Objects as volumes: A stochastic geometry view of opaque solids

Bailey Miller, Hanyu Chen, Alice Lai, Ioannis Gkioulekas
Carnegie Mellon University

Abstract

We develop a theory for the representation of opaque solids as volumes. Starting from a stochastic representation of opaque solids as random indicator functions, we prove the conditions under which such solids can be modeled using exponential volumetric transport. We also derive expressions for the volumetric attenuation coefficient as a functional of the probability distributions of the underlying indicator functions. We generalize our theory to account for isotropic and anisotropic scattering at different parts of the solid, and for representations of opaque solids as stochastic implicit surfaces. We derive our volumetric representation from first principles, which ensures that it satisfies physical constraints such as reciprocity and reversibility. We use our theory to explain, compare, and correct previous volumetric representations, as well as propose meaningful extensions that lead to improved performance in 3D reconstruction tasks.

1. Introduction

Volumetric representations have a long history in applied physics [38] and computer graphics [44], where they enable efficient light transport simulation in translucent objects (e.g., tissue, clouds, materials such as wax and soap) and participating media (e.g., smoke, fog, murky water). Since the introduction of NeRF by Mildenhall et al. [36], there has been a proliferation of neural rendering techniques [5, 17, 45, 50, 61, 63, 66, 68] that use volumetric representations for scenes much unlike the above examples, comprising opaque objects (rather than translucent ones) in free space (rather than volumetric media). The tremendous success of volumetric representations for scenes without subsurface or volumetric scattering motivates questions such as: Why can we use volumetric light transport to simulate scenes with only light-surface interactions? What is the mathematical underpinning for modeling an opaque object as a volume? What are the properties of such a volume?

Our goal is to answer these questions. We start from first principles, revisiting the derivation of volumetric representations for translucent objects and participating media. As recent work in computer graphics highlights [9, 15, 26], volumetric representations are a formalism for querying *stochastic geometry* [12, 48]: From this lens, volumetric quantities such as transmittance and free-flight distribution are the an-

swers to queries such as "are two points mutually visible" (a visibility query) and "what is the distance to first intersection along a ray" (a ray-casting query), when the geometry occluding visibility and terminating rays is stochastic.

Volumetric representations for translucent and participating media are stochastic abstractions of their microscopic structure: Such media comprise numerous microscopic particles that reflect and occlude light rays. Modeling explicit microparticle configurations, and rendering light transport through them, is prohibitively expensive. As a compromise for efficiency, volumetric representations allow to simulate light transport in such media *on average* [6]. These representations replace explicit with *statistical* descriptions of microparticle configurations (e.g., average particle location, size, shape, and orientation), analogously to statistical BRDF models for surface microgeometry [13, 14, 24, 46, 60]. Computer graphics has developed volumetric representations for microparticle media that account for details such as varying particle size and material [16, 32], shape and orientation [23, 25], and placement correlations [9, 15, 26].

We develop (Section 3) analogous volumetric representations for scenes comprising opaque macroscopic 3D objects, or *opaque solids*, using stochastic geometry theory. We prove (Section 3.1) formal conditions for *exponential* volumetric representations to apply to stochastic opaque solids; and functional relationships between volumetric parameters (i.e., attenuation coefficient) and stochastic geometry models. We adapt (Section 3.2) anisotropic volumetric representations of microparticle geometry to opaque solids, to account for effects such as directionally-dependent foreshortening near surfaces, and directionally-independent attenuation far from them. We extend (Section 3.3) our volumetric representations to utilize geometry models common in current practice (e.g., implicit surfaces). Our theory delivers volumetric representations equipped with properties necessary for physical plausibility (e.g., reciprocity and reversibility).

Our work is not the first to derive volumetric representations for opaque solids [45, 52, 63, 66]. Previous derivations generally consider how to transform a deterministic geometry representation (e.g., signed distance function) into a volumetric one that behaves approximately like the deterministic geometry. Despite its empirical success, this methodological

approach remains *heuristic* and requires arbitrary choices (e.g., deciding what properties of deterministic geometry to preserve in the volumetric representation). By contrast, our derivation is *rigorous*, starting from only the axioms of volumetric transport, and helps place this prior work on a solid mathematical footing: We show (Section 4) that our theory explains previous volumetric representations as special cases of ours, corresponding to different stochastic modeling choices for the underlying opaque geometry. Our theory additionally highlights and addresses critical defects of previous volumetric representations (e.g., lack of reciprocity and reversibility), and generalizes them in principled ways.

Our volumetric representation can be readily incorporated within existing volumetric neural rendering pipelines [37, 54, 63, 67]. We show experimentally (Section 5) that replacing previous volumetric representations [63, 66] with ours leads to significantly better (qualitatively and quantitatively) 3D reconstructions on common datasets. We provide interactive visualizations, open-source code, and a supplement with all appendices on the project website.[1]

2. Volumetric light transport background

We begin with background on *volumetric light transport* (also known as radiative transfer). We follow Bitterli et al. [9] for our review, and refer to Preisendorfer [48, Chapter XV] and Chiu et al. [12] for a more comprehensive discussion.

Setup. Volumetric light transport models scenes with *stochastic* geometry (Figure 1). Classically in computer graphics, these are scenes comprising numerous microscopic particles [39] (e.g., translucent materials); whereas in neural rendering, they are scenes comprising macroscopic opaque objects. We term the two settings *stochastic microparticle geometry* and *stochastic solid geometry*, respectively.

In both settings, volumetric light transport algorithms simulate *expected* radiometric measurements over all realizations of the stochastic geometry [6]. They leverage the fact that deterministic light transport algorithms (e.g., path tracing) interact with scene geometry only through two geometric queries: **Q1.** *visibility queries* to compute the *visibility* $V(\boldsymbol{x}, \boldsymbol{y}) \in \{0, 1\}$ between points $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^3$; **Q2.** *ray-casting queries* to compute the *free-flight distance* $t^*_{\boldsymbol{x},\boldsymbol{\omega}} \in [0, \infty)$ a ray with origin $\boldsymbol{x} \in \mathbb{R}^3$ and direction $\boldsymbol{\omega} \in \mathcal{S}^2$ travels until it first intersects the scene. Thus, we can translate light transport algorithms from the deterministic to the volumetric setting by *stochasticizing* these geometric queries [9].

To facilitate discussion of these stochastic queries, we introduce some notation. We denote by $\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv \boldsymbol{x} + t \cdot \boldsymbol{\omega}$ the point on a ray with origin $\boldsymbol{x} \in \mathbb{R}^3$ and direction $\boldsymbol{\omega} \in \mathcal{S}^2$ after travel distance $t \in [0, \infty)$; and by $V_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv V(\boldsymbol{x}, \boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t))$ the visibility along the ray. Then, the free-flight distance becomes $t^*_{\boldsymbol{x},\boldsymbol{\omega}} \equiv \max\{t \in [0, \infty) : V_{\boldsymbol{x},\boldsymbol{\omega}}(t) = 1\}$, and we denote by $\boldsymbol{r}^*_{\boldsymbol{x},\boldsymbol{\omega}} \equiv \boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t^*_{\boldsymbol{x},\boldsymbol{\omega}})$ the first intersection point.
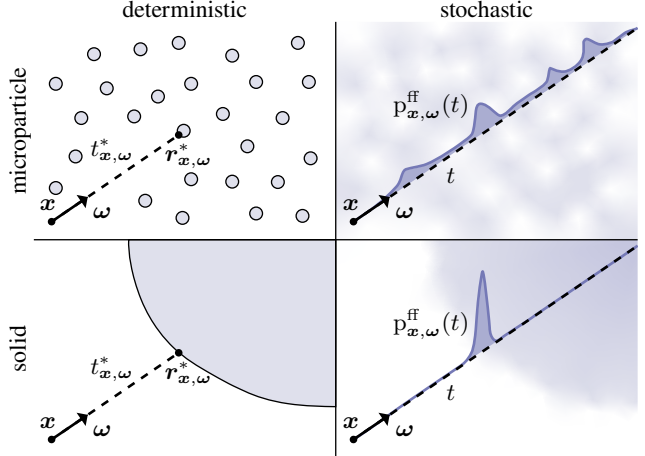
Figure 1. Volumetric representations replace deterministic (left) with stochastic (right) ray casting: rather than find the first intersection with deterministic geometry, they use the free-flight distribution along a ray to represent the probability of first intersection with stochastic geometry. Classical volumetric representations describe stochastic microparticle geometry (top). We derive volumetric representations for stochastic solid geometry (bottom).

