

Unbiased Inverse Volume Rendering with Differential Trackers

MERLIN NIMIER-DAVID, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

THOMAS MÜLLER, NVIDIA, Switzerland

ALEXANDER KELLER, NVIDIA, Germany

WENZEL JAKOB, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

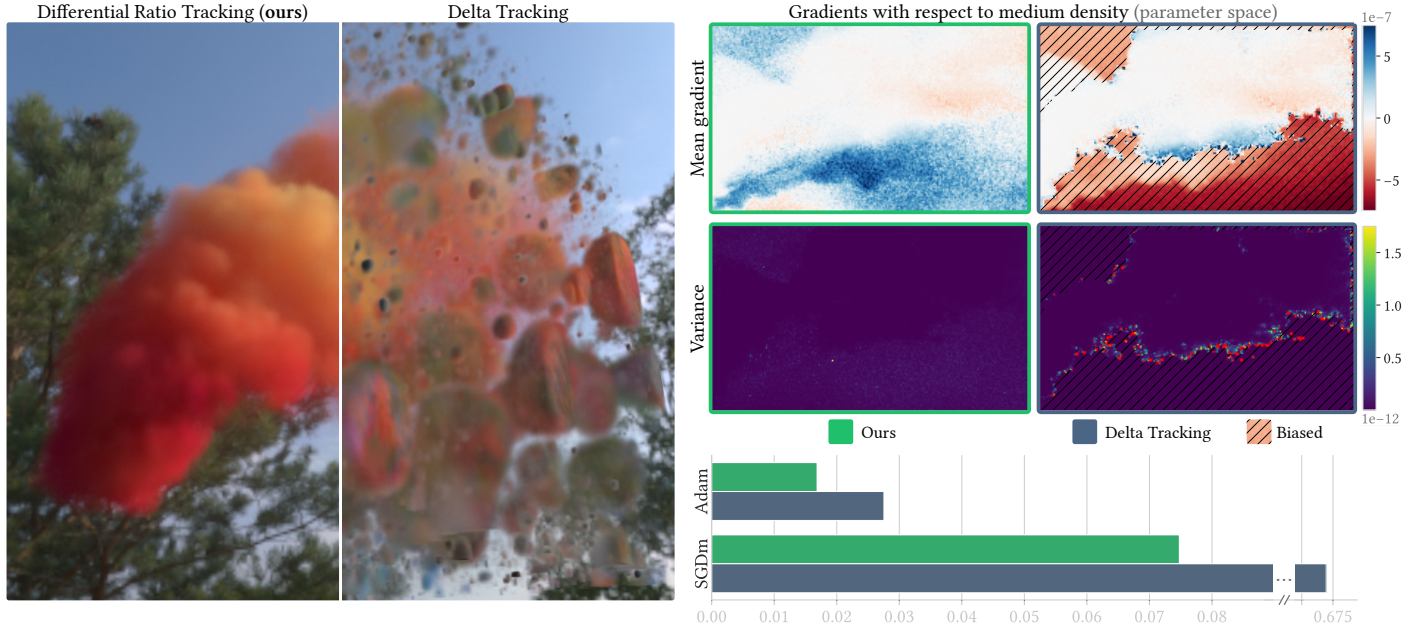


Fig. 1. We demonstrate the high-quality reconstruction of volumetric scattering parameters from RGB images with known camera poses (left). This is enabled by our novel differential ratio tracking formulation, which yields unbiased, low-variance gradients of the radiative transfer equation that can be directly used for optimization. Traditional free-flight sampling—e.g. by delta tracking—while effective at low-variance rendering, exhibits bias and high variance in gradient estimation with respect to medium density (top right), which negatively affects optimization. Gradient mean and variance values are shown for slice $z = 64$ of the $256 \times 128 \times 128$ parameter space. In the chart (bottom right), we report the improvements in reconstruction error for stochastic gradient descent with momentum (SGDm) as well as Adam. Using aggressive step size reduction, the Adam optimizer limits the impact of large gradient outliers, though our unbiased gradients lead to the lowest reconstruction error with either optimizer.

Volumetric representations are popular in inverse rendering because they have a simple parameterization, are smoothly varying, and transparently handle topology changes. However, incorporating the full volumetric transport of light is costly and challenging, often leading practitioners to implement simplified models, such as purely emissive and absorbing volumes with “baked” lighting. One such challenge is the efficient estimation of the gradients of the volume’s appearance with respect to its scattering and absorption parameters. We show that the straightforward approach—differentiating a volumetric free-flight sampler—can lead to biased and high-variance gradients, hindering optimization. Instead, we propose using a new sampling strategy: *differential ratio tracking*, which is unbiased, yields low-variance

gradients, and runs in linear time. Differential ratio tracking combines ratio tracking and reservoir sampling to estimate gradients by sampling distances proportional to the unweighted transmittance rather than the usual extinction-weighted transmittance. In addition, we observe local minima when optimizing scattering parameters to reproduce dense volumes or surfaces. We show that these local minima can be overcome by bootstrapping the optimization from nonphysical emissive volumes that are easily optimized.

CCS Concepts: • **Computing methodologies** → **Rendering**.

Additional Key Words and Phrases: differentiable rendering, inverse rendering, volumetric rendering, radiative backpropagation, importance sampling

ACM Reference Format:

Merlin Nimier-David, Thomas Müller, Alexander Keller, and Wenzel Jakob. 2022. Unbiased Inverse Volume Rendering with Differential Trackers. *ACM Trans. Graph.* 41, 4, Article 44 (July 2022), 20 pages. <https://doi.org/10.1145/3528223.3530073>

1 INTRODUCTION

3D reconstruction from images (inverse rendering) through differentiation has become an important method in a graphics practitioner’s

Authors’ addresses: Merlin Nimier-David, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, merlin.nimier-david@epfl.ch; Thomas Müller, NVIDIA, Zürich, Switzerland, tmuller@nvidia.com; Alexander Keller, NVIDIA, Berlin, Germany, akeller@nvidia.com; Wenzel Jakob, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, wenzel.jakob@epfl.ch.

© 2022 Copyright held by the owner/author(s). Publication rights licensed to ACM. This is the author’s version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Graphics*, <https://doi.org/10.1145/3528223.3530073>.

toolbox. The rapidly growing set of practical applications spans many fields, including light field capture [Mildenhall et al. 2020], 3D print optimization [Nindel et al. 2021], cloud tomography [Loeub et al. 2020; Ronen et al. 2021; Sde-Chen et al. 2021] non-line of sight imaging [Tsai et al. 2019], computational lens design [Sun et al. 2021b], as well as automatic simplification and reconstruction of video game assets [Hasselgren et al. 2021; Munkberg et al. 2021].

Volumetric inverse rendering approaches are particularly attractive due to their flexibility in representing both hard and soft surfaces, their inherent smoothness, as well as their invariance to topological changes. Nonetheless, the efficient evaluation of radiative transfer and its derivatives—as required in a gradient descent algorithm—remains difficult. Current methods generally follow two strategies to cope with this difficulty: (i) simplifying the model, such as by only permitting emission and absorption [Mildenhall et al. 2020] or limiting the permissible materials and illumination [Srinivasan et al. 2021], and (ii) developing more advanced algorithms for efficiently differentiating the radiative transfer equation in its general form [Nimier-David et al. 2020; Vicini et al. 2021b; Zhang et al. 2021b]. Our contributions pertain to both strategies.

First, we follow the general observation that forward-rendering strategies are not necessarily suitable for differential rendering [Zeltner et al. 2021; Zhang et al. 2021a], showing that standard volumetric free-flight sampling yields biased and high-variance gradients. This can be explained intuitively: free-flight sampling places samples on the portions of the volume contributing most to its appearance (formally: proportional to the product of transmittance and density). However, even in empty space, gradients may have non-trivial magnitude, e.g. to reconstruct an object where previously there was just air. In Section 4 we show that placing samples in *all* visible regions (proportional to just transmittance) significantly reduces the variance of these types of gradients. Based on this insight, we propose a novel sampling scheme, *differential ratio tracking*, that combines (residual) ratio tracking and reservoir sampling to sample distances proportional to transmittance. We use an additional weighted reservoir to integrate differential ratio tracking into the linear-time path replay backpropagation algorithm [Vicini et al. 2021b], leading to better behaved gradient descent optimization and lower reconstruction error with limited overhead, as shown in Figure 1. Our sampling scheme is dedicated to physically-based scattering and absorbing volumes (which are inherently relightable), as opposed to NeRF-style emissive volumes [Mildenhall et al. 2020].

Second, we observe that the loss landscape of the volumetric scattering coefficient contains local minima in which the optimization can get stuck. We propose alleviating this problem by first training a simplified model with known favorable convergence properties: a purely emissive and absorbing volume, similar to concurrent work on voxel-based radiance fields [Yu et al. 2021a]. We use the resulting densities to bootstrap our scattering volume optimization, optimizing only for the unknown scattering albedos. Our experiments show empirically that this procedure results in significantly more accurate reconstruction; see Figure 2.

We begin by reviewing related work (Section 2) and the relevant volumetric rendering theory (Section 3). Our differential ratio tracking strategy for low-variance gradient estimation is derived in Section 4, followed by the bootstrapping of our optimization with



(a) Delta Tracking (b) Ours (constant init.) (c) Ours (emissive init.)

Fig. 2. Optimizing a scattering volume to reproduce an object from 64 reference images. The false-color visualization indicates the absolute difference of the final result to the reference image. (a) Using the same free-flight sampling technique to estimate both the primal image and the gradients results in biased and high-variance gradients (Section 4.2), hindering convergence in empty or low-density regions. (b) Using our novel differential ratio tracking technique (Section 4.3) significantly reduces gradient variance, leading to a better reconstruction. (c) We further propose to initialize the scattering volume optimization from the result of a nonphysical emissive volume optimization in the style of NeRF [Mildenhall et al. 2020] (Section 5). This helps overcome local minima, greatly improving sharpness of the final model.

an initial simplified model in Section 5. Finally, we conclude with an evaluation and discussion of our combined method (Section 6). Our implementation is made available at the following URL:

<https://rgl.epfl.ch/publications/NimierDavid2022Unbiased>

2 RELATED WORK

Existing work on differentiable rendering covers a large range of capabilities and admissible representations, spanning from ray-marched, purely emissive and absorbing volumes [Mildenhall et al. 2020; Yu et al. 2021a] over directly lit, rasterized objects [Hasselgren et al. 2021; Kato et al. 2017; Laine et al. 2020; Liu et al. 2019; Loper and Black 2014; Munkberg et al. 2021; Petersen et al. 2019; Rhodin et al. 2015], to globally illuminated physically-based surfaces [Azinović et al. 2019; Bangaru et al. 2020; Li et al. 2018; Zhang et al. 2020], volumes [Che et al. 2020; Gkioulekas et al. 2013; Khungurn et al. 2016; Zhang et al. 2021b], and both [Nimier-David et al. 2020, 2019; Vicini et al. 2021b; Zeltner et al. 2021; Zhang et al. 2019]. Our work belongs to the category of differentiable physically-based volume rendering, where we contribute a novel importance sampling strategy for low-variance unbiased gradient estimation that runs in linear time.

Efficient differentiable global illumination. One major challenge of differentiating a physically based renderer is complex interreflection: in high-albedo materials, such as clouds, the simulated path length can approach dozens to hundreds of vertices, making naïve automatic differentiation of the rendering algorithm impractical. Nimier-David et al. [2020] instead reformulate the derivatives of the rendering equation as an adjoint integral that distributes gradients to scene parameters through a separate differential path tracing

pass, vastly outperforming naïve automatic differentiation in low-order scattering. High-order scattering, however, is expensive due to the $O(n^2)$ cost of n -vertex paths; this has been reduced to $O(n)$ computation and $O(1)$ memory [Vicini et al. 2021b] by exploiting arithmetic invertibility. Vicini et al. [2021b] also demonstrated the feasibility of differentiable volume rendering using delta tracking, as opposed to ray marching.

One notable difference when solving an adjoint integral is that the set of Monte Carlo sampling strategies used for primal rendering can be replaced by specialized methods targeting the derivative. Zeltner et al. [2021] observe that reuse of primal strategies is often suboptimal for gradient estimation and that tailored differential strategies should be used. Our work points out volumetric free-flight sampling as such a case. We propose augmenting free-flight sampling with our differential ratio tracking algorithm, offering unbiased and low-variance estimates of the adjoint integral. Special care is taken to retain compatibility with the method of Vicini et al. [2021b], preserving the linear time and constant memory complexity. Our method can reconstruct highly scattering, physically based volumes.

Tregan et al. [2020] identify a related problem in the differentiation of radiative transport integrals: a singularity in the gradient of *null collisions*. We find that this singularity can be sidestepped by selecting a large enough majorant, or is solved naturally in the context of our method (Appendix A). We focus on the singularity in the gradients of *real collisions*, which is not addressed by Tregan et al.

Zhang et al. [2021b] derive a path-space formulation of the combined derivatives of surface and volume light transport, including new boundary gradients samplers. These samplers are orthogonal to ours: for volumetric gradients, Zhang et al. utilize standard forward-rendering sampling, which we propose to augment.

Inverse volume rendering. In early work, Klehm et al. [2014] optimize a linearized representation of a single-scattering and emissive volume to match artist input. Khungurn et al. [2016] fit the non spatially-varying parameters of a volumetric cloth appearance model from measurements using differentiable rendering, while Zhao et al. [2016] optimize single-scattering albedos and phase functions. However, neither optimize for the extinction coefficient σ_t , which is the main focus of our work. Gkioulekas et al. [2016] present a general inverse rendering framework for non-emissive heterogeneous volumessupporting a broader type of measurements such as transient imaging. Their inverse rendering algorithm, based on the *score function*, is susceptible to the same gradient bias and variance issues addressed by our method.

Emissive and absorbing volumes. Mildenhall et al. [2020] and follow-ups [Barron et al. 2021a,b; Dellaert and Yen-Chen 2021; Liu et al. 2020; Mildenhall et al. 2021; Müller et al. 2022] demonstrate that the appearance of objects can be faithfully reproduced by an absorbing volume with directionally varying emission profile and no internal scattering whatsoever. With *neural radiance fields* (NeRF), they parameterize the 3D density (absorption coefficient) as well as the 5D spatio-directional emission using fully connected neural networks. However, neural networks appear to not be necessary for high-quality results [Garbin et al. 2021; Sun et al. 2021a; Yu et al. 2021a,b]. Rather, the parameterization in terms of absorption and emission, coupled with a differentiable rendering algorithm, seems

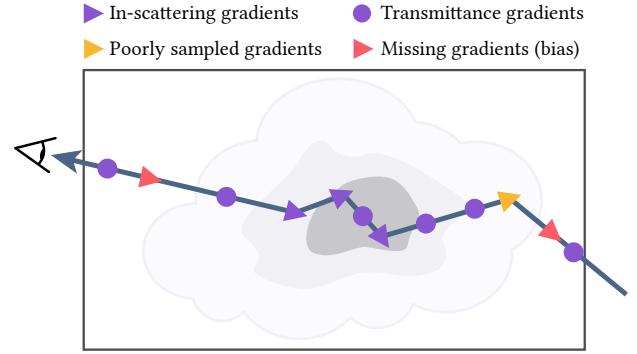


Fig. 3. Forward rendering of volumes involves sampling free-flight distances t proportional to the density-weighted transmittance $\sigma_t(t) T(t)$. Directly applying forward sampling techniques in the context of gradient estimation (the adjoint) leads to biased and high-variance gradients. In particular, gradients due to in-scattering are poorly sampled where density is low and never sampled where it is zero. We propose a tailored importance sampling scheme for the adjoint, which resolves both bias and variance concerns.

to be key. Like prior work, we observe that—unlike scattering volumes which are prone to unfavorable local minima—purely emissive and absorbing volumes reliably converge to plausible 3D densities. Thus, in addition to reducing gradient variance, we first reconstruct one such emissive and absorbing volume, then use its density to bootstrap the reconstruction of a relightable, scattering volume. For simplicity we represent all volumetric parameters as scalars on a 3D grid; our learned emission is isotropic. In X-ray tomography Geva et al. [2018] similarly initialize their optimization with the result of an absorption-only reconstruction, then account for scattering.

