhou, Jingkai Zhang, Christian Viehland, Al-Hafeez Dhalla, and Joseph A Izatt. 2022. Video-Rate High-Precision Time-Frequency Multiplexed 3D Coherent Ranging. *Nature Communications* 13, 1 (2022), 1476. https://doi.org/10/gpznk8

Ankit Raghuram, Adithya Pediredla, Srinivasa G. Narasimhan, Ioannis Gkioulekas, and Ashok Veeraraghavan. 2019. STORM: Super-resolving Transients by OveRsampled Measurements. In *IEEE International Conference on Computational Photography (ICCP)*. 1–11. https://doi.org/10/gk48pq

Ravi Ramamoorthi, John Anderson, Mark Meyer, and Derek Nowrouzezahrai. 2012. A Theory of Monte Carlo Visibility Sampling. *ACM Transactions on Graphics* 31, 5 (Sept. 2012). https://doi.org/10/gbbrnz

Guy Satat, Barmak Heshmat, Dan Raviv, and Ramesh Raskar. 2016. All Photons Imaging through Volumetric Scattering. *Scientific Reports* 6, 1 (Sept. 2016), 33946. https://doi.org/10/f85wj2

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/kt6s arXiv:2211.00166 [cs.GR]

Shikhar Shrestha, Felix Heide, Wolfgang Heidrich, and Gordon Wetzstein. 2016. Computational Imaging with Multi-Camera Time-of-Flight Systems. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 35, 4 (2016). https://doi.org/10/gfz5m9

Gurprit Singh, Cengiz Öztireli, Abdalla G.M. Ahmed, David Coeurjolly, Kartic Subr, Oliver Deussen, Victor Ostromoukhov, Ravi Ramamoorthi, and Wojciech Jarosz. 2019. Analysis of Sample Correlations for Monte Carlo Rendering. *Computer Graphics Forum (Proceedings of Eurographics State of the Art Reports)* 38, 2 (April 2019). https://doi.org/10/gf6rzc

Shlomi Steinberg and Ling-Qi Yan. 2021. A Generic Framework for Physical Light Transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 40, 4 (July 2021). https://doi.org/10/gpzk56

Shuochen Su, Felix Heide, Gordon Wetzstein, and Wolfgang Heidrich. 2018. Deep End-to-End Time-of-Flight Imaging. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6383–6392. https://doi.org/10/gspq5b

Kartic Subr, Derek Nowrouzezahrai, Wojciech Jarosz, Jan Kautz, and Kenny Mitchell. 2014. Error Analysis of Estimators That Use Combinations of Stochastic Sampling Strategies for Direct Illumination. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)* 33, 4 (June 2014), 93–102. https://doi.org/10/f6fgw4

Chia-Yin Tsai, Aswin C. Sankaranarayanan, and Ioannis Gkioulekas. 2019. Beyond Volumetric Albedo — a Surface Optimization Framework for Non-Line-of-Sight Imaging. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1545–1555. https://doi.org/10/gspq3k

Eric Veach and Leonidas J. Guibas. 1995. Optimally Combining Sampling Techniques for Monte Carlo Rendering. In *Annual Conference Series (Proceedings of SIGGRAPH)*, Vol. 29. ACM Press, 419–428. https://doi.org/10/d7b6n4

Andreas Velten, Everett Lawson, Andrew Bardagjy, Moungi Bawendi, and Ramesh Raskar. 2011. Slow Art with a Trillion Frames per Second Camera. In *ACM SIGGRAPH Talks*.

Amanpreet Walia, Stefanie Walz, Mario Bijelic, Fahim Mannan, Frank Julca-Aguilar, Michael Langer, Werner Ritter, and Felix Heide. 2022. Gated2Gated: Self-supervised Depth Estimation from Gated Images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2801–2811. https://doi.org/10/gspq4j

Turner Whitted. 1980. An Improved Illumination Model for Shaded Display. *Commun. ACM* 23, 6 (June 1980), 343–349. https://doi.org/10/dgdpj7

Refael Whyte, Lee Streeter, Michael J Cree, and Adrian A Dorrington. 2015. Resolving Multiple Propagation Paths in Time of Flight Range Cameras Using Direct and Global Separation Methods. *Optical Engineering* 54, 11 (2015). https://doi.org/10/gfz5nf

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

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

Fang Yuan, Agnes Swadzba, Roland Philippsen, Orhan Engin, Marc Hanheide, and Sven Wachsmuth. 2009. Laser-Based Navigation Enhanced with 3D Time-of-Flight Data. In *IEEE International Conference on Robotics and Automation*. IEEE Computer Society, Kobe, 2844–2850. https://doi.org/10/c7np5j

Tizian Zeltner, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob. 2021. Monte Carlo Estimators for Differential Light Transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 40, 4 (July 2021), 78:1–78:16. https://doi.org/10/gqjn76

Cheng Zhang, Zhao Dong, Michael Doggett, and Shuang Zhao. 2021. Antithetic Sampling for Monte Carlo Differentiable Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 40, 4 (July 2021), 77:1–77:12. https://doi.org/10/gmmkx7

Cheng Zhang, Bailey Miller, Kai Yan, Ioannis Gkioulekas, and Shuang Zhao. 2020. Path-Space Differentiable Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 39, 4 (July 2020). https://doi.org/10/gg8xc2

Tianyi Zhang, Mel J. White, Akshat Dave, Shahaboddin Ghajari, Ankit Raghuram, Alyosha C. Molnar, and Ashok Veeraraghavan. 2022. First Arrival Differential LiDAR. In *IEEE International Conference on Computational Photography (ICCP)*. 1–12. https://doi.org/10/gspq38