Definition 1. *In a scene with stochastic geometry $\mathcal{O}$, the* transmittance *along a ray $\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$ is the probability of visibility from the ray origin $\boldsymbol{x}$—equivalently, the tail distribution of the free-flight distance $t^*_{\boldsymbol{x},\boldsymbol{\omega}}$:*

$$\mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv \mathrm{Pr}_{\mathcal{O}}\{V_{\boldsymbol{x},\boldsymbol{\omega}}(t) = 1\} = \mathrm{Pr}_{\mathcal{O}}\{t^*_{\boldsymbol{x},\boldsymbol{\omega}} \geq t\}. \quad (1)$$

The free-flight distribution *along a ray is the probability density function of the free-flight distance $t^*_{\boldsymbol{x},\boldsymbol{\omega}}$:*

$$\mathrm{p}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv -\frac{\mathrm{d}\,\mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}}{\mathrm{d}t}(t). \quad (2)$$

The attenuation coefficient *at point $\boldsymbol{x}$ and direction $\boldsymbol{\omega}$ is the probability density of zero free-flight distance (equivalently, probability density of ray termination through $\boldsymbol{x}$ along $\boldsymbol{\omega}$):*

$$\sigma(\boldsymbol{x}, \boldsymbol{\omega}) \equiv \mathrm{p}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}(0). \quad (3)$$

We postpone definitions of stochastic geometry $\mathcal{O}$ till Section 3. The transmittance inherits the following properties from visibility: 1. it is *reciprocal*, $\mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \mathrm{T}_{\boldsymbol{y},-\boldsymbol{\omega}}(t)$ if $\boldsymbol{y} \equiv \boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$; 2. it is monotonically non-increasing, $\mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \leq \mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}(s)$ if $t < s$; 3. it satisfies $\mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}(0) = 1$. The transmittance and free-flight distribution generalize the visibility (Q1) and ray-casting (Q2) queries, respectively: for deterministic geometry, Equation (1) reduces to the deterministic visibility, and Equation (2) reduces to the Dirac delta distribution $\delta(t - t^*_{\boldsymbol{x},\boldsymbol{\omega}})$ centered at the deterministic free-flight distance. The attenuation coefficient will become important when we discuss exponential transport below.

We can use these definitions to generalize deterministic light transport algorithms, which recursively use the equation for the conservation of radiance along a ray, $L_{\mathrm{i}}(\boldsymbol{x}, \boldsymbol{\omega}) = L_{\mathrm{o}}(\boldsymbol{r}^*_{\boldsymbol{x},\boldsymbol{\omega}}, -\boldsymbol{\omega})$, to volumetric light transport algorithms,
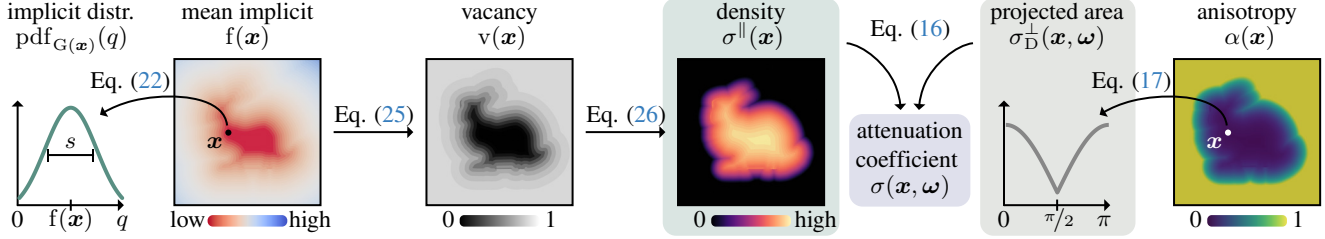
Figure 2. Overview of our theory, presented in Theorem 4, Definition 5, and Proposition 7.

which recursively use the *expectation* of this equation:

$$\mathbb{E}_{\mathcal{O}}[L_{\mathrm{i}}(\boldsymbol{x}, \boldsymbol{\omega})] = \mathbb{E}_{\mathcal{O}}\big[L_{\mathrm{o}}(\boldsymbol{r}^*_{\boldsymbol{x},\boldsymbol{\omega}}, -\boldsymbol{\omega})\big] \qquad (4)$$

$$= \int_0^\infty \underbrace{\mathrm{p}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}(t)}_{\text{geometry}} \cdot \underbrace{\mathbb{E}_{\mathcal{O}}[L_{\mathrm{o}}(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t), -\boldsymbol{\omega}) \,|\, t]}_{\text{global illumination}} \; \mathrm{d}t. \quad (5)$$

If we drop the distinction between expected and actual radiances, Equation (5) is the *volume rendering equation* that neural volume rendering techniques use [36]. Such techniques typically separately model the geometry and global illumination terms, the latter as either volumetric emission [36] or in-scattered radiance [58]. We focus on the geometry term, but discuss in Appendix B implications for the global illumination term from geometry modeling choices.

Exponential transport. Most commonly in computer vision and graphics, the free-flight distance is an *exponential random variable*, an assumption we call *exponential transport*. Then, Equations (1)–(3) and transmittance reciprocity imply:

$$\mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \exp\left(-\int_0^t \sigma(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(s), \boldsymbol{\omega}) \,\mathrm{d}s\right), \qquad (6)$$

$$\mathrm{p}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \sigma(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t), \boldsymbol{\omega}) \, \mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}(t), \qquad (7)$$

$$\sigma(\boldsymbol{x}, \boldsymbol{\omega}) = \sigma(\boldsymbol{x}, -\boldsymbol{\omega}). \qquad (8)$$

Thus, the attenuation coefficient becomes the *rate parameter* of the free-flight distance. Given known coefficient values, there exist efficient and accurate numerical approximations for the free-flight distribution and transmittance [18, 27, 43].

Exponential transport has been extensively studied for stochastic microparticle geometry [38]: It is equivalent to the *Poisson Boolean model* of stochastic geometry, where microparticle locations are independent [9, 15, 26] and distributed as a spatial Poisson process [12, 31]. This model allows expressing the attenuation coefficient analytically as a function of the probability distributions for the particle location, size, material, shape, and orientation [16, 23, 25]. The recent success of exponential transport in neural rendering [36, 45, 63, 66] motivates our study in Section 3, where we derive, for the first time, exponential transport models for stochastic solid geometry. Notably, Vicini et al. [59] suggest using non-exponential transport for stochastic solid geometry, a suggestion we briefly discuss in Section 6.