Relighting. Often, the goal of inverse rendering is not merely to render novel viewpoints of reconstructed objects, but also to place them under novel lighting conditions. To this end, two general strategies emerged: first, the appearance under different lighting conditions can be *learned* from data, e.g. in terms of abstract latent codes [Martin-Brualla et al. 2021] or radiative transfer functions [Zheng et al. 2021]. And second, a shading model—with direct lighting [Bi et al. 2020; Boss et al. 2021; Hasselgren et al. 2021; Munkberg et al. 2021; Srinivasan et al. 2021] or global illumination [Azinović et al. 2019; Gkioulekas et al. 2013; Khungurn et al. 2016; Nimier-David et al. 2019; Zhang et al. 2021b]—can be postulated, under which the inverse rendering process reconstructs shading parameters, such as the volumetric scattering coefficient, rather than directly solving for appearance. The latter kind tends towards (much) more difficult reconstruction, however in turn permitting more general relighting. Our method, belonging to this category, aims to alleviate the difficult reconstruction by bootstrapping from a simpler, non-relightable reconstruction and, subsequently, estimating low-variance scattering parameter gradients.

3 PRELIMINARIES

We review the most relevant aspects of volumetric path tracing (forward rendering) and the radiative backpropagation formulation for gradient estimation (adjoint rendering).

3.1 Volumetric Path Tracing

We consider a heterogeneous medium with small, independently distributed, scattering and absorbing particles. Such a medium is described by its spatially varying *absorption* $\sigma_a(\mathbf{x})$ and *scattering* coefficients $\sigma_s(\mathbf{x})$ which are proportional to the density of particles along a ray (units of m^{-1}). The *extinction* coefficient $\sigma_t := \sigma_a + \sigma_s$ is defined as the sum of absorption and scattering.

When referring to a position $\mathbf{x}_t = \mathbf{x}_0 + t \cdot \boldsymbol{\omega}$ along a ray $(\mathbf{x}_0, \boldsymbol{\omega})$, we will use t and \mathbf{x}_t interchangeably to avoid cluttering formulae. For example, the *transmittance* along the ray is defined as

$$T(a, b) := \exp\left(-\int_a^b \sigma_t(s) ds\right), \quad (1)$$

where $\sigma_t(s) \equiv \sigma_t(\mathbf{x}_0 + s \cdot \boldsymbol{\omega})$. We use the shorthand $T(t) \equiv T(0, t) \equiv T(\mathbf{x}_0, \mathbf{x}_t)$. Volumetric light transport is governed by the volume rendering equation [Kajiya and Von Herzen 1984]

$$L_i(\mathbf{x}, \boldsymbol{\omega}) = \int_0^{t_s} T(t) [\sigma_a(t) L_e(t) + \sigma_s(t) L_s(t, \boldsymbol{\omega})] dt + T(t_s) [L_e(t_s) + L_s(t_s, \boldsymbol{\omega})], \quad (2)$$

where $L_i(\mathbf{x}, \boldsymbol{\omega})$ is the radiance in direction $\boldsymbol{\omega}$ at position \mathbf{x} and L_e denotes emission from the medium, a surface, or from the environment. t_s corresponds to the distance along the ray to the closest surface or medium boundary. L_s is the in-scattered radiance on volumes or surfaces:

$$L_s(\mathbf{x}, \boldsymbol{\omega}) := \int_{S^2} L_i(\mathbf{x}, \boldsymbol{\omega}') f_s(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}') d\boldsymbol{\omega}', \quad (3)$$

where f_s denotes either the phase function or the bidirectional scattering distribution function (BSDF) depending on whether \mathbf{x} lies on a surface or in a medium. We parametrize media by $\boldsymbol{\theta} := (\sigma_t, \alpha)$, where the *scattering albedo* $\alpha := \sigma_s/\sigma_t$ captures the probability of scattering (as opposed to absorption) at an interaction with the medium.

Free-flight distance sampling. Estimating Equation (2) (as well as Equation (23) of Appendix A) with Monte Carlo integration involves efficiently sampling a so-called *free-flight* distance t , which corresponds to the distance to the next light-particle interaction along the ray. This distance is distributed proportional to $\sigma_t(t) T(t)$, which can be analytically sampled in homogeneous volumes. In heterogeneous volumes, however, a more advanced algorithm must be used. *Delta tracking* [Butcher and Messel 1958], also known as Woodcock tracking [Woodcock et al. 1965], *homogenizes* the medium by introducing fictitious matter such that the extinction coefficient of the volume is equal to the *majorant* $\bar{\sigma}$ everywhere. The algorithm then analytically samples a free-flight distance proportional to $\bar{\sigma} \bar{T}(t)$, where $\bar{T}(t)$ is the homogenized transmittance. Finally, the ray is stochastically determined to have encountered a *real* particle with probability $\sigma_t(t)/\bar{\sigma}$, or a fictitious one otherwise. The algorithm repeats until a real interaction has been found; the resulting distance t has been shown to be distributed proportional to $\sigma_t(t) T(t)$ as desired [Woodcock et al. 1965]. The choice of majorant $\bar{\sigma}$ affects the efficiency: the more closely it bounds σ_t , the smaller the number of fictitious interactions that are processed.

Transmittance estimation. A related operation in volumetric light transport is estimating the transmittance $T(t)$ along a ray segment: the fraction of light traveling a distance t without being absorbed or scattered away. Transmittance can be estimated in a number of ways, such as by marching along the ray in fixed intervals and correcting for bias [Kettunen et al. 2021], and, most relevant for us, by free-flight sampling and thus delta tracking. Delta tracking can be thought of as a *binary* estimate of $T(t)$. The estimate is one (all light is transmitted) if a sampled free-flight distance is larger than t and zero (no light is transmitted) otherwise. This interpretation can be generalized to lower-variance, non-binary estimates of the transmittance by keeping track of the proportion of real vs. fictitious matter $\sigma_t(t)/\bar{\sigma}$ at each step; this is called *ratio tracking* [Novák et al. 2014]. Our method combines ratio tracking and reservoir sampling to sample proportional to $T(t)$ only rather than $\sigma_t(t) T(t)$. We will show this density to be desirable for unbiased, low-variance gradient estimation in (near-)empty regions of space; illustrated in Figure 3.

3.2 Adjoint Radiative Transport

While our method does not depend on either radiative- [Nimier-David et al. 2020] or path replay backpropagation [Vicini et al. 2021b], these methods make it possible to efficiently estimate gradients for millions of parameters in parallel, including in the presence of high-order scattering. It is thus desirable to ensure our proposed algorithm is compatible with them. We review the relevant aspects here, and refer the reader to the original papers for full detail.

Radiative backpropagation. Radiative backpropagation recasts the estimation of gradients with respect to scene parameters as a modified transport problem. For example, in the context of image-based optimization, the gradient of an objective function w.r.t. image pixel values is transported from the camera and into the scene. Paths interacting with scene objects containing differentiable parameters “deposit” the transported gradients as an adjoint quantity that is accumulated in the parameters’ gradients. The accumulated quantity is weighted by the incident radiance L_i at that point.

Radiative backpropagation admits efficient CPU and GPU implementations and can apply separate specialized sampling strategies to the *primal*- (forward rendering of the scene) and *adjoint* problems (estimating gradients w.r.t. scene parameters).

Path replay backpropagation. As originally presented, the unbiased variant of radiative backpropagation requires starting recursive paths at each interaction with differentiable objects in order to obtain an estimate of the incident radiance L_i . This raises the worst-case complexity to $O(n^2)$, with n being the number of path vertices. Path replay backpropagation [Vicini et al. 2021b] leverages the deterministic nature of pseudo-random number generators to bring complexity back to linear time and constant memory. The algorithm consists of three steps:

- I A forward (primal) rendering of the current state of the scene is performed and used to evaluate the objective function.

- II A second, uncorrelated¹ primal rendering pass estimates and stores the radiance L_i^{II} for each path.
- III An adjoint rendering pass *replays* the paths traced in step (II) using the same random number generator seed. The stored incident radiance values L_i^{II} are used to weight gradients, without the need to trace a recursive path.

In the context of this paper, it is important to note that the steps (II) and (III) must use the same paths, and hence the same sampling methods to construct them. We must take care to make any new sampling technique compatible with this constraint in order to preserve the linear runtime (Section 4.7).

4 UNBIASED INVERSE VOLUMETRIC PATH TRACING

We now turn to the problem of efficiently estimating gradients with respect to scene parameters in the presence of heterogeneous media.

4.1 Differentiating the Radiative Transfer Equation

Reconstructing media that are both reflective and emissive introduces ambiguities that are challenging to resolve using only image-based observations. Since the appearance of real-world objects is usually determined by their scattering properties rather than internal emission, we restrict our goal to reconstructing non-emissive volumes that obtain their color from scattering and absorption; we set $L_e = 0$ within the medium. Using the $\theta = (\sigma_t, \alpha)$ parametrization of the medium, the radiative transfer equation (2) then simplifies to:

$$L_i(\mathbf{x}, \omega) = \int_0^{t_s} T(t) \sigma_t(t) \alpha(t) L_s(t, \omega) dt + T(t_s) [L_e(t_s) + L_s(t_s, \omega)], \quad (4)$$

where t_s denotes the position of the nearest surface (or volume bounding box) in the direction of the ray.

For a given ray traversing the medium, the radiance derivatives w.r.t. volume parameters θ are as follows. We separate and rearrange the terms to explain their individual roles, as well as to map more directly to the sampling algorithms. We omit the dependency on ω for conciseness. The first term captures how the *in-scattered* radiance can increase due to a local density (σ_t) or albedo (α) increase:

$$\partial_\theta L_i(\mathbf{x}) = \int_0^{t_s} T(t) \partial_\theta [\sigma_t(t) \alpha(t)] L_s(t) dt \cdots \quad (5)$$

The second term describes how a density increase at an earlier position t' will attenuate the local contribution, whether from a medium interaction (first row) or a surface (second row):

$$\cdots + \int_0^{t_s} T(t) \sigma_t(t) \alpha(t) \left[\int_0^t -\partial_\theta \sigma_t(t') dt' \right] L_s(t) dt + T(t_s) \left[\int_0^{t_s} -\partial_\theta \sigma_t(t') dt' \right] [L_e(t_s) + L_s(t_s)] \cdots \quad (6)$$

¹Gkioulekas et al. [2013], Equation (22), show that an uncorrelated set of paths must be used to estimate the image loss and scene parameter gradients to obtain an unbiased estimate of the overall gradients.

Finally, the third term captures changes in incident radiance after a scattering event at t in the medium or at t_s on a surface. Changes later along the path are weighted by the transmittance of the current segment and the scattering coefficient of the current event:

$$\cdots + \int_0^{t_s} T(t) \sigma_t(t) \alpha(t) \partial_\theta L_s(t) dt + T(t_s) [\partial_\theta L_e(t_s) + \partial_\theta L_s(t_s)]. \quad (7)$$

4.2 Free-Flight Based Gradient Estimators

Standard free-flight distance sampling proportional to $\sigma_t(t) T(t)$, such as by delta tracking, is well suited for the primal problem of Equation (4): the probability density function cancels most factors. However, using free-flight sampling to estimate the adjoint is problematic, as we will show below. Note its usage can be hidden, e.g. when applying automatic differentiation to a volumetric path tracer.

Using free-flight importance sampling to sample the adjoint integrals (5)–(7) yields the following estimators. A valid free-flight distance $t < t_s$ is sampled within the medium with probability $\sigma_t(t) T(t)$, resulting in a sample with contribution

$$\langle \partial L_1^{\text{DT}} \rangle = \frac{\partial_\theta [\sigma_t(t) \alpha(t)]}{\sigma_t(t)} L_s(t) + \alpha(t) \partial_\theta L_s(t), \quad (8)$$

Otherwise, if $t \geq t_s$, the surface at t_s is sampled with probability $T(t_s)$, contributing:

$$\langle \partial L_2^{\text{DT}} \rangle = \partial_\theta L_e(t_s) + \partial_\theta L_s(t_s). \quad (9)$$

In either case, transmittance gradients are accumulated at locations t' along the ray segment $(0, \min(t, t_s))$ corresponding to the location of null-scattering interactions:

$$\langle \partial L_3^{\text{DT}} \rangle = \begin{cases} -\partial_\theta \sigma_t(t') \alpha(t) L_s(t) & \text{if } t < t_s, \\ -\partial_\theta \sigma_t(t') [L_e(t_s) + L_s(t_s)] & \text{otherwise.} \end{cases} \quad (10)$$

We will refer to these estimators as “free-flight” as a shorthand for free-flight sampling-based gradient estimators. There are two issues with estimator $\langle \partial L_1^{\text{DT}} \rangle$.

Bias. If the medium density $\sigma_t(t)$ is zero, estimator $\langle \partial L_1^{\text{DT}} \rangle$ is never sampled even though the integrand of Equation (5), responsible for in-scattering gradients, is not zero. This results in bias. Figure 4 illustrates this bias in a specially constructed worst-case scenario, with $\sigma_t(t)$ initialized to zero. Since the background is darker than the target image, transmittance gradients point in the wrong direction: brightening the image would require increasing transmittance by further lowering σ_t , which is not possible. Because in-scattering gradients are missing, the optimization cannot escape this initial state. Interestingly, an analogous situation arises for surface BSDF optimization when the BSDF is exactly zero. In that case however, it is legitimate to clamp BSDFs to a small nonzero value, as truly black surfaces do not occur in normal circumstances.