Isotropic and anisotropic transport. In isotropic transport, the attenuation coefficient is independent of direction, $\sigma(\boldsymbol{x}, \boldsymbol{\omega}) = \sigma(\boldsymbol{x})$; and conversely for anisotropic transport [25]. In stochastic microparticle geometry, isotropic transport models microparticles as rotationally-symmetric scatterers (spheres). We explain isotropic versus anisotropic transport for stochastic solid geometry in Section 3.2.

3. Stochastic opaque solids

We develop our exponential transport theory for stochastic solid geometry in three parts: 1. In Section 3.1, we introduce a stochastic model for solid geometry, prove conditions for exponential transport, and derive expressions for the attenuation coefficient. 2. In Section 3.2, we generalize these expressions to model variable anisotropic behavior. 3. In Section 3.3, we adapt our expressions to implicit-surface geometry representations. Figure 2 summarizes our theory, and the project website includes a video explanation.

3.1. Conditions for exponential transport

To formalize our exponential transport model for stochastic *opaque* solid geometry, we first define an opaque solid.[2]

Definition 2. *We define an* indicator function $\mathrm{I} : \mathbb{R}^3 \to \{0, 1\}$ *as a binary scalar field, and associate with it a* solid $\mathcal{O} \equiv \{\boldsymbol{x} \in \mathbb{R}^3 : \mathrm{I}(\boldsymbol{x}) = 1\}$. *The solid* $\mathcal{O}$ *is* opaque *if and only if, for every point* $\boldsymbol{x} \in \mathcal{O}$ *and direction* $\boldsymbol{\omega} \in \mathcal{S}^2$, *the visibility satisfies* $\mathrm{V}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \delta(t)$.

The definition of opacity implies that no ray of light can reach points inside the solid $\mathcal{O}$. Therefore, our volumetric light transport formulation will exclude refractive surfaces. We can now use Definition 2 to also define a *stochastic solid*.

Definition 3. *When the indicator function* $\mathrm{I}(\boldsymbol{x})$ *is a* random *scalar field, we call the associated solid* $\mathcal{O}$ *a* stochastic solid, *for which we define the* occupancy $\mathrm{o} : \mathbb{R}^3 \to [0, 1]$ *and* vacancy $\mathrm{v} : \mathbb{R}^3 \to [0, 1]$ *as the scalar fields:*

$$\mathrm{o}(\boldsymbol{x}) \equiv \mathrm{Pr}\{\mathrm{I}(\boldsymbol{x}) = 1\}, \qquad (9)$$

$$\mathrm{v}(\boldsymbol{x}) \equiv \mathrm{Pr}\{\mathrm{I}(\boldsymbol{x}) = 0\} = 1 - \mathrm{o}(\boldsymbol{x}). \qquad (10)$$

With this definition, we can interpret probabilities involving $\mathcal{O}$ in Equations (1)–(3) as probabilities over all realizations of the random indicator function I. We will consider the restriction of the indicator function, occupancy, and vacancy on a ray with origin $x \in \mathbb{R}^3$ and direction $\boldsymbol{\omega} \in \mathcal{S}^2$: $\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv \mathrm{I}(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t))$, and analogously for $\mathrm{o}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$ and $\mathrm{v}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$. We can now state our main technical result.

---

[2]We borrow the term *solid* from Koenderink [29].

Theorem 4: Exponential transport in opaque solids

We assume a random indicator function I and associated stochastic opaque solid $\mathcal{O}$. Then, for any ray with origin $\boldsymbol{x} \in \mathbb{R}^3$ and direction $\boldsymbol{\omega} \in \mathcal{S}^2$, the free-flight distribution $\mathrm{p}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}$ is exponential if and only if the restriction of the indicator function on this ray, $\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}$, is a *continuous-time discrete-space Markov process*; that is, it satisfies:

$$\Pr\{\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \,|\, \mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t_n), t_n < t, n = 1, \ldots, N\}$$
$$= \Pr\{\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \,|\, \mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(\max_n t_n)\}. \quad (11)$$

Additionally, the process $\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}$ is reversible and the corresponding transmittance $\mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}$ is reciprocal if and only if the attenuation coefficient σ equals:

$$\sigma_\delta(\boldsymbol{x},\boldsymbol{\omega}) \equiv |\boldsymbol{\omega} \cdot \nabla \log(\mathrm{v}(\boldsymbol{x}))| = \frac{|\boldsymbol{\omega} \cdot \nabla \, \mathrm{v}(\boldsymbol{x})|}{\mathrm{v}(\boldsymbol{x})}. \quad (12)$$

We explain the notation σ_δ in Section 3.2. Expressions for transmittance and free-flight distribution follow from Equations (6)–(7), and the attenuation coefficient satisfies Equation (8), as required for reciprocity. We discuss reversibility and prove Theorem 4 in Appendix F.1.

3.2. Anisotropy

Returning to Equation (12), we can rewrite it as the product:

$$\underset{\sigma_\delta(\boldsymbol{x},\boldsymbol{\omega})}{} = \overset{\equiv \sigma^{\|}(\boldsymbol{x})}{\frac{\|\nabla \, \mathrm{v}(\boldsymbol{x})\|}{\mathrm{v}(\boldsymbol{x})}} \cdot \overset{\equiv \sigma^{\perp}_\delta(x,\omega)}{|\boldsymbol{\omega} \cdot \boldsymbol{n}(\boldsymbol{x})|}, \quad (13)$$

where $\boldsymbol{n}(\boldsymbol{x}) \equiv \nabla \, \mathrm{v}(\boldsymbol{x})/\|\nabla \, \mathrm{v}(\boldsymbol{x})\|$ is the *unit normal* of the level set of v passing through $\boldsymbol{x}$. We compare Equation (13) to the attenuation coefficient expressions for *anisotropic* stochastic microparticle geometry by Jakob et al. [25, Equation (11)] and Heitz et al. [23, Equation (2)]. As in those works, we can decompose the attenuation coefficient as the product of an isotropic *density* $\sigma^{\|}(\boldsymbol{x})$ and an anisotropic *projected area* $\sigma^{\perp}_\delta(\boldsymbol{x},\boldsymbol{\omega})$.[3] Intuitively: 1. The density $\sigma^{\|}(\boldsymbol{x})$ increases as vacancy $\mathrm{v}(\boldsymbol{x})$ decreases; thus the ray termination probability through $\boldsymbol{x}$ increases the more likely $\boldsymbol{x}$ is to be occupied. 2. The projected area $\sigma^{\perp}_\delta(\boldsymbol{x})$ models foreshortening as a ray of direction $\boldsymbol{\omega}$ encounters a surface patch of normal $\boldsymbol{n}(\boldsymbol{x})$; at *grazing angles* ($|\boldsymbol{\omega} \cdot \boldsymbol{n}(\boldsymbol{x})| = 0$) the patch is invisible, whereas at *normal incidence* ($|\boldsymbol{\omega} \cdot \boldsymbol{n}(\boldsymbol{x})| = 1$) it is maximally visible, corresponding to zero and maximal, respectively, ray termination probability.

This anisotropic behavior mimics deterministic surfaces, and thus is suitable for points $\boldsymbol{x}$ likely to lie on the surface of a stochastic opaque solid (i.e., $\mathrm{v}(\boldsymbol{x}) \approx 1/2$). However, points $\boldsymbol{x}$ that are likely inside the solid (i.e., $\mathrm{v}(\boldsymbol{x}) \approx 0$, respectively)

---

[3]The projected area and density should include multiplicative factors $\mathrm{A}_\mathrm{v}(\boldsymbol{x})$ and $1/\mathrm{A}_\mathrm{v}(\boldsymbol{x})$, respectively, where $\mathrm{A}_\mathrm{v}(\boldsymbol{x}) \equiv \|\nabla \, \mathrm{v}(\boldsymbol{x})\| \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z$ is the differential area of the tangent plane of the level set of v at $\boldsymbol{x}$. These factors cancel out in Equation (13), so we omit them to simplify notation, at the cost of our expressions *appearing* to have incorrect units.

should behave *isotropically*: rays passing through them at different directions should terminate with the same probability. To model these different behaviors, inspired from microflake models for microparticle geometry [23, 25], we generalize our definitions of $\sigma^{\perp}_\delta$ and σ_δ.[4]

Definition 5: Attenuation coefficient for opaque solids

We associate with each point $\boldsymbol{x} \in \mathbb{R}^3$ a *distribution of normals* $\mathrm{D}_{\boldsymbol{x}} : \mathcal{S}^2 \to \mathbb{R}_{\geq 0}$ that satisfies $\int_{\mathcal{S}^2} \mathrm{D}_{\boldsymbol{x}}(\boldsymbol{m}) \, \mathrm{d}\boldsymbol{m} = 1$. Then, we define at $\boldsymbol{x}$ the *projected area* for any direction $\boldsymbol{\omega} \in \mathcal{S}^2$ as the expected foreshortening:

$$\sigma^{\perp}_{\mathrm{D}}(\boldsymbol{x},\boldsymbol{\omega}) \equiv \int_{\mathcal{S}^2} |\boldsymbol{\omega} \cdot \boldsymbol{m}| \, \mathrm{D}_{\boldsymbol{x}}(\boldsymbol{m}) \, \mathrm{d}\boldsymbol{m}, \quad (14)$$

the *density* as

$$\sigma^{\|}(\boldsymbol{x}) \equiv \frac{\|\nabla \, \mathrm{v}(\boldsymbol{x})\|}{\mathrm{v}(\boldsymbol{x})}, \quad (15)$$

and the *generalized attenuation coefficient* as the product:

$$\sigma(\boldsymbol{x},\boldsymbol{\omega}) \equiv \sigma^{\|}(\boldsymbol{x}) \cdot \sigma^{\perp}_{\mathrm{D}}(\boldsymbol{x},\boldsymbol{\omega}). \quad (16)$$

For $\mathrm{D}_{\boldsymbol{x},\delta}(\boldsymbol{m}) \equiv \delta(\boldsymbol{m} - \boldsymbol{n}(\boldsymbol{x}))$, the projected area reduces to $|\boldsymbol{\omega} \cdot \boldsymbol{m}|$, explaining the notation $\sigma_\delta, \sigma^{\perp}_\delta$ in Equations (12)–(13). By contrast, for the uniform distribution $\mathrm{D}_{\boldsymbol{x},\mathrm{unif}}(\boldsymbol{m}) \equiv 1/4\pi$, the projected area becomes $\sigma^{\perp}_{\mathrm{unif}}(\boldsymbol{x},\boldsymbol{\omega}) \equiv 1/2$; then both the projected area and attenuation coefficient are *isotropic*. Definition 5 allows behaviors between these two extremes, e.g., by using a linear mixture distribution of normals $\mathrm{D}_{\boldsymbol{x},\mathrm{mix}}(\boldsymbol{m}) \equiv \alpha(\boldsymbol{x}) \, \mathrm{D}_{\boldsymbol{x},\delta}(\boldsymbol{m}) + (1 - \alpha(\boldsymbol{x})) \, \mathrm{D}_{\boldsymbol{x},\mathrm{unif}}(\boldsymbol{m})$ and corresponding projected area:

$$\sigma^{\perp}_{\mathrm{mix}}(\boldsymbol{x},\boldsymbol{\omega}) \equiv \alpha(\boldsymbol{x}) \, \sigma^{\perp}_\delta(\boldsymbol{x},\boldsymbol{\omega}) + (1 - \alpha(\boldsymbol{x})) \, \sigma^{\perp}_{\mathrm{unif}}(\boldsymbol{x},\boldsymbol{\omega})$$
$$= \alpha(\boldsymbol{x})|\boldsymbol{\omega} \cdot \boldsymbol{n}(\boldsymbol{x})| + \frac{1 - \alpha(\boldsymbol{x})}{2}. \quad (17)$$

The *anisotropy parameter* $\alpha(\boldsymbol{x}) \in [0,1]$ continuously interpolates between fully anisotropic ($\alpha(\boldsymbol{x}) = 1$) and fully isotropic ($\alpha(\boldsymbol{x}) = 0$) projected area. Making this parameter spatially varying allows adapting the anisotropy behavior at different parts of an opaque solid, e.g., more anisotropic near its boundary, more isotropic in its interior (Figure 3). We discuss additional choices for the distribution of normals D and associated projected area $\sigma^{\perp}_{\mathrm{D}}$ in Appendices B and E.

3.3. Stochastic implicit surfaces

Definitions 2 and 3 define a (stochastic) solid through the binary indicator function, because the indicator (respectively vacancy) function at a point $\boldsymbol{x}$ is the *minimal* information we need to determine visibility (respectively transmittance) for a ray that passes through $\boldsymbol{x}$. However, it is common practice to define a solid through a non-binary scalar field, which provides richer information about the solid and its surface [47]. We explore this case next.

---

[4]We follow Jakob et al. [25] and use normalized distributions of normals.

3D shape | anisotropy $\alpha(\boldsymbol{x})$ | projected area
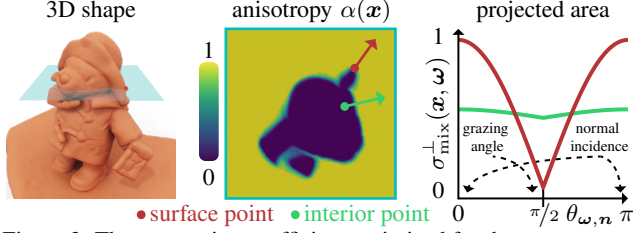
• surface point   • interior point

Figure 3. The attenuation coefficient optimized for the BEAR scene in BlendedMVS behaves as anticipated by our theory: isotropically in the object interior, and anisotropically near its surface.