Variance. Furthermore, if the density $\sigma_t(t)$ is *close to* zero, the factor $1/\sigma_t(t)$ in Equation (8) leads to large outliers in gradients w.r.t. density σ_t . Because distances are sampled proportional to extinction-weighted transmittance, it may appear that the low probability of sampling t where $\sigma_t(t)$ is small would compensate for

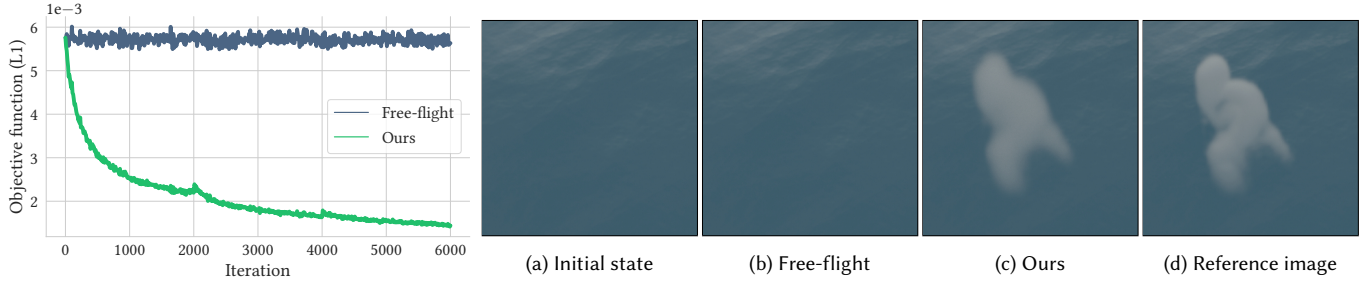


Fig. 4. In this illustrative example, the medium density σ_t is optimized to match 32 reference images of a cloud **(d)**. The initial state **(a)** is set to $\sigma_t = 0$. This represents a worst-case scenario for the free-flight based gradient estimator **(b)**: its bias prevents it from escaping the initial state. Indeed, in-scattering gradients are missing where $\sigma_t = 0$ and transmittance gradients point in the direction opposite to the solution (*i.e.* further decreasing σ_t to increase brightness, which is impossible). Our method correctly estimates in-scattering gradients even in empty space, which allows it to converge to a good solution **(c)**.

the additional variance. However, Tregan et al. [2020] have shown (for the case of a homogeneous slab) that the resulting variance is actually unbounded². Note that this issue occurs as well when using the null-scattering integral formulation, see Appendix A. Gradients w.r.t. albedo α , however, are not affected.

The effect of these gradient outliers is illustrated on an inverse rendering problem in Figure 1. Reconstruction becomes impossible using standard stochastic gradient descent. Optimizers that track gradient variance via second moments, such as Adam [Kingma and Ba 2014], will at this point aggressively reduce the step size, enabling acceptable convergence. Reconstructions, however, remain inferior to that obtainable with the unbiased gradient estimators presented in this article (Section 6.3).

Defensive sampling. A seemingly obvious solution to the problem we have just identified would be to modify the sampling strategy to include a constant “background density” ϵ :

$$p_{\text{def}}(t) = (\sigma_t(t) + \epsilon) T_\epsilon(t), \quad (11)$$

which can be interpreted as a form of *defensive sampling*. If ϵ is sufficiently large, the problematic term in the adjoint would no longer produce gradient outliers:

$$\frac{1}{\sigma_t(t) + \epsilon} \leq \frac{1}{\epsilon}. \quad (12)$$

While this may seem like a clear advantage, we must also consider the effect of the modified density on all other terms of the adjoint. Importantly, the unmodified density allowed path throughput to stay close to one in Equations (8-9). With p_{def} , non-unit sampling weights compound over the length of the path. In a volumetric rendering context where long paths are common, this can become problematic.

Moreover, recall that when using path replay backpropagation (Section 3.2), the same paths must be constructed in phases (II) and (III). This means that either the suboptimal defensive sampling strategy must also be used to estimate the incident radiance terms in the adjoint, injecting additional variance into the gradient estimates, or defensive samples must be drawn *in addition* to the paths constructed in phases (II) and (III), increasing cost. In the following, we

²More precisely, the argument of Tregan et al. concerns variance due to the reciprocal event of null scattering where $\sigma_n \approx \bar{\sigma}$, which is analogous to $\sigma_t \approx 0$. See Appendix A.

will take the latter approach—drawing additional samples—however using a sampling technique tailored to the integrand of Equation (5).

Alternative strategies. Adopting a ray marching-based estimator would be susceptible to the same issues, even when ignoring the bias due to the discretization of the transmittance function. Indeed, the underlying density being sampled, $\sigma_t(t) T(t)$, is identical to the delta tracking-based estimator discussed above. Likewise, we did not find reparameterizing the volume density, *e.g.* $\sigma_t(t) = \log(\theta(t))$, to be a suitable solution—near-empty regions retain high-variance gradients, and are moreover susceptible to vanishing gradients.

4.3 Unbiased Estimators

We have identified the $1/\sigma_t$ term of Equation (8) to be the source of bias and variance. Fundamentally, the issue stems from the direct application of a primal sampling technique to the adjoint problem, where the integrand is different. This same problem was identified in other contexts by Zeltner et al. [2021]. We propose replacing the problematic $\langle \partial L_1^{\text{DT}} \rangle$ estimator with a simple, tailored estimator that samples distances proportional to transmittance only.

The need to adopt a specialized sampling approach makes intuitive sense: regions of the medium with $\sigma_t \approx 0$ are only sampled rarely with primal sampling strategies (or not at all, if $\sigma_t = 0$). At the same time, if transmittance is close to 1, then a small change of σ_t can have a significant impact on the solution of Equation (5). These regions therefore generate gradients during the adjoint transport step and must receive sufficiently many samples to produce low-variance gradient estimates. Sampling proportional to transmittance addresses both the bias and variance issues discovered in Section 4.2.

Assuming for a moment that we are able to sample $t' \sim T(t')$, we replace the two summands in $\langle \partial L_1^{\text{DT}} \rangle$ (Equation (8)) by the following two respective estimators:

$$\langle \partial L_{1a}^{\text{DRT}} \rangle := L_s(t') \partial_\theta [\sigma_t(t') \alpha(t')], \quad (13)$$

$$\langle \partial L_{1b}^{\text{DRT}} \rangle := \alpha \partial_\theta L_s(t). \quad (14)$$

Here, t' is used *only* to sample in-scattering gradients $\langle \partial L_{1a}^{\text{DRT}} \rangle$ where the original strategy was problematic. For all other terms, $\sigma_t T$ is a better suited density due to cancelling more factors, therefore

we keep $t \sim \sigma_t T$ in $\langle \partial L_{1b}^{\text{DRT}} \rangle$ and set $\langle \partial L_2^{\text{DRT}} \rangle := \langle \partial L_2^{\text{DT}} \rangle$ as well as $\langle \partial L_3^{\text{DRT}} \rangle := \langle \partial L_3^{\text{DT}} \rangle$ to estimate the remaining terms.

Readers familiar with path replay backpropagation may notice a potential drawback in Equation (13): because t' is sampled from a different strategy than the one used to construct the path, the path replay algorithm cannot efficiently produce an estimate of incident illumination L_i at t' . In the worst case, we may have to start a recursive path at each sampled point t' , raising the time complexity to $O(n^2)$. We will return to this issue in Section 4.7.

4.4 Sampling Proportional to Transmittance

We now turn to the derivation of our transmittance importance sampling technique. To this end, we build over the family of delta- [Butcher and Messel 1958; Woodcock et al. 1965] and ratio trackers [Novák et al. 2014].

Target distribution. Let us write down the density $p_T(t)$ of this hypothetical transmittance-based sampling strategy. Because T is not a normalized density on its own, we must apply a normalization constant defined in terms of an integral:

$$p_T(t) = \frac{T(t)}{\int_0^{t_s} T(t') dt'} = \frac{T(t)}{C}. \quad (15)$$

The unbiased estimation of such an integral in the reciprocal presents a challenge. Moreover, T is itself defined in terms of an integral. These circumstances prevent the use of standard methods like inverse transform sampling.

Note that there is an implicit assumption that C remains finite, which requires that the transmittance decays sufficiently quickly. In practice, the scene would likely contain only media within limited spatial extent, in which case sampling can be restricted to this volume to avoid this technicality. We have accounted for this assumption by setting the integral's upper limit to t_s , the location of the closest surface or medium bounding box along the ray.

It would seem that the presence of the normalizing constant C forces us to revise the estimator of Equation (13) to

$$\langle \partial L_{1a}^{\text{DRT}} \rangle = C L_s(t') \partial_\theta [\sigma_t(t') \alpha(t')]. \quad (16)$$

However, instead of evaluating Equation (16) using unweighted samples that are ideally proportional to the re-normalized transmittance, we pursue a strategy that generates *weighted* samples. In other words, the samples alone will not have the correct distribution, but they will be distributed according to p_T when considered along with their weights. These weights will also serve a secondary purpose: to provide an unbiased estimate of the normalization constant C . Serendipitously, all of this is possible with a simple extension to the family of delta- and ratio trackers.

Delta tracking. We start with a simple strategy based on the delta tracking free-flight distance sampling method. On a given ray segment $[0, t_s]$, delta tracking repeatedly samples steps t_0 from a homogenized medium with density $\bar{\sigma} \geq \sigma_t(t) \forall t$. At each step, the nature of the interaction is determined stochastically: a *real* particle is encountered with probability $\sigma_t(t)/\bar{\sigma}$. Otherwise, a *null* particle is encountered and the traversal continues.

If a real interaction occurs at $t_1 \in [0, t_s]$, light is scattered or absorbed and the ray throughput falls to zero. The free-flight distance

t_1 is sampled with probability $p_1(t) = \sigma_t(t) T(t)$. We can interpret this randomized process as creating the following binary function or “tracking”:

$$f_{\text{DT}}(t) = \begin{cases} 1 & \text{if } 0 < t \leq \min(t_1, t_s), \\ 0 & \text{otherwise.} \end{cases}$$

An example tracking is drawn in Figure 6 (left, blue curve). By averaging together many such trackings, we obtain an unbiased estimate of transmittance at all points.

Sampling from a medium realization. We could interpret each sampled tracking f_{DT} as the *actual* transmittance function of a particular realization of the stochastic medium. We propose sampling proportional to that transmittance function, and show that the resulting density exactly corresponds to our target.

Given $t_1 \sim p_1$ sampled by delta tracking, we simply sample a new distance $t_2 := t_1 s$, with $s \sim p_s(s) := \mathbb{1}_{[0,1]}(s)$. This step is illustrated in Figure 6 (left, green arrow). Let us examine the corresponding product density

$$\begin{aligned} \tilde{p}_2(t) &= \int_{-\infty}^{+\infty} p_1\left(\frac{t}{s}\right) p_s(s) \frac{1}{|s|} ds \\ &= \int_0^1 p_1\left(\frac{t}{s}\right) \frac{1}{s} ds \\ &\stackrel{(1)}{=} \int_t^{+\infty} p_1(q) \frac{1}{q} dq, \end{aligned}$$

where step (1) involves the change of integration variable $q := t/s$. Unfortunately, \tilde{p}_2 does not quite correspond to our desired density. We correct it by associating a weight $w := q = t/s$ to each sample t . The weighted density is

$$\begin{aligned} p_2(t) &= \int_{-\infty}^{+\infty} p_1\left(\frac{t}{s}\right) p_s(s) \frac{t}{s} \frac{1}{|s|} ds \\ &\stackrel{(1)}{=} \int_t^{+\infty} p_1(q) dq \\ &= \int_t^{+\infty} \sigma_t(q) T(0, q) dq \\ &= T(0, t) \int_t^{+\infty} \sigma_t(q) T(t, q) dq, \end{aligned} \quad (17)$$

where (1) denotes the same change of variable as before. Georgiev et al. [2019, Equation (9)] have shown that transmittance can be expressed as the following Volterra integral equation:

$$T(a, b) = 1 - \int_a^b \sigma_t(s) T(s, b) ds.$$

Substituting into Equation (17) yields

$$\begin{aligned} p_2(t) &= T(0, t) (1 - T(t, +\infty)) \\ &= T(0, t) - T(0, +\infty). \end{aligned}$$

Finally, using our earlier assumption that the normalization constant C is finite (in practice, equivalent to setting $\sigma_t = +\infty$ beyond the volume's bounding box), we have $T(0, +\infty) = 0$ and obtain

$$p_2(t) = T(0, t).$$

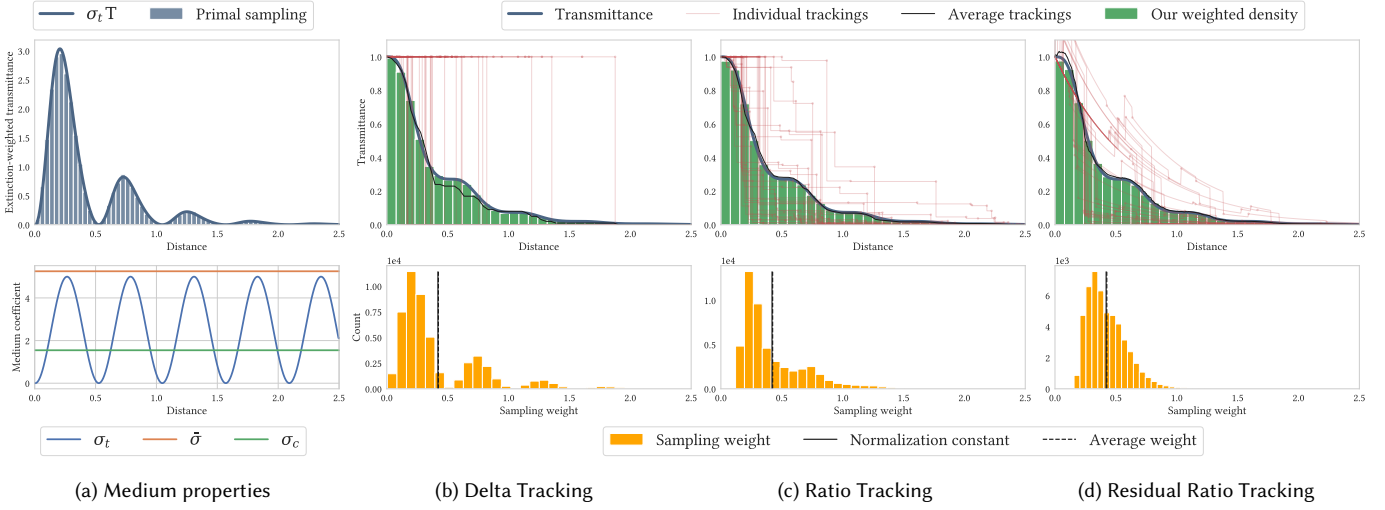


Fig. 5. Demonstrating our novel transmittance sampling methods based on delta- and (residual) ratio tracking. (a, bottom) We use a 1D analytic medium with density following a sinusoidal function. The chosen majorant and control densities are also shown. (a, top) Forward rendering of such a medium involves sampling free-flight distances proportional to $\sigma_t(t) T(t)$. Locations where $\sigma_t \approx 0$ receive few or no samples, which leads to bias and high variance when using this same sampling technique to estimate gradients. Instead, we propose dedicated transmittance sampling techniques for the adjoint. Delta tracking (b), ratio tracking (c) and residual ratio tracking (d) can be interpreted as building estimates of the transmittance function using binary, piecewise constant and piecewise-exponential functions respectively. We build transmittance sampling algorithms from those trackers by sampling distances from their individual function approximations. They produce weighted samples (green histogram) with density equal to the transmittance function T , where the weights (yellow histogram) approximate the normalization constant of T .

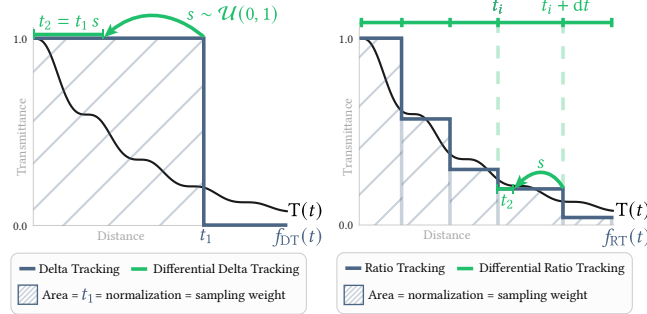


Fig. 6. Our proposed sampling methods build on the family of delta- (left) and ratio (right) trackers. The trackers build piecewise approximations f of the true transmittance function T . We simply sample a distance proportional to the area under f which, together with a sampling weight equal to the total area under the curve, yields unbiased samples of T .