Definition 6. *We define an* implicit function $\mathrm{G}: \mathbb{R}^3 \to \mathbb{R}$ *as a real scalar field, and associate with it an indicator function* $\mathrm{I}(\boldsymbol{x}) \equiv \mathbf{1}_{\{\mathrm{G}(\boldsymbol{x}) \leq 0\}}$ *and corresponding solid $\mathcal{O}$. If G is also a* random *field, then we define the pointwise* cumulative distribution function $\mathrm{cdf}_{\mathrm{G}(\boldsymbol{x})}$, probability density function $\mathrm{pdf}_{\mathrm{G}(\boldsymbol{x})}$, *and* mean implicit function $\mathrm{f}(\boldsymbol{x})$ *as, respectively:*

$$\mathrm{cdf}_{\mathrm{G}(\boldsymbol{x})}(q) \equiv \Pr\{\mathrm{G}(\boldsymbol{x}) \leq q\}, \; q \in \mathbb{R}, \tag{18}$$

$$\mathrm{pdf}_{\mathrm{G}(\boldsymbol{x})}(q) \equiv \frac{\mathrm{d}\,\mathrm{cdf}_{\mathrm{G}(\boldsymbol{x})}(q)}{\mathrm{d}q}, \; q \in \mathbb{R}, \tag{19}$$

$$\mathrm{f}(\boldsymbol{x}) \equiv \mathbb{E}[\mathrm{G}(\boldsymbol{x})] = \int_{-\infty}^{+\infty} q \cdot \mathrm{pdf}_{\mathrm{G}(\boldsymbol{x})}(q)\,\mathrm{d}q. \tag{20}$$

From Definition 6, the stochastic solid $\mathcal{O}$ is an *excursion set* of the random field G [3, Chapter 1], with its surface at the zero-level set of G, its interior where $\mathrm{G} < 0$, and its exterior elsewhere. Such excursion sets have been extensively studied in applied mathematics, especially when G is a *Gaussian process*, i.e., its (joint) distribution at one or more points is Gaussian [3, Appendix]. Sellán and Jacobson [52, 53] recently proposed using excursion sets of Gaussian processes as a point-based stochastic implicit surface representation (Appendix A). Extending our theory to excursion sets of various stochastic implicit functions G will allow us to provide stochastic geometry interpretations for previous volumetric representations for opaque solids [63, 66].

To this end, we specialize to stochastic implicit functions with a *symmetry* property: At each $\boldsymbol{x}$, $\mathrm{G}(\boldsymbol{x})$ equals, up to a spatially varying shift $\mathrm{f}(\boldsymbol{x})$ and spatially constant scale $s > 0$, a zero-mean, unit-variance, and *symmetric* random variable—that is, with a PDF $\psi : \mathbb{R} \to \mathbb{R}_{\geq 0}$ and CDF $\Psi : \mathbb{R} \to [0, 1]$ that satisfy, for all $q \in \mathbb{R}$ [11],

$$\psi(q) = \psi(-q) \text{ and } \Psi(q) = 1 - \Psi(-q). \tag{21}$$

Such a CDF Ψ is a *sigmoid* function [19, 22] whose exact shape depends on the probability distribution. Common symmetric distributions include the Gaussian, logistic, and Laplace (in their zero-mean, unit-variance versions), giving rise to the error, logistic, and Laplace (respectively) sigmoid functions. The symmetry property implies for G:

$$\mathrm{pdf}_{\mathrm{G}(\boldsymbol{x})}(q) = \psi(s(q - \mathrm{f}(\boldsymbol{x}))), \tag{22}$$

$$\mathrm{cdf}_{\mathrm{G}(\boldsymbol{x})}(q) = \Psi(s(q - \mathrm{f}(\boldsymbol{x}))). \tag{23}$$

We prove in Appendix F.2 the following proposition.

Proposition 7: Stochastic implicit geometry

We assume a stochastic implicit function $\mathrm{G}(\boldsymbol{x})$ satisfying Equations (22)–(23). Then, the occupancy and vacancy for the associated stochastic solid $\mathcal{O}$ equal:

$$\mathrm{o}(\boldsymbol{x}) = \Pr\{\mathrm{G}(\boldsymbol{x}) \leq 0\} = \Psi(-s\,\mathrm{f}(\boldsymbol{x})), \tag{24}$$

$$\mathrm{v}(\boldsymbol{x}) = \Pr\{\mathrm{G}(\boldsymbol{x}) > 0\} = \Psi(s\,\mathrm{f}(\boldsymbol{x})). \tag{25}$$

If the stochastic solid $\mathcal{O}$ also satisfies the conditions of Theorem 4, then the attenuation coefficient equals:

$$\sigma(\boldsymbol{x}, \boldsymbol{\omega}) = \underbrace{\frac{s\,\psi(s\,\mathrm{f}(\boldsymbol{x}))\|\nabla\,\mathrm{f}(\boldsymbol{x})\|}{\Psi(s\,\mathrm{f}(\boldsymbol{x}))}}_{\equiv\,\sigma^{\|}(\boldsymbol{x})} \cdot \sigma_{\mathrm{D}}^{\perp}(\boldsymbol{x}, \boldsymbol{\omega}), \tag{26}$$

with $\sigma_{\mathrm{D}}^{\perp}$ as in Equation (14).

Proposition 7 completes our volumetric representation, which we summarize in Figure 2. Notably, the modeling choice to use an implicit function impacts the density $\sigma^{\|}$ (through the vacancy v), but *not* the projected area $\sigma_{\mathrm{D}}^{\perp}$. To help intuition, we discuss the behavior of different quantities.

Vacancy. The vacancy v (Equation (25)) is a sigmoidal transform of the expected value f of the stochastic implicit function G (Equation (20)). This agrees with intuition: a large positive value of $\mathrm{f}(\boldsymbol{x})$ (i.e., high probability that $\boldsymbol{x}$ is *outside* the solid) results in $\mathrm{v}(\boldsymbol{x}) \approx 1$; a large negative value of $\mathrm{f}(\boldsymbol{x})$ (i.e., high probability that $\boldsymbol{x}$ is *inside* the solid) results in $\mathrm{v}(\boldsymbol{x}) \approx 0$; and $\mathrm{f}(\boldsymbol{x}) = 0$ (i.e., equal probability that $\boldsymbol{x}$ is inside or outside the solid) results in $\mathrm{v}(\boldsymbol{x}) = 1/2$.

Attenuation coefficient and free-flight distribution. The behaviors of the attenuation coefficient σ (Equation (26)) and free-flight distribution p^{ff} (Equation (7)) are easier to understand if we consider points on a ray along which the mean implicit f monotonically decreases with a constant gradient. Then, the attenuation coefficient monotonically increases along the ray (i.e., as we move from points likely in the exterior to points likely in the interior of the solid). The free-flight distribution is maximal where along the ray $\mathrm{f}(\boldsymbol{x}) = 0$ (i.e., at the point equally likely to be inside or outside the solid), and decreases as the magnitude of f increases (i.e., at points highly likely to be inside or outside the solid).

Scale and uncertainty. The scale s controls the width of the sigmoid Ψ, and thus how fast the vacancy v transitions from 0 to 1 as the mean implicit function f increases—the larger s is, the narrower Ψ becomes, and the faster v changes. We can interpret this behavior by noticing that, from Equations (22)–(23), s is the inverse of the pointwise standard deviation of G. As s increases, the standard deviation decreases, and thus: 1. the pointwise PDF $\mathrm{pdf}_{\mathrm{G}}$ becomes more concentrated around its mean f (i.e., the stochastic implicit function G becomes more certain); 2. the vacancy v and occupancy o become closer to the binary functions $\mathbf{1}_{\{\mathrm{f}>0\}}$ and $\mathbf{1}_{\{\mathrm{f}\leq 0\}}$ (i.e., the random indicator function I becomes more certain). 3. the free-flight distribution p^{ff} becomes closer to

Table 1. Classification of previous and new volumetric representations for opaque solids using our theory (Figure 2).

method	implicit function distribution Ψ	distribution of normals D
VolSDF	Laplace	uniform
NeuS	logistic	delta (with ReLU)
NeuS with cosine annealing	logistic	mixture (with ReLU, constant anisotropy)
ours	Gaussian	mixture (with $\boldsymbol{x}$-varying anisotropy)

a Dirac delta function centered at the zero-level set of f (i.e., free-flight distances become more certain).

4. Relationship to prior work

Our theory provides a volumetric representation for opaque solids that permits various probabilistic modeling choices, e.g., selecting a distribution of normals (Equation (14)) and an implicit function distribution (Equation (26)). We use our theory to explain and compare volumetric representations in prior work as versions of our representation corresponding to specific choices for these distributions (Table 1), albeit with critical caveats that our theory addresses.

NeuS. Most closely related to our work is the NeuS volumetric representation by Wang et al. [63, Equation (10)]. Using our notation, their extinction coefficient equals:

$$\sigma_{\mathrm{NeuS}}(\boldsymbol{x},\boldsymbol{\omega}) \equiv$$

$$\frac{s\,\psi_{\mathrm{logistic}}(s\,\mathrm{f}(\boldsymbol{x}))\|\nabla\,\mathrm{f}(\boldsymbol{x})\|}{\Psi_{\mathrm{logistic}}(s\,\mathrm{f}(\boldsymbol{x}))} \cdot \mathrm{ReLU}(-\boldsymbol{\omega}\cdot\boldsymbol{n}(\boldsymbol{x})) , \quad (27)$$

where: 1. ψ_{logistic} and Ψ_{logistic} are the PDF and CDF, respectively, for the zero-mean, unit-variance *logistic distribution*; and 2. the mean implicit function f is parameterized as a neural field that, during training, is also regularized to approximate a signed distance function (i.e., satisfy $\|\nabla\,\mathrm{f}(\boldsymbol{x})\| \approx 1$). Comparing Equation (27) with our Equations (13) and (26), we see that the NeuS model is close to our model with Dirac delta distribution of normals $\mathrm{D}_{\boldsymbol{x},\delta}$, specialized to specific choices for the pointwise distribution ψ and mean implicit function f, with an important difference: it replaces the Dirac delta projected area $\sigma_\delta^\perp(\boldsymbol{x},\boldsymbol{\omega}) \equiv |\boldsymbol{\omega}\cdot\boldsymbol{n}(\boldsymbol{x})|$ with the anisotropic term $\mathrm{ReLU}(-\boldsymbol{\omega}\cdot\boldsymbol{n}(\boldsymbol{x})) \equiv \max(0, -\boldsymbol{\omega}\cdot\boldsymbol{n}(\boldsymbol{x}))$, effectively clipping σ_{NeuS} to zero when a ray is traveling outwards (towards larger vacancy or mean implicit values). Unfortunately, this choice has the consequence that the attenuation coefficient σ_{NeuS} violates the reciprocity requirement of Equation (8), resulting in a physically implausible volumetric representation. We visualize this issue in Figure 4. As we discuss in Section 5, violation of reciprocity additionally negatively impacts 3D reconstruction quality.

Lastly, we prove in Appendix F.3 that using the logistic distribution for $\mathrm{G}(\boldsymbol{x})$ simplifies Equation (26):
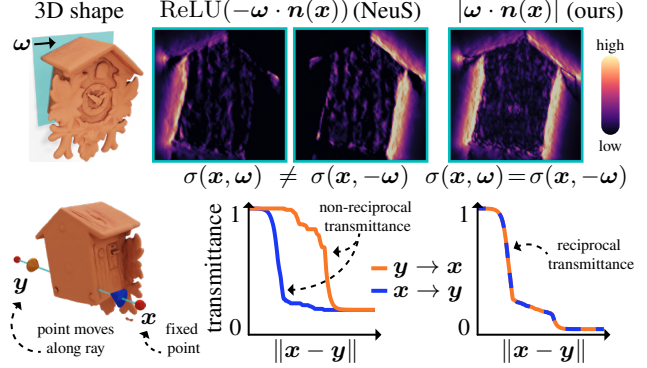

Figure 4. When optimizing for the CLOCK scene in BlendedMVS using NeuS, the ReLU term leads to attenuation coefficient (top) and transmittance (bottom) values that violate reciprocity. By contrast, using our representation leads to reciprocal results.

$$\sigma_{\mathrm{logistic}}(\boldsymbol{x},\boldsymbol{\omega}) =$$

$$s\Psi_{\mathrm{logistic}}(-s\,\mathrm{f}(\boldsymbol{x}))\|\nabla\,\mathrm{f}(\boldsymbol{x})\| \cdot \sigma_{\mathrm{D}}^\perp(\boldsymbol{x},\boldsymbol{\omega}) , \quad (28)$$

and analogously for $\sigma_{\mathrm{NeuS}}(\boldsymbol{x},\boldsymbol{\omega})$ in Equation (27). We use this observation as we discuss VolSDF next.

VolSDF. Another closely related model is the VolSDF model by Yariv et al. [66, Equations (2)–(3)]. Using our notation:

$$\sigma_{\mathrm{VolSDF}}(\boldsymbol{x},\boldsymbol{\omega}) \equiv s\Psi_{\mathrm{Laplace}}(-s\,\mathrm{f}(\boldsymbol{x}))\|\nabla\,\mathrm{f}(\boldsymbol{x})\| , \quad (29)$$

where Ψ_{Laplace} is the CDF for the zero-mean, unit-variance *Laplace distribution*; and the mean implicit function $\mathrm{f}(\boldsymbol{x})$ is parameterized as in NeuS. Comparing to Equation (28), we make two observations about the VolSDF model: 1. It uses the uniform distribution of normals $\mathrm{D}_{\boldsymbol{x},\mathrm{unif}}$ and isotropic (constant) projected area $\sigma_{\mathrm{unif}}^\perp$, thus the attenuation coefficient becomes *isotropic*. 2. It uses a density $\sigma^\|$ equal to that in Equation (28) after replacing the logistic with the Laplace CDF. However, this replacement is *not* equivalent to modeling $\mathrm{G}(\boldsymbol{x})$ as a Laplace random variable. This is because the simplified expression of Equation (28) is correct for only the logistic distribution, whereas the Laplace distribution requires the full expression of Equation (26). Consequently, the VolSDF model uses an incorrect density term.

Cosine annealing. The official NeuS implementation [62, `models/renderer.py#L232-L235`] uses *cosine annealing*—transitioning from isotropy to anisotropy as optimization progresses—to improve convergence. This means replacing the anisotropic term in Equation (27) with:

$$\alpha \cdot \mathrm{ReLU}(-\boldsymbol{\omega}\cdot\boldsymbol{n}(\boldsymbol{x})) + \frac{1-\alpha}{2} , \quad (30)$$

and changing the *global* anisotropy parameter α from 0 towards 1 using a predetermined annealing schedule. Comparing to Equation (17), our theory explains this heuristic as using a mixture distribution of normals, with the caveat that NeuS also replaces $\sigma_\delta^\perp$ with the reciprocity-violating

Table 2. Chamfer distance statistics on the DTU and NeRF Realistic Synthetic datasets. We provide the full tables in Appendix E.

DTU	VolSDF	NeuS	**ours**
mean	1.84	2.17	**1.57**
median	1.74	1.99	**1.56**
NeRF RS	VolSDF	NeuS	**ours**
mean	0.252	0.201	**0.113**
median	0.100	0.085	**0.057**

Table 3. We use chamfer distance statistics on the DTU dataset for an ablation study. We provide the full tables in Appendix E.