We conclude that the weighted samples ($t_2, w = t_2/s$) have the desired distribution and call this sampling technique *differential delta tracking*. In Figure 5b, we confirm experimentally that the weighted density matches the transmittance of an example 1D medium slice.

Normalization constant. Further inspection reveals that the chosen sampling weight is quite meaningful: $t_2/s = t_1$ is exactly the area under the curve of the binary function f_{DT} . In other words, it captures the difference between our unnormalized target density T and the normalized density we are actually sampling from. Moreover, by construction of delta tracking, the trackings approximate

the transmittance function [Novák et al. 2014]

$$\mathbb{E} \left[\int_0^{t_s} f_{DT}(s) ds \right] = \mathbb{E}[t_1] = \int_0^{t_s} T(s) ds = C. \quad (18)$$

The sampling weight w therefore takes on the role of C in Equation (16), completing the estimator.

4.5 Differential Ratio Tracking

Using the same intuition, we can build a more efficient transmittance sampling technique based on *ratio tracking* [Novák et al. 2014], which is a simple and effective strategy that expands on delta tracking to compute unbiased estimates of the transmittance along a ray segment $(0, t_s)$. As before, successive steps t_i are sampled in closed form from a homogenized medium with density $\bar{\sigma} \geq \sigma_t(x) \forall x$. However, rather than sampling a binary real/null decision at each step, the probability of interacting with a real particle is used to update the estimated transmittance up to the current point. This can be interpreted as building a piecewise constant, rather than binary, tracking. It approximates the transmittance function more closely:

$$f_{RT}(t) = \prod_{t_i \leq \min(t, t_s)} 1 - \frac{\sigma_t(t_i)}{\bar{\sigma}},$$

where $\sigma_t(t_i)/\bar{\sigma}$ is the probability of interacting with a real particle. An example piecewise constant tracking f_{RT} is drawn in Figure 6 (right, blue curve).

Sampling from ratio trackings. Following the same intuition as in Section 4.4, we derive a transmittance sampling technique from the ratio tracking algorithm. Where differential *delta* tracking simply

needed to sample from the binary function f_{DT} , we now need to sample from the piecewise constant f_{RT} .

Given a realization f_{RT} , it is easy to build and sample from a discrete 1D distribution based on the area under each constant segment. Once a segment has been selected, we simply sample uniformly along it. This step is illustrated in Figure 6 (right, green arrow). As before, we associate a sampling weight w equal to the area under the curve of f_{RT} :

$$w = \sum_{t_i} (t_{i+1} - t_i) f_{RT}(t_i), \quad (19)$$

$$\text{with } \mathbb{E}[w] = \mathbb{E}\left[\int_0^{t_s} f_{RT}(s) ds\right] = \int_0^{t_s} T(s) ds = C. \quad (20)$$

Due to the individual trackings' closer match to the transmittance curve, the sampling weights also converge faster to C .

Online segment sampling. The sampling method described above involves fully constructing a tracking f_{RT} before sampling proportional to its area. While this is certainly feasible, the storage and computational overhead would be unnecessarily high, especially considering that the number of segments is not known ahead of time and grows with the majorant $\bar{\sigma}$ and total distance t_s .

Instead, we turn to *weighted reservoir sampling* [Chao 1982] in order to sample a segment *online*, simultaneously with ratio tracking. Desirably, reservoir sampling uses $O(1)$ memory and has very limited computational overhead.

At each step $t_i \rightarrow t_{i+1}$ of ratio tracking, we propose the corresponding segment of f_{RT} with weight equal to the area under it:

$$w_i = d_i f_{RT}(t_i), \quad \text{with } d_i = t_{i+1} - t_i.$$

The reservoir is updated to this segment with probability $w_i / \sum_{j=1}^i w_j$. Once ratio tracking terminates by reaching the maximum distance t_s , we read the values (t_r, d_r) from the reservoir. By construction of reservoir sampling, this segment was sampled with probability $d_r f_{RT}(t_r) / \sum_j d_j f_{RT}(t_j)$. Moreover, the overall sample weight $w = \sum_j d_j T(t_j)$ is readily available as it is computed as part of the reservoir sampling algorithm.

We will refer to this estimator as *differential ratio tracking*. Its full pseudocode is given in Listing 1. We also demonstrate it in action on a 1D example in Figure 5c.

Differential residual ratio tracking. Finally, we can build a third sampling technique based on *residual ratio tracking* [Novák et al. 2014]. Residual ratio tracking can be seen as applying ratio tracking to a “residual medium” whose density is equal to the original density minus a control density σ_c . The trackings are therefore further improved from piecewise constant to piecewise-exponential. Some example trackings are shown in Figure 5d.

Likewise, *differential residual ratio tracking* samples proportional to the area under these trackings. In practice, we found that the limited improvement in gradient variance did not warrant the inclusion of an additional hyperparameter (the control density). Our experiments were therefore conducted with differential *ratio* tracking. For completeness, the pseudocode is nevertheless given in Appendix B.

```
def drt_sample(t_max):
    Tr = 1.
    t_run = w_acc = 0.

    while t_run < t_max:
        # Sample from the homogenized medium
        dt = -log(1-rng())/σ
        dt = min(dt, t_max - t_run)
        # Propose current constant segment with weight
        # equal to area under the curve.
        w_step = Tr * dt
        w_acc += w_step
        if (rng() * w_acc) < w_step:
            # Update reservoir (happens at least once)
            reservoir = (t_run, dt)

        # Update transmittance estimate
        Tr *= 1 - (σ_t(t_run) / σ)
        t_run = t_run + dt

    # Final uniform sampling over the chosen segment
    t, dt = reservoir
    t = t + rng() * dt
    assert t ≤ t_max
    return t, w_acc
```

Listing 1. Differential ratio tracking efficiently samples t proportional to transmittance $T(t)$ along the current ray up to t_{\max} by combining ratio tracking and reservoir sampling. The sampling weight is an unbiased estimate of the integral of transmittance $C = \int_0^{t_s} T(s) ds$.

4.6 Multiple Importance Sampling

We now have at our disposal the standard free-flight sampling technique proportional to $\sigma_t T$ as well as our novel technique sampling proportionally to T only. As was mentioned in Section 4.3, the former is well suited to sample all but the in-scattering gradient term.

Our sampling technique is used to estimate in-scattering gradients, with estimator $\langle \partial L_{1a}^{DRT} \rangle$. Since we will construct paths using free-flight sampling in any case (and the incident radiance term L_i at that point is provided by path replay), we may as well combine the corresponding in-scattering gradients $\langle \partial L_1^{DT} \rangle$ with ours using multiple importance sampling [Veach and Guibas 1995].

The multiple importance sampling weights themselves are simple to compute. For example, using the power heuristic:

$$p_{DT}(t) = \sigma_t(t) T(t), \quad p_{DRT}(t) = T(t),$$

$$w_{DT}(t) = \frac{p_{DT}(t)^2}{p_{DT}(t)^2 + p_{DRT}(t)^2} = \frac{\sigma_t(t)^2}{\sigma_t(t)^2 + 1}, \quad (21)$$

$$\text{and } w_{DRT}(t) = \frac{1}{\sigma_t(t)^2 + 1}. \quad (22)$$

Using multiple importance sampling, we improve variance at negligible extra cost³.

4.7 Preserving Linear Time Complexity

Path replay backpropagation (Section 3.2) can estimate derivatives in linear time. It is desirable that our method preserves this property. Unfortunately, if Equation (13) were used as stated, our estimator would require an additional estimate of incident illumination $L_i(t')$.

³Since $p_{DRT}(t)$ is unnormalized, variance reduction is neither guaranteed, nor governed by the bounds proven by Veach and Guibas [1995]. Nonetheless, we observe significant variance reduction in practice; see Figure 9d.

This is because the distance t' sampled by our estimator differs from the distance sampled to build the path; thus, path replay does not provide the needed value $L_i(t')$. Estimating $L_i(t')$ under global illumination requires tracing a recursive path, resulting in an overall $O(n^2)$ cost when repeated at every bounce of the original path.

Instead, to maintain $O(n)$ cost, we select only a *single bounce* of the original path, proportional to the throughput $\text{Tr}(d)$ of the corresponding path segment, at which we evaluate $\langle \partial L_{1a}^{\text{DRT}} \rangle$. To this end, we again use weighted reservoir sampling as the path is traversed. The reservoir is updated proportional to $\text{Tr}(d)$ and records the segment index as well as the necessary state to spawn the recursive path. Due to spawning only a single recursive path, its resulting estimate must be weighted by $w = \sum_{d=0}^n \text{Tr}(d)$ (the unnormalized inverse probability of being selected), where n is the path length. In Figure 8, we empirically validate that—while this optimization increases gradient variance—it remains well below that of delta tracking.

5 COMBATING LOCAL MINIMA

We have derived an estimator tailored to the estimation of gradients with respect to heterogeneous medium parameters, eliminating the bias and variance found in existing approaches. However, experience shows that regardless of the estimator used, inverse rendering of scattering media using gradient based optimization may not always converge to a global minimum. We propose leveraging nonphysical emissive volumes, whose convergence is better behaved, to bootstrap the optimization of our scattering media.

5.1 Source of Local Minima

Given the complexity of the reconstruction problem and the presence of many ambiguities, it is understandable that the loss landscape may include many local minima. For an intuitive example, consider the following situation: at a given stage of the optimization, the objective function indicates that the brightness of a given pixel should be increased to better match the reference image. At each point along the paths contributing to this pixel, increasing local density increases in-scattered radiance at that point, but simultaneously reduces transmittance and therefore attenuates contribution from interactions further down the path. This delicate balance is captured by the terms of opposite signs in Equation (5) and Equation (6).

In Figure 7, we plot the evolution of the objective function as we interpolate from a converged optimization result to the ground truth solution. While this is not an exhaustive exploration of the local neighborhood (impractical due to the millions of medium parameters), it suggests the presence of a local minimum.

5.2 Emissive Volume Conversion

On the other hand, nonphysical emissive volume representations such as the one employed by NeRF [Mildenhall et al. 2020] and subsequent works seem to converge to accurate solutions without fault⁴. In our experience, this remains true when replacing the neural network by a grid of interpolated values (similar observations were

⁴Here, by “accurate” we mean low training error in image space. The correspondence of the reconstructed volume to its true shape (test error) additionally depends on a sufficiently large number of training views to constrain the problem sufficiently.

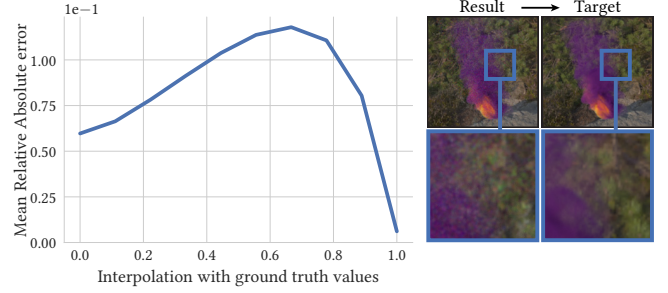


Fig. 7. Regardless of gradient quality, inverse rendering is prone to difficult local minima. We compute the value of the re-rendering loss as we interpolate from the result of an optimization (top left image) to the ground truth value (top right image). While this is not an exhaustive exploration of the loss landscape (the volumetric model comprises 16+ million parameters), it suggests the presence of a local minimum.

made by others [Sun et al. 2021a; Yu et al. 2021a]), and—in our simple scenes—even when omitting directionally-dependent emission.

Relying on this property, we propose the following simple two steps to optimize scattering volume parameters. First, reconstruct a nonphysical emissive volume density σ_t and emission L_e . This is expected to converge fast both in terms of iteration count and runtime: the maximum path depth is one (no scattering) and variance is minimal. Second, use the recovered density σ_t to initialize the optimization of the physical medium parameters σ_t and α . The density σ_t will typically only require small adjustments—in particular when low-order scattering is the dominant source of color—because of the close resemblance between L_s and L_e in the radiative transfer equation (2). The albedo α , on the other hand, is easy to recover because it linearly relates to pixel color; it is not affected by the opposing gradients in Equation (5) and Equation (6) that govern occlusion. We have experimented with different ways of converting the previously optimized emission L_e into initial albedo values to bootstrap the second optimization, but did not find it to perform better than initializing albedo from a constant value, e.g. 0.6. All initializations led to rapid convergence.

We found this two-step optimization to be particularly helpful for inverse rendering of dense objects: see for example the increased sharpness in Figure 2c.

Alternative initialization. One may consider initializing the density σ_t in the second step from a number of other shape estimation techniques such as silhouette carving [Laurentini 1994], multi-view stereo [Jin et al. 2005], or COLMAP [Schönberger and Frahm 2016]. Unfortunately, many of these methods are not designed for participating media or semitransparent objects. Even when reconstructing just solid objects, the precise density values σ_t to extract from hard surfaces is non-obvious. By bootstrapping from an emissive *volume*, despite it being nonphysical, we directly obtain an interpretable σ_t value. Moreover, emissive volume optimization is easily implemented as a special case of scattering volume optimization. Orthogonally, reparametrizations leveraging similarity relations can be exploited to improve convergence [Zhao et al. 2014]. For surface reconstructions, VolSDF [Yariv et al. 2021] could be substituted to

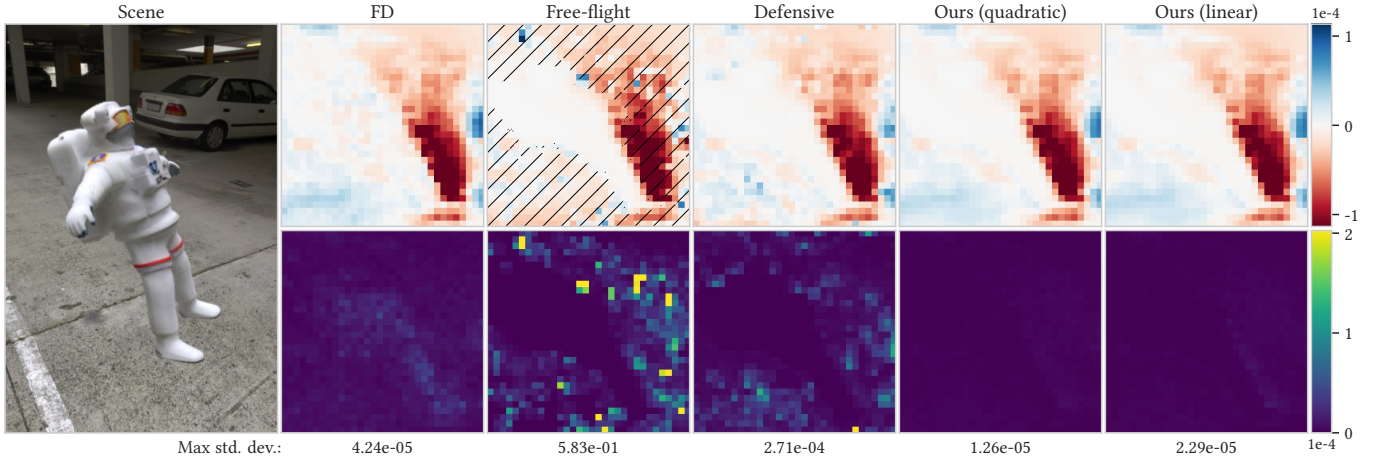


Fig. 8. We validate gradients with respect to medium density σ_t in a synthetic scene (left). The mean gradient values (top row) and standard deviation (bottom row) are shown for the slice $z = 15$ of the dense σ_t 3D parameter grid. Finite differences (FD) provide reference gradient values, but cannot be used in the inner loop of an optimization due to the prohibitive runtime. Using a free-flight sampling based estimator (Section 4.2) to estimate gradients results in bias where $\sigma_t = 0$ (striped area) and high variance where $\sigma_t \approx 0$ (also shown in Figure 1). Defensive sampling (Section 4.2) with $\epsilon = 0.1$ mitigates the largest outliers, but does not eliminate the remaining variance. Our novel sampling technique, whether used at each path segment (“quadratic”, Section 4.3) or once per path (“linear”, Section 4.7) addresses both the bias and variance concerns.

the emissive volume model for a higher fidelity initialization, at the cost of implementation complexity.

6 EVALUATION

We now turn to the evaluation of our proposed gradient estimator and inverse rendering bootstrapping technique. An interactive viewer showcasing inverse rendering results is accessible at:

<https://rgl.epfl.ch/publications/NimierDavid2022Unbiased>

6.1 Correctness and Variance

We first validate the correctness and efficiency of our method before evaluating an inverse rendering application in Section 6.3.

Validation against finite differences. We have established that gradients computed using free-flight sampling-based estimators can suffer from bias and high variance. Figure 8 provides empirical verification: we compare the mean gradients and their standard deviation computed by several methods on the same dense medium, lit by a realistic high dynamic range environment map. Specifically, we compare reference gradients, computed by brute force using finite differences (FD)—hence the low resolution—with free-flight sampling, defensive sampling (Equation (11)), and two configurations of our method: an $O(n^2)$ implementation where our estimator is used at every path segment, as well as our linear time variant.

As expected, gradients estimated using free-flight sampling exhibit bias in empty regions ($\sigma_t = 0$, striped area), as well as high standard deviation, manifesting as large outliers, where density is low. Using defensive sampling helps eliminate the largest outliers, but standard deviation remains high due to the mismatch between the sampling density and integrand. Our estimator yields correct

gradients as well as significantly lower standard deviation in both the $O(n^2)$ and $O(n)$ configurations.

Validation against analytic ground truth gradients in a simplified setting is given in Appendix E.

Ablation study. We study the contribution of each component of our method in Figure 9. The test medium contains empty, thin, and dense regions, and is likewise lit by a realistic outdoor environment map. Given initial medium density ($\sigma_t(x)$, shown in the leftmost column) and albedo parameters, we estimate gradients with respect to $\sigma_t(x)$ using different estimators. Density is represented by a trilinearly interpolated dense grid with resolution $256 \times 128 \times 128$ (more than four million parameters). We then visualize the standard deviation of gradients w.r.t. each σ_t value in a 2D slice of that grid.

We observe a very large standard deviation in regions of low density when estimating gradients with the free-flight sampling-based estimator (a). Note that due to in-scattering gradients being entirely missing in empty regions ($\sigma_t = 0$, striped area denoting bias), variance is artificially low. Our method (b) fully eliminates gradient outliers. In regions where $\sigma_t = 0$, variance is moderately increased due to the (correct) inclusion of the in-scattering gradient term. Reducing the sampling frequency from every path segment to a single segment (c) brings our estimators’ complexity back to linear time (Section 4.7), but also increases variance in regions of low throughput such as the center of the dense smoke plume. However, this is corrected by combining our estimator with regular free-flight sampling via multiple importance sampling (d) (Section 4.6). We find that switching to an estimator based on *residual* ratio tracking (e) rather than ratio tracking does not bring noticeable improvements. Likewise, increasing the medium’s majorant from 1.01 to 10 times the largest σ_t value does not significantly improve gradient variance

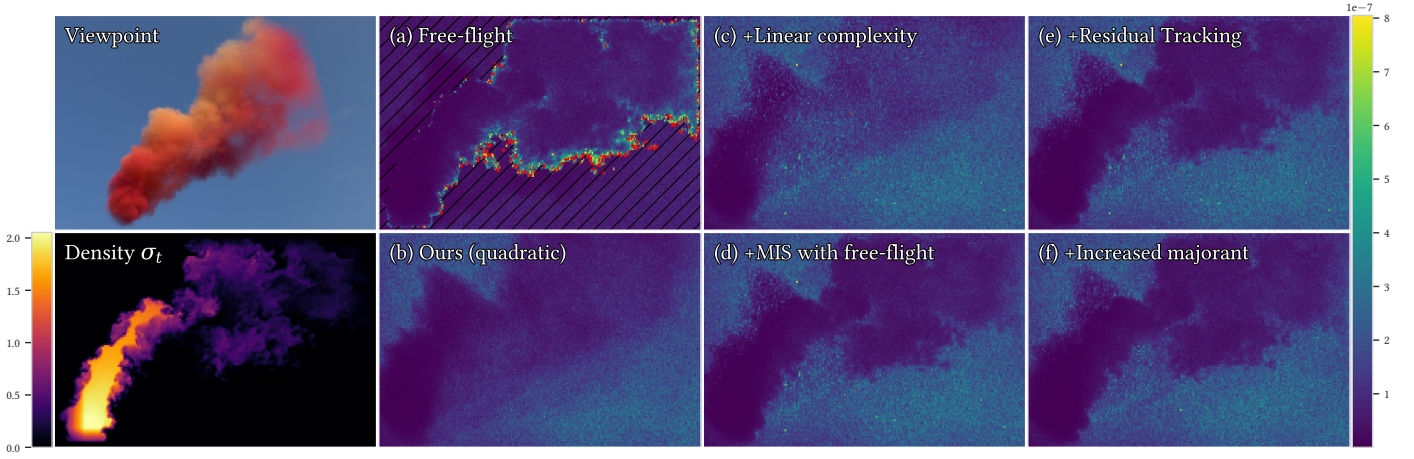


Fig. 9. Ablation study. We compare the standard deviation of gradients with respect to the density σ_t of the medium shown in the top-left rendering. Inverse rendering of this medium is shown in Figure 11). The medium includes regions of high, low and zero density (bottom left). We show slice $z = 64$ of parameter space. (a) As was seen previously (Figure 8), free-flight based gradients are prone to large outliers where $\sigma_t \approx 0$. Values outside of the visualization range are denoted in red. In regions where $\sigma_t = 0$ (striped area), the missing in-scattering gradients (bias) lead to artificially low standard deviation. (b) Our unbiased *differential ratio tracking* estimator resolves both the bias and variance issues when used on every path segment (leading to quadratic complexity). (c) Selectively using our estimator once per path lowers the complexity back to linear time, but introduces additional variance in denser regions where path throughput is lower. However, combining both techniques with multiple importance sampling (d) makes up for it at virtually no cost. Finally, switching to differential *residual* ratio tracking (e) or increasing the medium majorant tenfold (f) does not seem to significantly improve standard deviation.

Table 1. We report the median cost per training iteration (s/it) for vanilla path replay backpropagation [Vicini et al. 2021b] (free-flight) as well as with our proposed $O(n)$ sampling technique enabled. Our extension involves tracing an additional recursive path, which amounts to a $\sim 20\%$ overhead. To ensure the same experimental conditions, both methods compute gradients w.r.t. the same starting medium, of which we report the mean optical depth and maximum density. In practice, different optimization techniques lead to different media, which additionally affects performance.

	Optical depth	Density σ_t	Median runtime (s/it)	
	mean	max	Free-flight	Ours
Dust devil	2.12	45.99	0.20	0.24
Red cow	2.45	12.87	0.32	0.37
Smoke plume	3.24	95.57	0.24	0.29

(f). In order to avoid the additional hyperparameter (medium control density) and additional computation cost respectively, we use differential *ratio* tracking and a majorant of $1.01 \sigma_t$ in the remaining experiments.

In Appendix E, we additionally study gradient variance against the density and ratio of empty space of a test scene.

6.2 Performance

Compared with pure free-flight sampling in the path replay backpropagation framework [Vicini et al. 2021b], our $O(n)$ algorithm requires tracing one additional recursive path for each training path. Table 1 shows that the recursive path amounts to an overhead of $\sim 20\%$ in our implementation, provided that both algorithms compute gradients w.r.t. the same starting medium. In practice, different

optimization techniques lead to different media, which additionally affects performance.

We implemented both methods in Mitsuba 2 [Nimier-David et al. 2019]. The timings of Table 1 were measured on an NVIDIA RTX 3090 GPU. Appendix D contains additional implementation details.

6.3 Inverse Rendering

Given reference images and initial values for the medium properties of interest, we use gradient-based optimization to minimize a re-rendering objective function. Note that our work focuses on the effective estimation of gradients: assembling a fully robust reconstruction pipeline using real-world, imperfect data has its own set of challenges. Therefore, we use synthetic scenes with known camera parameters, illumination conditions, and isotropic phase function. Since we target non-emissive objects, we only optimize for L_e in our initial stage of fitting a nonphysical, emissive volume (Section 5.2). In the second stage of optimization, we set medium emission L_e to zero and require all color to originate from lighting and the reconstructed scattering albedo α . Consistently resolving the additional ambiguities and complexity brought by a real-world capture setup is an important direction for future work, most likely involving domain-specific assumptions and inductive biases.

Experimental conditions. We implemented a differentiable volumetric path tracer with path replay backpropagation [Vicini et al. 2021b]. Unbiased global illumination is simulated with a path length up to 64 vertices (although there is no particular limit, as all evaluated methods have linear time complexity in path length). Scenes are lit by realistic indoor and outdoor high dynamic range light probes. All experiments that are not noted to use the initialization scheme

of Section 5 use a coarse-to-fine approach. More details about the experimental setup are given in Appendix C. Implementation details and performance are discussed in Appendix D.

Role of the optimizer. When optimizing over a large parameter space with noisy gradients, the choice of optimization algorithm (optimizer) has a significant impact on convergence. As we have seen, gradients computed with free-flight sampling-based estimators are prone to large outliers (Figure 9a). Using stochastic gradient descent, either with- (SGDm) or without momentum (SGD), these gradient outliers are almost directly reflected in the descent steps. In practice, regions of the medium that should be fully empty end up being filled with artifacts; see Figures 1, 10, and 11. The outliers occur frequently enough that the optimization can never recover. In contrast, variance-adaptive optimizers such as Adam [Kingma and Ba 2014] are effective at suppressing such outliers—to a perhaps surprising degree. The outliers merely trigger correspondingly low step sizes. Because the outliers only occur in regions where $\sigma_t \approx 0$, the *per-parameter* step size being small in those regions does not prevent the rest of the medium parameters from converging⁵. Altogether, optimizations converge to comparatively good solutions when using Adam, despite the numerous outliers.

Due to these significantly different outcomes, we report results using both SGDm and Adam.

Inverse rendering results. We showcase equal-iterations inverse rendering results on three solid objects (Figure 10) and two complex heterogeneous participating media (Figure 11). The free-flight sampling-based gradient estimator fails to converge on all scenes when used together with SGDm (a). Using the Adam optimizer, high gradient variance manifests instead as a persistent thin “haze” around the reconstructed volumes (b). These artifacts are visible in the included color-mapped density slices and are best appreciated by flipping back and forth between images; please see the interactive viewer included in the **supplemental material**. Defensive sampling (c), while eliminating the larger gradient outliers, does not sufficiently reduce overall variance and therefore exposes the same issues in the reconstructions.

Our proposed gradient estimator and initialization scheme achieve the best reconstructions (e). The lowered gradient variance helps eliminate the haze-like artifacts while the emissive initialization helps converge to sharper results (see also Figure 2c). Note that the reproduction of hard surfaces with volumes could be improved by *e.g.* incorporating the non-exponential transmittance model of Vicini et al. [2021a].

Relighting. A great advantage of using a fully physically based and unbiased inverse rendering method is that it yields inherently editable and relightable media. In Figure 12, we relight the reconstructed medium using an environment that was not seen during optimization. As expected, the NeRF-style nonphysical emissive volume appears as in the original illumination, while ours reacts correctly to the new conditions. Note that robustly disentangling

⁵Note this is only possible because the grid-based parameterization used in our experiments maps parameter space directly to world space. If the medium was defined as *e.g.* the result of a procedural computation, the aggressive reduction of step sizes may slow or prevent overall convergence. Our method’s reduced gradient variance is then even more advantageous.

lighting and reflectance for fully accurate relighting is challenging in its own right and may require observations under different lighting conditions.

7 CONCLUSION AND FUTURE WORK

The physically based differentiable rendering problem is being tackled from many angles: algorithms, discontinuities, systems, inductive priors, implicit representations, etc. Among them, volumetric representations have been particularly successful, due in part to their seemingly trivial differentiability.

Our work shows that severe issues persist in existing gradient estimation methods, leading to biased and high-variance gradients. Adaptive optimizers such as Adam reduce the impact of poor-quality gradients, which may be the reason these issues were not noticed at first. We present a simple, unbiased method tailored to the estimation of gradients with respect to the parameters of scattering volumes. Our estimator retains the linear time and constant memory complexity of the state-of-the-art differentiable rendering algorithm [Vicini et al. 2021b], allowing the efficient estimation of gradients w.r.t. millions of parameters in parallel.

Additionally, we propose a simple way to leverage a nonphysical emissive volume model to bootstrap the optimization of scattering volumes, thus avoiding suboptimal local minima. In applications where the availability of reference data is limited, such as cloud tomography [Loeub et al. 2020], an interesting direction for future work would be to further utilize the recovered emissive medium to act as a proxy model from which imperfect, but infinite reference observations can be generated. In essence, the proposed initialization scheme post-processes a non-physical model to introduce physical realism. Generalizations of this idea, *e.g.* progressively increasing the maximum path depth over the course of the optimization, should be studied and compared in more detail.

Combined, our contributions allow the inverse rendering of challenging media and surfaces, recovering high-resolution density and albedo parameters. The result is inherently editable and relightable. Our evaluation focuses on synthetic scenes with known camera parameters and illumination. We hope that our method can form a building block for further applications, such as the inverse rendering of real-life scenes with imperfect input and using complex appearance models.