Ψ model	logistic	Laplace	**Gaussian**	
mean	1.98	1.96	**1.78**	
median	1.86	1.92	**1.74**	
D model	delta (ReLU)	delta	mixture (const.)	**mixture (var.)**
mean	2.17	1.98	1.97	**1.75**
median	1.99	1.86	1.85	**1.59**

Figure 5. Visualization of shape and key quantities of our volumetric representation learned for scenes in the BlendedMVS dataset.

ReLU term, as we noted earlier. Additionally, NeuS uses a spatially constant anisotropy parameter α that cannot capture the qualitatively different foreshortening behavior of surface versus interior and exterior points (Section 5).

Scale optimization and adaptive shells. NeuS and VolSDF optimize the scale s, which typically increases as optimization progresses. Our theory explains this behavior as decreasing uncertainty of the stochastic geometry and convergence towards deterministic geometry (binary vacancy).

The *adaptive shells* representation by Wang et al. [64] modifies the NeuS representation to use a spatially varying scale $s(\boldsymbol{x})$. Our theory explains this choice as spatially varying pointwise standard deviation, and thus uncertainty, for the stochastic implicit function $G(\boldsymbol{x})$. In Appendix F.2 we explain how to modify the density $\sigma^{\parallel}$ in Equation (26) to account for spatially varying scale $s(\boldsymbol{x})$.

Other approaches. In the appendix we discuss other approaches for modeling stochastic solid geometry, such as occupancy networks [35, 40] and discretization approaches [8, 45, 55] in Appendix C, as well as stochastic implicit surfaces using point clouds [52] in Appendix A.

5. Experimental evaluation

Our theory provides a framework for systematically *correcting* volumetric representations for opaque solids in prior work (using reciprocal projected area and correct density), and *designing* new representations (using different distributions for the implicit function and normals). These changes are straightforward to implement within existing neural rendering pipelines. We do so within a simplified version of NeuS [63] and perform extensive experiments on multi-view 3D reconstruction tasks. The goal of these experiments is not state-of-the-art performance, but an exploration of the
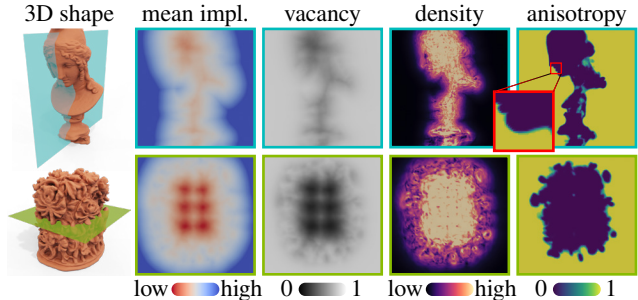
design space and an evaluation of volumetric representations within an equal framework. Our results show that, despite their simple nature, the changes our theory suggests greatly improve performance, qualitatively and quantitatively. Overall, our experiments demonstrate the utility of a rigorous theory of volumetric representations for opaque solids.

In this section, we summarize our experiments and findings. We discuss implementation details and list complete numerical results in Appendix E. Lastly, we provide interactive visualizations and code on the project website.

Comparison to prior work. We evaluate our best performing volumetric representation against those of NeuS [63] and VolSDF [66]. Table 1 summarizes the three representations. We use three datasets for evaluation: DTU [2], BlendedMVS [65], and NeRF realistic synthetic (NeRF RS) [36].

Table 2 and Figure 6 show quantitative and qualitative results. We observe the following: 1. Our representation performs the best across all datasets, both quantitatively and qualitatively. Qualitatively, the improvements are more pronounced in BlendedMVS and NeRF RS—whose scenes have more complex geometry, materials, and background—than in DTU—whose simpler scenes are reconstructed well by all representations. 2. Our representation learns meaningful scalar fields (Figure 5) for the mean implicit function f, vacancy v, density $\sigma^{\parallel}$, and anisotropy α. The mean implicit function and vacancy lend themselves to downstream surface processing tasks (e.g., mesh extraction [33]). 3. The use of a spatially varying anisotropy α allows our representation to model the qualitatively different behaviors of points $\boldsymbol{x}$ on the surface ($\alpha(\boldsymbol{x}) \approx 1$, i.e., strongly anisotropic) versus in the interior ($\alpha(\boldsymbol{x}) \approx 0$, i.e., isotropic). By contrast, VolSDF and NeuS require *all* points $\boldsymbol{x}$—surface or interior—to have either isotropic ($\alpha(\boldsymbol{x}) := 0$) or perfectly anisotropic ($\alpha(\boldsymbol{x}) := 1$), respectively, behavior.

Design and ablation. We use the DTU dataset to evaluate design choices for the implicit function distribution Ψ and distribution of normals D in Equations (14) and (26), respectively. This evaluation also serves as an ablation study for our best performing representation in Table 1. Table 3 shows the results. We observe the following: 1. For Ψ, using the Gaussian distribution (i.e., a Gaussian process [52]) outperforms using the Laplace (as in VolSDF) or logistic (as in
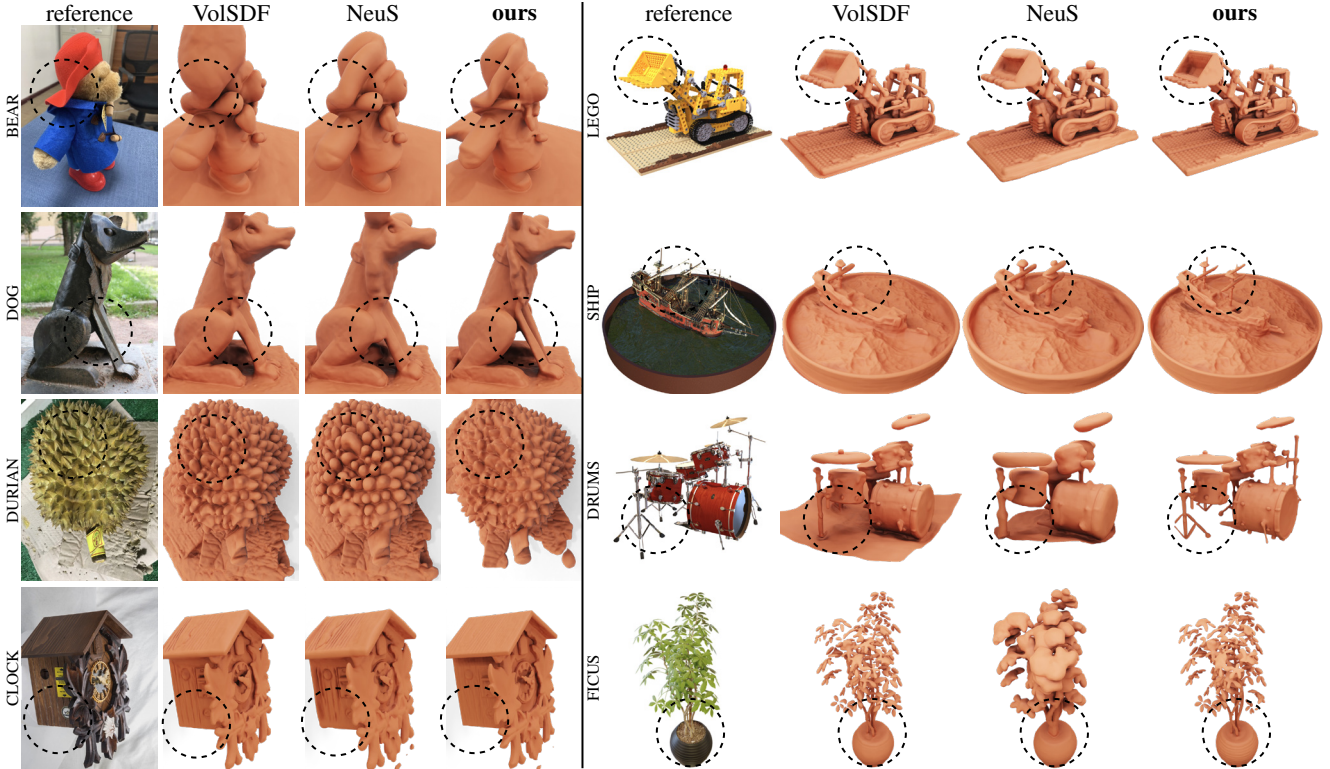
Figure 6. Qualitative comparisons on the BlendedMVS (left) and NeRF Realistic Synthetic (right) datasets. The dashed circles indicate areas of interest. We provide interactive visualizations of results on the complete datasets on the project website.