Finally, we believe the development of new sampling techniques dedicated to gradients estimation is a highly promising direction for future work.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their helpful feedback and Delio Vicini for his insightful suggestions, as well as providing his implementation of volumetric PRB. Volumes, environment maps and 3D models were generously provided by JangaFX, PolyHaven, Antoan Shiyachki, jgilhutton, vajrablue and Zuendholz. Thanks to Bitterli et al. [2020] for highlighting the reservoir sampling technique. The Publish or Perish software [Harzing 2007] helped carry out a review of recent NeRF articles. This research was

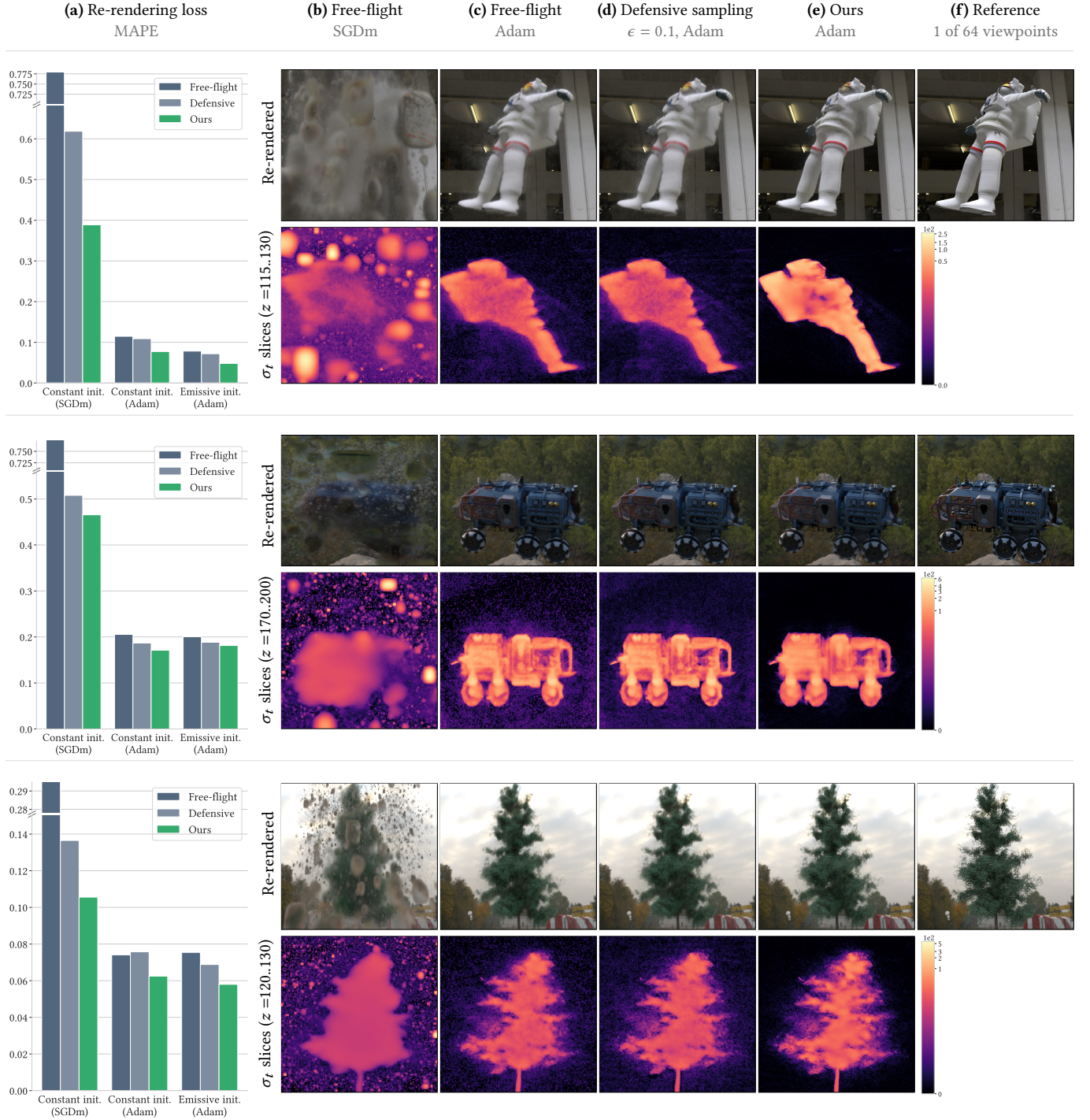


Fig. 10. Our method enables high-quality inverse rendering of complex objects under realistic illumination. We compare inverse rendering results from different combinations of optimizer and gradient estimators. **(a)** We report the mean absolute percentage error computed on all 64 reference images after optimization. **(b)** The high-variance gradients produced by the free-flight sampling based estimator prevent any convergence when optimizing with stochastic gradient descent with momentum. **(c)** The Adam optimizer’s per-parameter adaptive step size significantly reduces the impact of gradient outliers. Nevertheless, persistent haze-like artifacts remain present at the end of optimization. They are also visible in the false-color density slice visualizations. **(d)** Defensive sampling (Section 4.2) prevents large gradient outliers but otherwise introduces additional variance, resulting in similar artifacts. **(e)** Our method uses a novel sampling scheme dedicated to the adjoint to eliminate most of the gradient variance. Combined with our proposed emissive volume initialization (Section 5), we achieve sharper reconstructions and eliminate haze artifacts.

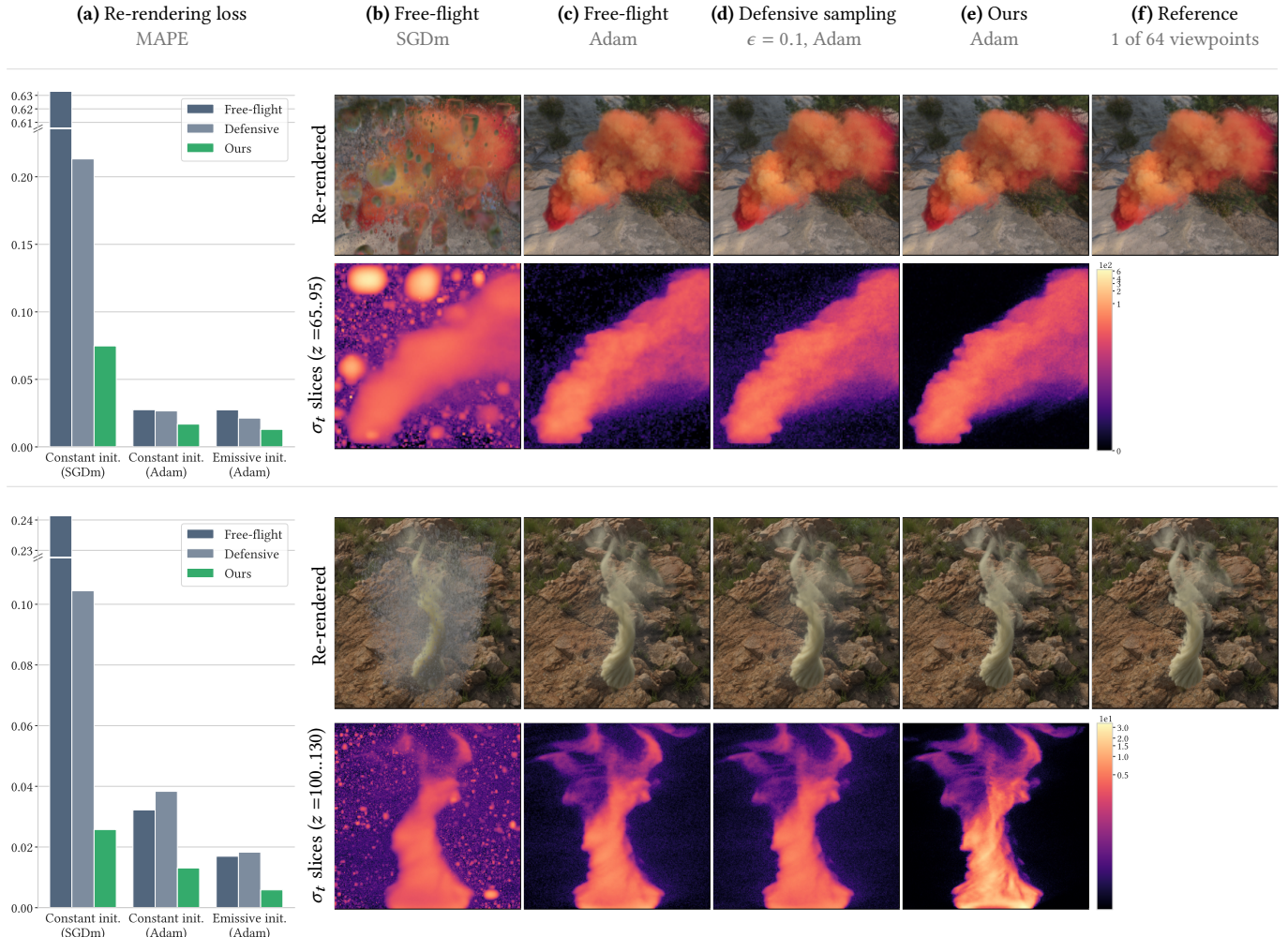


Fig. 11. *Continued from Figure 10.* Our method enables high-quality inverse rendering of heterogeneous scattering and absorbing media under realistic illumination. The differences are best appreciated by flipping between the images, please see the interactive viewer available at: <https://rgl.epfl.ch/publications/NimierDavid2022Unbiased>.

supported by the Swiss National Science Foundation (SNSF) as part of grant 200021_184629 and a Facebook Graduate Fellowship.

REFERENCES

- Dejan Azinović, Tzu-Mao Li, Anton Kaplanyan, and Matthias Nießner. 2019. Inverse Path Tracing for Joint Material and Lighting Estimation. In *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, IEEE.
- Sai Bangaru, Tzu-Mao Li, and Frédo Durand. 2020. Unbiased Warped-Area Sampling for Differentiable Rendering. *ACM Trans. Graph.* 39, 6 (2020), 245:1–245:18.
- Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. 2021a. Mip-NeRF: A Multiscale Representation for Anti-Aliasing Neural Radiance Fields. *arXiv* (2021). <https://jonbarron.info/mipnerf/>
- Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. 2021b. Mip-NeRF 360: Unbounded Anti-Aliased Neural Radiance Fields. *arXiv:2111.12077* (Nov. 2021).
- Sai Bi, Zexiang Xu, Pratul Srinivasan, Ben Mildenhall, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi. 2020. Neural Reflectance Fields for Appearance Acquisition.
- 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>
- Mark Boss, Raphael Braun, Varun Jampani, Jonathan T. Barron, Ce Liu, and Hendrik P.A. Lensch. 2021. NeRD: Neural Reflectance Decomposition from Image Collections. In *IEEE International Conference on Computer Vision (ICCV)*.
- J. C. Butcher and H. Messel. 1958. Electron Number Distribution in Electron-Photon Showers. *Phys. Rev.* 112 (Dec 1958), 2096–2106. Issue 6. <https://doi.org/10.1103/PhysRev.112.2096>
- Min-Te Chao. 1982. A general purpose unequal probability sampling plan. *Biometrika* 69, 3 (12 1982), 653–656. <https://doi.org/10.1093/biomet/69.3.653> [arXiv:https://academic.oup.com/biomet/article-pdf/69/3/653/591311/69-3-653.pdf](https://academic.oup.com/biomet/article-pdf/69/3/653/591311/69-3-653.pdf)
- Chengqian Che, Fujun Luan, Shuang Zhao, Kavita Bala, and Ioannis Gkioulekas. 2020. Towards Learning-based Inverse Subsurface Scattering. 1–12. <https://doi.org/10.1109/ICCP48838.2020.9105209>
- Frank Dellaert and Lin Yen-Chen. 2021. Neural Volume Rendering: NeRF And Beyond.
- Mathieu Galtier, Stéphane Blanco, Cyril Caliot, Christophe Coustet, Jérémie Dauchet, Mouna El Hafi, Vincent Eymet, Richard Fournier, Jacques Gautrais, Anaïs Khuong, et al. 2013. Integral formulation of null-collision Monte Carlo algorithms. *Journal of Quantitative Spectroscopy and Radiative Transfer* 125 (2013).
- Stephan J. Garbin, Marek Kowalski, Matthew Johnson, Jamie Shotton, and Julien Valentin. 2021. FastNeRF: High-Fidelity Neural Rendering at 200FPS. *arXiv:2103.10380* (March 2021).
- Iliyan Georgiev, Zackary Misso, Toshiya Hachisuka, Derek Nowrouzezahrai, Jaroslav Krivánek, and Wojciech Jarosz. 2019. Integral Formulations of Volumetric Transmittance. *ACM Trans. Graph.* 38, 6, Article 154 (nov 2019), 17 pages. <https://doi.org/10.1145/3354000>

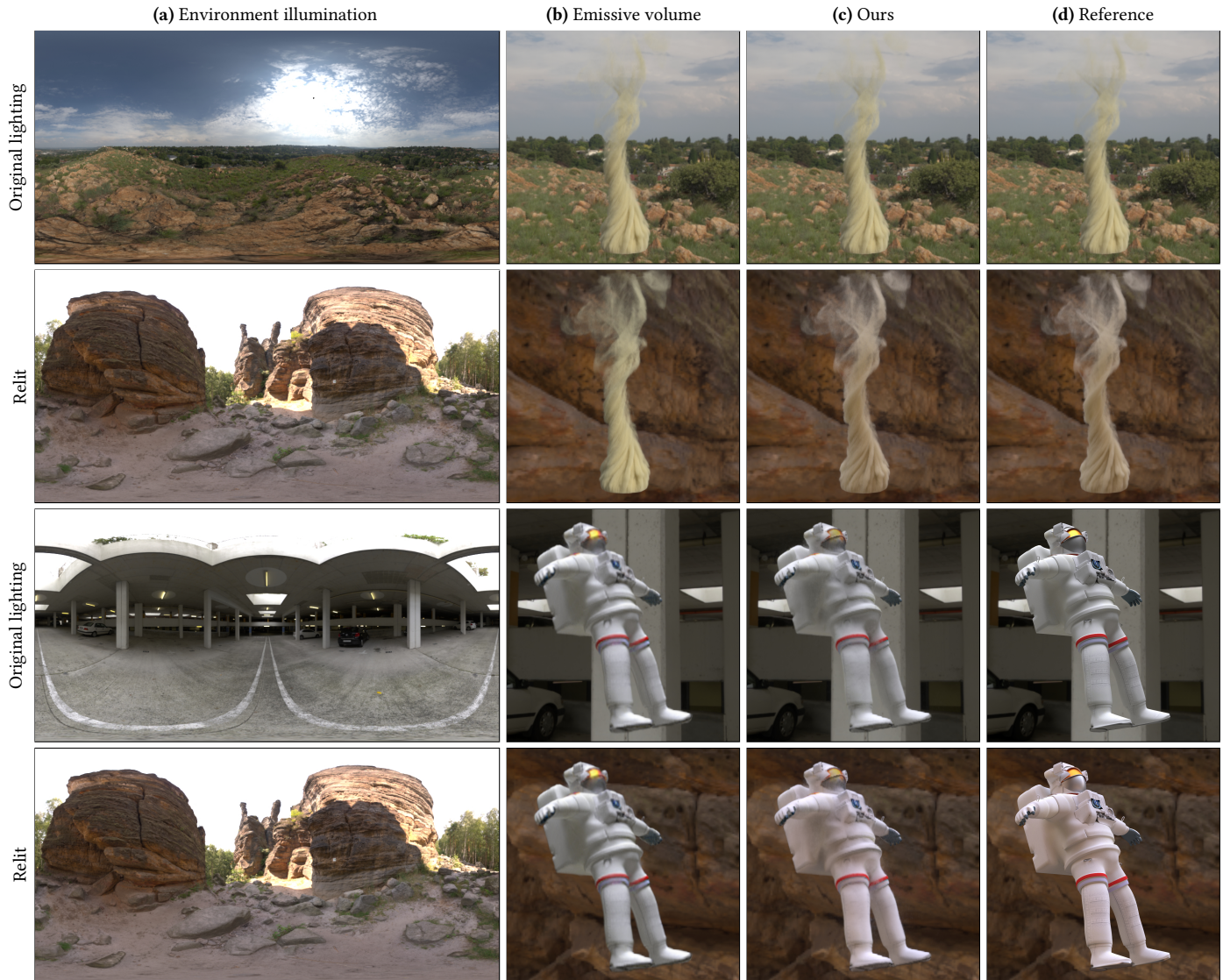


Fig. 12. Nonphysical emissive volume models such as NeRFs [Mildenhall et al. 2021] and Plenoxels [Yu et al. 2021a] faithfully reproduce the target object, but also “bake” the original lighting into the model parameters (b). Using a fully physically based volumetric model and inverse rendering pipeline, we obtain high resolution density and albedo medium parameters. These parameters have a concrete physical meaning, and are inherently editable and relightable. When re-rendered in previously unseen lighting conditions, our optimized models react correctly to the new illumination (c). Note that we did not attempt to model more complex surface reflectance model (e.g. specular reflections), which are orthogonal to our method.

- [//doi.org/10.1145/3355089.3356559](https://doi.org/10.1145/3355089.3356559)
- Adam Geva, Yoav Y Schechner, Yonatan Chernyak, and Rajiv Gupta. 2018. X-ray computed tomography through scatter. In *Proceedings of The European Conference on Computer Vision (ECCV)*. 34–50.
- Ioannis Gkioulekas, Anat Levin, and Todd Zickler. 2016. An evaluation of computational imaging techniques for heterogeneous inverse scattering. In *European Conference on Computer Vision*. Springer, 685–701.
- Ioannis Gkioulekas, Shuang Zhao, Kavita Bala, Todd Zickler, and Anat Levin. 2013. Inverse volume rendering with material dictionaries. *ACM Transactions on Graphics (TOG)* 32, 6 (2013), 1–13.
- A.W. Harzing. 2007. Publish or Perish. <https://harzing.com/resources/publish-or-perish>.
- Jon Hasselgren, Jacob Munkberg, Jaakko Lehtinen, Miika Aittala, and Samuli Laine. 2021. Appearance-Driven Automatic 3D Model Simplification. In *Eurographics Symposium on Rendering*.
- Hailin Jin, Stefano Soatto, and Anthony J. Yezzi. 2005. Multi-view stereo reconstruction of dense shape and complex appearance. *International Journal of Computer Vision* 63, 3 (2005), 175–189.
- James T. Kajiya and Brian P Von Herzen. 1984. Ray Tracing Volume Densities. *SIG-GRAPH Comput. Graph.* 18, 3 (jan 1984), 165–174. <https://doi.org/10.1145/964965.808594>
- Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. 2017. Neural 3D Mesh Renderer. *CoRR* abs/1711.07566 (2017). arXiv:1711.07566 <http://arxiv.org/abs/1711.07566>
- Markus Kettunen, Eugene D’Eon, Jacopo Pantaleoni, and Jan Novák. 2021. An Unbiased Ray-Marching Transmittance Estimator. *ACM Trans. Graph.* 40, 4, Article 137 (jul 2021), 20 pages. <https://doi.org/10.1145/3450626.3459937>
- Pramook Khungurn, Daniel Schroeder, Shuang Zhao, Kavita Bala, and Steve Marschner. 2016. Matching Real Fabrics with Micro-Appearance Models. *ACM Trans. Graph.*

- 35, 1, Article 1 (dec 2016), 26 pages. <https://doi.org/10.1145/2818648>
- Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- Oliver Klehm, Ivo Ihrke, Hans-Peter Seidel, and Elmar Eisemann. 2014. Property and Lighting Manipulations for Static Volume Stylization Using a Painting Metaphor. *IEEE Transactions on Visualization and Computer Graphics* 20, 7 (2014), 983–995. <https://doi.org/10.1109/TVCG.2014.13>
- Peter Kutz, Ralf Habel, Yining Karl Li, and Jan Novák. 2017. Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2017)* 36, 4, Article 111 (2017), 111:1–111:16 pages. <https://doi.org/10.1145/3072959.3073665>
- Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. 2020. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–14.
- Aldo Laurentini. 1994. The Visual Hull Concept for Silhouette-Based Image Understanding. *IEEE Trans. Pattern Anal. Mach. Intell.* 16, 2 (feb 1994), 150–162. <https://doi.org/10.1109/34.273735>
- Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. 2018. Differentiable Monte Carlo Ray Tracing through Edge Sampling. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 37, 6 (2018), 222:1–222:11.
- Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. 2020. Neural Sparse Voxel Fields. *NeurIPS* (2020). <https://lingjie0206.github.io/papers/NSVF/>
- Shichen Liu, Weikai Chen, Tianye Li, and Hao Li. 2019. Soft Rasterizer: Differentiable Rendering for Unsupervised Single-View Mesh Reconstruction. *CoRR* abs/1901.05567 (2019). [arXiv:1901.05567](https://arxiv.org/abs/1901.05567) <http://arxiv.org/abs/1901.05567>
- Tamar Loeub, Aviad Levis, Vadim Holodovsky, and Yoav Y Schechner. 2020. Monotonicity prior for cloud tomography. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII* 16. Springer, 283–299.
- Matthew M. Loper and Michael J. Black. 2014. OpenDR: An approximate differentiable renderer. In *European Conference on Computer Vision*. Springer.
- Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. 2021. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *CVPR*.
- Ben Mildenhall, Peter Hedman, Ricardo Martin-Brualla, Pratul Srinivasan, and Jonathan T. Barron. 2021. NeRF in the Dark: High Dynamic Range View Synthesis from Noisy Raw Images. *arXiv:2111.13679* (Nov. 2021).
- Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV*.
- Bailey Miller, Iliyan Georgiev, and Wojciech Jarosz. 2019. A Null-Scattering Path Integral Formulation of Light Transport. *ACM Trans. Graph.* 38, 4, Article 44 (jul 2019), 13 pages. <https://doi.org/10.1145/3306346.3323025>
- Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *arXiv:2201.05989* (Jan. 2022).
- Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. 2021. Extracting Triangular 3D Models, Materials, and Lighting From Images. *arXiv:2111.12503* (2021).
- Merlin Nimier-David, Sébastien Speierer, Benoît Ruiz, and Wenzel Jakob. 2020. Radiative Backpropagation: An Adjoint Method for Lightning-Fast Differentiable Rendering. *Transactions on Graphics (Proceedings of SIGGRAPH)* 39, 4 (July 2020). <https://doi.org/10.1145/3386569.3392406>
- Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. 2019. Mitsuba 2: A Retargetable Forward and Inverse Renderer. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38, 6 (Dec. 2019). <https://doi.org/10.1145/3355089.3356498>
- Thomas Klaus Nindl, Tomáš Iser, Tobias Rittig, Alexander Wilkie, and Jaroslav Krivánek. 2021. A Gradient-Based Framework for 3D Print Appearance Optimization. *ACM Trans. Graph.* 40, 4, Article 178 (July 2021), 15 pages. <https://doi.org/10.1145/3450626.3459844>
- Jan Novák, Andrew Selle, and Wojciech Jarosz. 2014. Residual Ratio Tracking for Estimating Attenuation in Participating Media. *ACM Trans. Graph.* 33, 6, Article 179 (nov 2014), 11 pages. <https://doi.org/10.1145/2661229.2661292>
- Felix Petersen, Amit H. Bermano, Oliver Deussen, and Daniel Cohen-Or. 2019. Pix2Vex: Image-to-Geometry Reconstruction using a Smooth Differentiable Renderer. *CoRR* abs/1903.11149 (2019). [arXiv:1903.11149](https://arxiv.org/abs/1903.11149) <http://arxiv.org/abs/1903.11149>
- Helge Rhodin, Nadia Robertini, Christian Richardt, Hans-Peter Seidel, and Christian Theobalt. 2015. A Versatile Scene Model with Differentiable Visibility Applied to Generative Pose Estimation. In *Proceedings of ICCV 2015*.
- Roi Ronen, Yoav Y. Schechner, and Eshkol Eytan. 2021. 4D Cloud Scattering Tomography. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5520–5529.
- Johannes Lutz Schönberger and Jan-Michael Frahm. 2016. Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Yael Sde-Chen, Yoav Y. Schechner, Vadim Holodovsky, and Eshkol Eytan. 2021. 3DeepCT: Learning volumetric scattering tomography of clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5671–5682.
- Pratul P. Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T. Barron. 2021. NeRV: Neural Reflectance and Visibility Fields for Relighting and View Synthesis. In *CVPR*.
- Cheng Sun, Min Sun, and Hwann-Tzong Chen. 2021a. Direct Voxel Grid Optimization: Super-fast Convergence for Radiance Fields Reconstruction. *arXiv:2111.11215* (Nov. 2021).
- Qilin Sun, Congli Wang, Fu Qiang, Dun Xiong, and Heidrich Wolfgang. 2021b. End-to-end complex lens design with differentiable ray tracing. *ACM Transactions on Graphics* 40, 4 (2021), 1–13.
- Jean-Marc Tregan, Stéphane Blanco, Jérémie Dauchet, Mouna El Hafi, Richard Fournier, L. Ibarra, P. Lapeyre, and Najda Villefranque. 2020. Convergence issues in derivatives of Monte Carlo null-collision integral formulations: a solution. *J. Comput. Phys.* 413 (2020), 109463.
- Chia-Yin Tsai, Aswin C. Sankaranarayanan, and Ioannis Gkioulekas. 2019. Beyond Volumetric Albedo—A Surface Optimization Framework for Non-Line-Of-Sight Imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1545–1555.
- Eric Veach and Leonidas J. Guibas. 1995. Optimally Combining Sampling Techniques for Monte Carlo Rendering. 419–428. <https://doi.org/10.1145/218380.218498>
- Delio Vicini, Wenzel Jakob, and Anton Kaplanyan. 2021a. A non-exponential transmittance model for volumetric scene representations. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–16.
- Delio Vicini, Sébastien Speierer, and Wenzel Jakob. 2021b. Path Replay Backpropagation: Differentiating Light Paths using Constant Memory and Linear Time. *Transactions on Graphics (Proceedings of SIGGRAPH)* 40, 4 (Aug. 2021), 108:1–108:14. <https://doi.org/10.1145/3450626.3459804>
- E. Woodcock, T. Murphy, P. Hemmings, and S. Longworth. 1965. Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry. In *Proceedings of the Conference on Applications of Computing Methods to Reactor Problems*. Argonne National Laboratory, 557.
- Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. 2021. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems* 34 (2021).
- Alex Yu, Sara Fridovich-Keil, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. 2021a. Plenoxels: Radiance Fields without Neural Networks. *arXiv:2112.05131* (Dec. 2021).
- Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. 2021b. PlenOctrees for Real-time Rendering of Neural Radiance Fields. In *ICCV*.
- Tizian Zeltner, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob. 2021. Monte Carlo Estimators for Differential Light Transport. *Transactions on Graphics (Proceedings of SIGGRAPH)* 40, 4 (Aug. 2021). <https://doi.org/10.1145/3450626.3459807>
- Cheng Zhang, Zhao Dong, Michael Doggett, and Shuang Zhao. 2021a. Antithetic sampling for Monte Carlo differentiable rendering. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–12.
- Cheng Zhang, Bailey Miller, Kai Yan, Ioannis Gkioulekas, and Shuang Zhao. 2020. Path-Space Differentiable Rendering. *ACM Trans. Graph.* 39, 4 (2020), 143:1–143:19.
- Cheng Zhang, Lifan Wu, Changxi Zheng, Ioannis Gkioulekas, Ravi Ramamoorthi, and Shuang Zhao. 2019. A differential theory of radiative transfer. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–16.
- Cheng Zhang, Zihan Yu, and Shuang Zhao. 2021b. Path-Space Differentiable Rendering of Participating Media. *ACM Trans. Graph.* 40, 4 (2021), 76:1–76:15.
- Shuang Zhao, Ravi Ramamoorthi, and Kavita Bala. 2014. High-order similarity relations in radiative transfer. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–12.
- Shuang Zhao, Lifan Wu, Frédo Durand, and Ravi Ramamoorthi. 2016. Downsampling Scattering Parameters for Rendering Anisotropic Media. *ACM Trans. Graph.* 35, 6, Article 166 (nov 2016), 11 pages. <https://doi.org/10.1145/2980179.2980228>
- Quan Zheng, Gurprit Singh, and Hans-Peter Seidel. 2021. Neural Relightable Participating Media Rendering. *Advances in Neural Information Processing Systems* 34 (2021).

A NULL-SCATTERING INTEGRAL FORMULATION

In Section 4.2, we identified a singularity $1/\sigma_t$ when evaluating in-scattering gradients using free-flight sampling based estimators. We now show that this singularity materializes as well when deriving the estimators starting from the null-scattering integral formulation.

Using the notation and terms introduced in Section 3.1, the null-scattering integral form of the radiative transfer equation [Galtier

et al. 2013; Kutz et al. 2017; Miller et al. 2019] is

$$L_i(\mathbf{x}, \omega) = \int_0^{t_s} \bar{\sigma} \bar{T}(t) \left[\frac{\sigma_a(t)}{\bar{\sigma}} L_e(t) + \frac{\sigma_s(t)}{\bar{\sigma}} L_s(t, \omega) + \frac{\sigma_n(t)}{\bar{\sigma}} L_i(t, \omega) \right] dt + T(t_s) [L_e(t_s) + L_s(t_s, \omega)], \quad (23)$$

where t_s is the distance to the next surface along the ray (\mathbf{x}, ω) , e.g. the medium's boundary. $\bar{T}(t) = \exp(-\bar{\sigma} t)$ corresponds to the transmittance of the homogenized medium and $\sigma_n = \bar{\sigma} - \sigma_t$.

Adjoint null-scattering radiative transfer equation. Assuming $L_e = 0$ within the medium and using the $\theta = (\sigma_t, \alpha)$ parametrization, Equation (23) simplifies to

$$L_i(\mathbf{x}, \omega) = \int_0^{t_s} \bar{\sigma} \bar{T}(t) \left[\frac{\sigma_t(t) \alpha(t)}{\bar{\sigma}} L_s(t, \omega) + \frac{\sigma_n(t)}{\bar{\sigma}} L_i(t, \omega) \right] dt + T(t_s) [L_e(t_s) + L_s(t_s, \omega)]. \quad (24)$$

In the following, we consider $\partial_\theta \bar{\sigma} = 0$, as the majorant is constant. Taking the derivative with respect to scene parameters θ and omitting the dependence on ω , we obtain the following terms. Similar to Equation (13), the first term captures how the *in-scattered* radiance can increase due to a local density increase:

$$\partial_\theta L_i = \int_0^{t_s} \bar{\sigma} \bar{T}(t) \frac{\partial_\theta [\sigma_t(t) \alpha(t)]}{\bar{\sigma}} L_s(t) dt \dots \quad (25)$$

Recalling that $\sigma_n = 1 - \sigma_t$, we see that the next expression has a role similar to Equation (6)—it accounts for the density increases at a positions prior to real interactions:

$$\dots + \int_0^{t_s} \bar{\sigma} \bar{T}(t) \frac{\partial_\theta \sigma_n(t)}{\bar{\sigma}} L_i(t) dt + T(t_s) \left[\int_0^{t_s} -\partial_\theta \sigma_t(t') dt' \right] [L_e(t_s) + L_s(t_s)] \dots \quad (26)$$

The third expression captures changes later along the path, which are weighted by the path throughput:

$$\dots + \int_0^{t_s} \bar{\sigma} \bar{T}(t) \left[\frac{\sigma_t(t) \alpha(t)}{\bar{\sigma}} \partial_\theta L_s(t) + \frac{\sigma_n(t)}{\bar{\sigma}} \partial_\theta L_i(t) \right] dt + T(t_s) [\partial_\theta L_e(t_s) + \partial_\theta L_s(t_s)]. \quad (27)$$

Naïve estimators with null-scattering. Since the null-scattering formulation implies sampling free-flight distances $t \sim \bar{\sigma} \bar{T}(t)$, it may appear as if the problematic $1/\sigma_t$ term discussed in Section 4.2 has been avoided. However, it manifests at the very next step: once the distance t has been sampled from the homogenized medium, testing whether a *real* or *null* (virtual) particle has been encountered is determined by the probabilities

$$p_{\text{real}}(t) = \frac{\sigma_t(t)}{\bar{\sigma}} \quad \text{and} \quad p_{\text{null}}(t) = \frac{\sigma_n(t)}{\bar{\sigma}}, \quad (28)$$

reintroducing the problematic factor. Hence, the resulting estimators of adjoint terms (25) and (26) suffer from the same issue as before:

$$\langle \partial L_1^N \rangle = \frac{\bar{\sigma}}{\sigma_t(t)} \frac{\partial_\theta [\sigma_t(t) \alpha(t)]}{\bar{\sigma}} L_s(t) = \frac{\partial_\theta [\sigma_t(t) \alpha(t)]}{\sigma_t(t)} L_s(t), \quad (29)$$

$$\text{and} \quad \langle \partial L_2^N \rangle = \frac{\partial_\theta \sigma_n(t)}{\sigma_n(t)} L_i(t). \quad (30)$$

Symmetric issue in null interaction gradients. While we have focused our efforts on issues due to real interactions where $\sigma_t \approx 0$, Tregan et al. [2020] have identified a similar issue for $\sigma_n \approx \bar{\sigma}$, which becomes apparent in Equation (30). Depending on the application, selecting a sufficiently large majorant $\bar{\sigma}$ can be a practical way to sidestep the problem. Another simple solution would be to estimate transmittance gradients of Equations (10) and (26) not at locations $t' \in (0, \min(t, t_s))$ corresponding to null interactions, but rather sampled uniformly at random on the segment. Since the integrands do not include other factors beyond $\partial_\theta \sigma_t$, uniform sampling is appropriate. The estimator would then include a sampling weight equal to the length of the segment. Our method generalizes the solution of Tregan et al. by handling both the in-scattering and transmittance gradient singularities.

In practice, we found transmittance gradients to be well-behaved and did not observe significant differences between these alternatives when using a majorant $1.01\times$ larger than the maximum σ_t value.

B DIFFERENTIAL RESIDUAL RATIO TRACKING

While we decided not to use that variant in our experiments due to the additional hyperparameter (medium control density), we include the pseudocode for *differential residual ratio tracking* in Listing 2.

C EXPERIMENTAL SETTING

We provide more details about the inverse rendering experiments presented in Section 6.3.

The optimization aims to minimize the L_1 difference to the 64 reference images, with camera positions sampled uniformly in a circle around the object at random altitudes. All optimizations were run for 6000 iterations. In each iteration, a batch of 32768 pixels is sampled uniformly from the set of all pixels from all reference images. Primary rays are then sampled from within the footprint of selected pixels.

In step (I) of path replay backpropagation (Section 3.2), the primal value is estimated with 1024 samples per pixel, for a total of 33.5 million rays. The high-quality primal estimate is used to compute the objective function. Steps (II) and (III) then use a second, uncorrelated set of primary rays with 16 samples per pixel to estimate gradients.

As noted by Azinović et al. [2019], we found that allocating a high sample count to the primal estimate results in better convergence, especially in challenging volumetric scenes. Luckily, primal paths are also less computationally intensive to evaluate.

In any given experiment, the same learning rate is used for all estimators. This value must be adapted to the specific scene's physical scale and any factor applied to the medium density. Intuitively,

```

def drtt_sample(t_max):
    Tr = 1.
    t_run = w_acc = 0.

    while t_run < t_max:
        # Sample from the homogenized medium
        dt = -log(1-rng())/σ
        dt = min(dt, t_max - t_run)

        # Propose current exponential segment with weight
        # equal to area under the curve.
        w_step = Tr * (1 - exp(-dt σ_c)) / σ_c
        w_acc += w_step
        if (rng() * w_acc) < w_step:
            # Update reservoir (happens at least once)
            reservoir = (t_run, dt)

        # Update transmittance estimate
        Tr *= exp(-σ_c dt) * (1 - (σ_t(t_run) - σ_c) / σ)
        t_run = t_run + dt

    # Final sampling over the chosen exponential segment
    t, dt = reservoir
    t -= log(1 - rng() * (1 - exp(-σ_c dt))) / σ_c
    assert t <= t_max
    return t, w_acc

```

Listing 2. Differential residual ratio tracking pseudocode. σ_c is the control density and $\bar{\sigma}$ the majorant of the residual medium: $\bar{\sigma} \geq |\sigma_t - \sigma_c|$. Similarly to differential *ratio* tracking (Listing 1), we combine the residual ratio tracking transmittance estimator with reservoir sampling to produce weighted samples t with density equal to the transmittance function.

a scalar factor applied to the σ_t values would be reflected in the gradients, resulting in a different effective step size. In practice, we set the learning rate to values in the ranges $5 \cdot 10^{-2}$ to $5 \cdot 10^1$ for SGDM and 10^{-3} to 10^{-2} for Adam. It is automatically set twice as high for the albedo parameters.

Results are shown at equal iteration count accross methods, as the runtime depends not only on the chosen method but also on the state of the reconstruction itself. Equal-time results would therefore be difficult to interpret.

Several commonly used techniques further improve the quality of the results: the optimization starts with 16^3 fewer parameters than the desired resolution. The grid resolution is then doubled four times during the optimization (coarse-to-fine). When using SGDM, the learning rate is multiplied by 4 at each upsampling step. Finally, the learning rate is halved six times over the last quarter of the optimization.

D IMPLEMENTATION AND PERFORMANCE

All evaluated methods were implemented within a path replay back-propagation integrator in Mitsuba 2 [Nimier-David et al. 2019]. Our implementation runs efficiently in reverse mode and on the GPU, making it possible to optimize many parameters in parallel: the medium density σ_t is typically represented by a $256 \times 256 \times 256$ dense grid, and the single-scattering albedo α by a $256 \times 256 \times 256 \times 3$ grid for a total of more than 67 million parameters. It is therefore possible to optimize scattering volumes with full unbiased global illumination, perhaps contrary to common belief.

Beyond important features such as next-event estimation, our implementation includes support for a spatially-varying majorant

(to which our method is agnostic). Allowing the majorant to vary on a coarse grid becomes important for performance when the medium’s bounding box contains both very dense and very thin or empty regions, see Section 3.1.

Compared to directly using the free-flight sampling technique for gradient estimation, our method requires tracing one recursive path per path in order to estimate incident illumination at the newly sampled location. In practice, we found the additional overhead to be relatively low compared to the expected $2\times$ factor. In Table 1, we report the median runtime per iteration on three different scenes with different densities. One iteration includes the three steps of path replay backpropagation described in Section 3.2, with next event estimation and a maximum path length of 64. Timings were measured on an NVIDIA RTX 3090 GPU over 50 runs.

Note that there remains room for optimization, for example using an adaptive voxel grid to store medium properties, porting performance-critical functions to CUDA or adaptively sampling primary rays based on the re-rendering error.

E ADDITIONAL VALIDATIONS

Correctness. We further validate our method’s correctness by comparing the computed gradient to analytic ground truth values in a simplified case. We consider a homogeneous, fully forward scattering, non-emissive, single-scattering medium. The volume rendering equation (2) then simplifies to:

$$L_i(\mathbf{x}, \boldsymbol{\omega}) = \int_0^{t_s} T(0, t) \sigma_s(t) L_s(t, \boldsymbol{\omega}) dt + T(0, t_s) L_e(t_s) \quad (31)$$

Since the medium is fully forward scattering, $f_s(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}') = \delta(\boldsymbol{\omega} - \boldsymbol{\omega}')$ and therefore $L_s(t, \boldsymbol{\omega}) = L_i^1(t, \boldsymbol{\omega})$:

$$= \sigma_t \int_0^{t_s} T(0, t) L_i^1(t, \boldsymbol{\omega}) dt + T(0, t_s) L_e \quad (32)$$

After an interaction within the single-scattering medium at t , the ray continues without further scattering, therefore $L_i^1(t, \boldsymbol{\omega}) = T(t, t_s) L_e(t_s)$. This drastically simplifies the expression of incident radiance:

$$= \sigma_t \int_0^{t_s} T(0, t) T(t, t_s) L_e dt + T(0, t_s) L_e \quad (33)$$

$$= \sigma_t \alpha L_e \int_0^{t_s} T(0, t_s) dt + T(0, t_s) L_e \quad (34)$$

$$= L_e T(t_s) (\sigma_t \alpha t_s + 1), \quad (35)$$

which is easily evaluated in closed form.

We create a scene containing such a single-scattering medium with $t_s = L_e = 1$. We then estimate density and albedo gradients using both the free-flight based estimator (Section 4.2) and ours (Section 4.3). The results are plotted as a function of the medium’s density value in Figure 13. While all methods match the analytic gradients, density gradients estimated with the free-flight estimator suffer from high variance as σ_t approaches zero. Our estimator computes unbiased and low-variance gradients.

Variance. In Figure 15, we study the effect of the proportion of empty space and the density of a volume on the standard deviation of

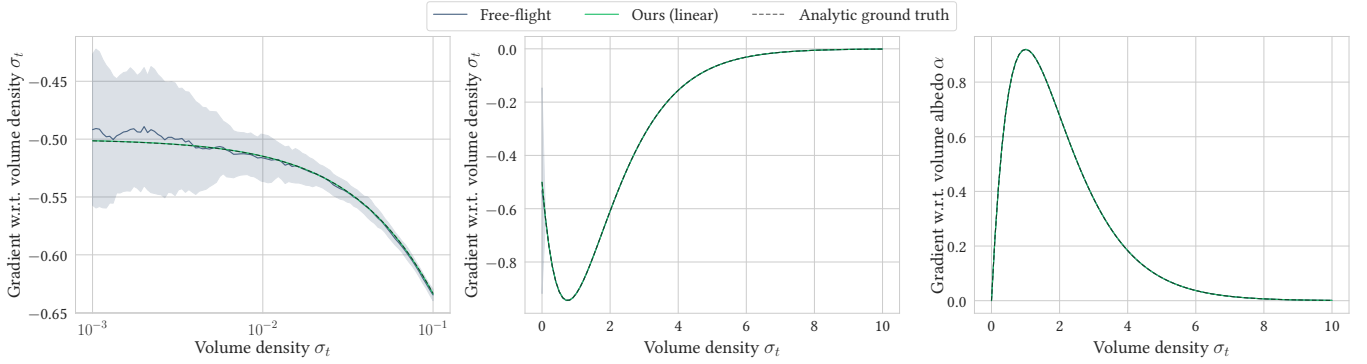


Fig. 13. In a single-scattering, fully forward scattering homogeneous medium, we compare the gradients computed by the standard free-flight based estimator and our method to the ground truth analytic gradients. While all methods compute correct gradients in this setting for all $\sigma_t > 0$, we note once again that free-flight based gradients suffer from high standard deviation as σ_t approaches zero (shaded area, left plot). With our estimator, the standard deviation is low enough not to appear on the plots.

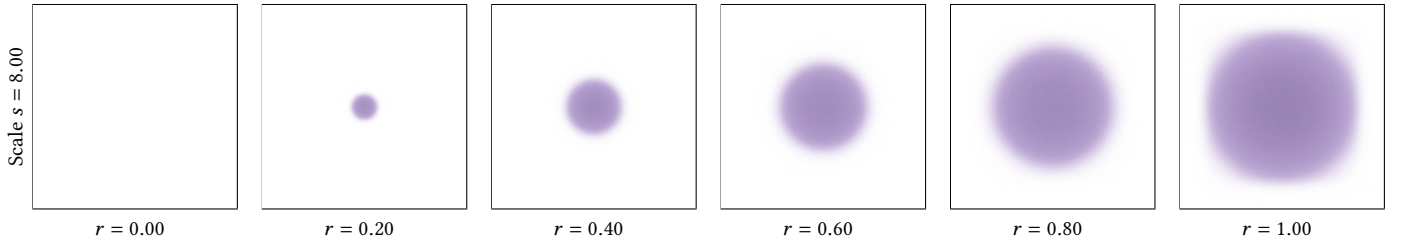


Fig. 14. Preview of the scene used in Figure 15 for density scale 0.8, with filling ratio r varying from 0 to 1. Note that the density is generated following a 3D Gaussian, therefore even areas appearing bright in the image have a small, but nonzero density.

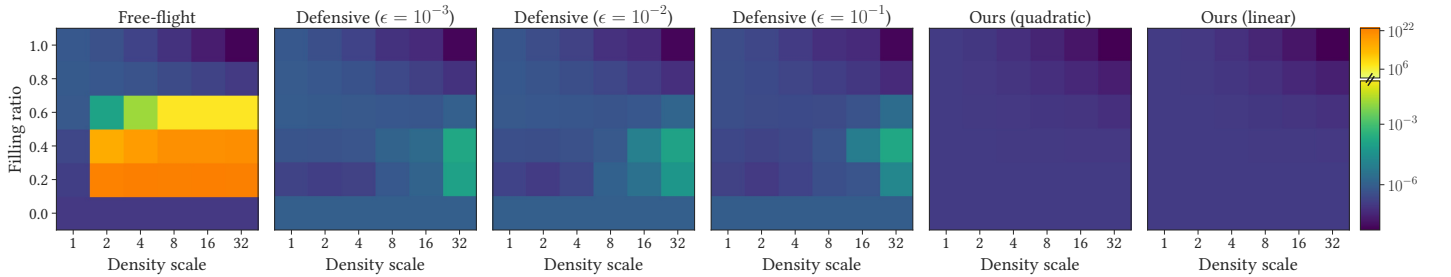


Fig. 15. Effect of the proportion of empty space and the density of a volume on the standard deviation of gradients. For each method, we show the average standard deviation of σ_t gradients for combinations of (s, r) parameters. To visualize standard deviation values spanning many orders of magnitude, we use a log-scaled colormap. A discontinuity in the mapping is introduced to represent extreme variance from the free-flight estimator in the low σ_t regime.

gradients computed by various estimators. The test scene, visualized in Figure 14, contains a heterogeneous medium with density:

$$\sigma_t(\mathbf{x}) = s \exp\left(-\frac{1}{2} \left(\frac{\mathbf{x} - 0.5}{r/6}\right)^2\right), \quad (36)$$

where the parameters s and r control the overall medium density and proportion of empty space respectively. The medium albedo and environment illumination are constant ($\alpha = (0.8, 0.7, 0.9)$, $L_e = 1$). The average standard deviation of density gradients are shown for various combinations of (s, r) parameters and gradient estimators in

Figure 15. For all combinations of medium scale and filling ratio, both our quadratic and linear complexity estimators achieve significantly lower standard deviation than the evaluated baselines.