NeuS) distributions. 2. For D, using the spatially varying mixture distribution (Equation (17), to adapt to surface versus interior points) outperforms using the spatially constant mixture distribution (as in cosine annealing) or the delta distribution. 3. When using the delta distribution, including a reciprocity-violating ReLU term (Equation (27), as in NeuS) underperforms not doing so. This result highlights that enforcing reciprocity not only ensures physical plausibility, but also improves reconstruction performance.

6. Conclusion

We have taken first steps towards formalizing volumetric representations for opaque solids, using stochastic geometry theory. Our results should be of theoretical and practical interest: On the theory side, they help explain why volumetric neural rendering can reconstruct solid geometry, and justify previous volumetric representations for it. On the practice side, they provide a toolbox for the design of physically-plausible volumetric representations that greatly improve performance. We hope that our results will motivate further research along both theory and practice thrusts.

For the theory thrust, our theory is far from a complete formalism of volumetric representations for solid geometry. We highlight three important shortcomings that deserve further investigation. First, revisiting Equation (5), we focused on the *geometry* of opaque solids, but neglected their *global illumination* effects. In Appendix B, we briefly examine how geometry and global illumination must be coupled to ensure reciprocity. However, this topic requires further investigation. Second, we excluded *(semi-)transparent* solids, where interior points may be visible to each other (violating Definition 2). Developing volumetric representations for such solids will allow modeling complex reflective-refractive appearance. Third, we focused on *exponential* transport because it has served as a convenient approximation for most prior work. However, both empirical evidence [59] and stochastic geometry theory suggest that exponentiality may not be a suitable assumption for opaque solids. Indeed, the excursion sets of Gaussian processes typically have free-flight distributions (also known as the *first-passage times* of the Gaussian processes) that are *non-exponential* [1]. As Sellán and Jacobson [53, Section 5.1] point out, characterizing these distributions requires reasoning about the *spatial covariance* structure of the Gaussian process [3, Appendix].

For the practice thrust, we have done only limited evaluation of how different volumetric representations interplay with different algorithms for free-flight estimation and sampling (Appendix D). The theoretical investigation and empirical evaluation of such algorithms will be critical for optimizing volumetric neural rendering pipelines for solid geometry.

References

[1] Odd O Aalen and Håkon K Gjessing. Understanding the shape of the hazard rate: A process point of view (with comments and a rejoinder by the authors). *Statistical Science*, 16(1): 1–22, 2001. 8

[2] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, pages 1–16, 2016. 7

[3] Robert J Adler. *The geometry of random fields*. SIAM, 2010. 5, 8

[4] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5

[5] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6290–6301, 2022. 1

[6] Chen Bar, Marina Alterman, Ioannis Gkioulekas, and Anat Levin. A Monte Carlo framework for rendering speckle statistics in scattering media. *ACM Transactions on Graphics (TOG)*, 38(4):1–22, 2019. 1, 2

[7] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. 3, 4

[8] Rahul Bhotika, David J Fleet, and Kiriakos N Kutulakos. A probabilistic theory of occupancy and emptiness. In *Computer Vision—ECCV 2002: 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part III 7*, pages 112–130. Springer, 2002. 7, 3, 4

[9] Benedikt Bitterli, Srinath Ravichandran, Thomas Müller, Magnus Wrenninge, Jan Novák, Steve Marschner, and Wojciech Jarosz. A radiative transfer framework for nonexponential media. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, 37(6):225:1–225:17, 2018. 1, 2, 3, 6

[10] James F Blinn. Models of light reflection for computer synthesized pictures. In *Proceedings of the 4th annual conference on Computer graphics and interactive techniques*, pages 192–198, 1977. 2

[11] George Casella and Roger L Berger. *Statistical inference*. Cengage Learning, 2021. 5

[12] Sung Nok Chiu, Dietrich Stoyan, Wilfrid S Kendall, and Joseph Mecke. *Stochastic geometry and its applications*. John Wiley & Sons, 2013. 1, 2, 3

[13] Robert L Cook and Kenneth E. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics (ToG)*, 1(1):7–24, 1982. 1

[14] Jonathan Dupuy, Eric Heitz, and Eugene d'Eon. Additional progress towards the unification of microfacet and microflake theories. In *Proceedings of the Eurographics Symposium on Rendering: Experimental Ideas & Implementations*, page 55–63, 2016. 1

[15] Eugene d'Eon. A reciprocal formulation of nonexponential radiative transfer. 1: Sketch and motivation. *Journal of Computational and Theoretical Transport*, 47(1-3):84–115, 2018. 1, 3

[16] Jeppe Revall Frisvad, Niels Jørgen Christensen, and Henrik Wann Jensen. Computing the scattering properties of participating media using lorenz-mie theory. In *ACM SIGGRAPH 2007 papers*, pages 60–es. 2007. 1, 3

[17] Qiancheng Fu, Qingshan Xu, Yew-Soon Ong, and Wenbing Tao. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. In *Advances in Neural Information Processing Systems*, 2022. 1

[18] Iliyan Georgiev, Zackary Misso, Toshiya Hachisuka, Derek Nowrouzezahrai, Jaroslav Křivánek, and Wojciech Jarosz. Integral formulations of volumetric transmittance. *ACM Transactions on Graphics (TOG)*, 38(6):1–17, 2019. 3, 4

[19] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings, 2011. 5

[20] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *Proceedings of Machine Learning and Systems 2020*, pages 3569–3579. 2020. 5

[21] Charles Han, Bo Sun, Ravi Ramamoorthi, and Eitan Grinspun. Frequency domain normal map filtering. In *ACM SIGGRAPH 2007 papers*, pages 28–es. 2007. 2

[22] Jun Han and Claudio Moraga. The influence of the sigmoid function parameters on the speed of backpropagation learning. In *International workshop on artificial neural networks*, pages 195–201. Springer, 1995. 5

[23] Eric Heitz, Jonathan Dupuy, Cyril Crassin, and Carsten Dachsbacher. The sggx microflake distribution. 34(4), 2015. 1, 3, 4, 5

[24] Eric Heitz, Johannes Hanika, Eugene d'Eon, and Carsten Dachsbacher. Multiple-scattering microfacet bsdfs with the smith model. *ACM Transactions on Graphics (TOG)*, 35(4): 1–14, 2016. 1

[25] Wenzel Jakob, Adam Arbree, Jonathan T. Moon, Kavita Bala, and Steve Marschner. A radiative transfer framework for rendering materials with anisotropic structure. *ACM Trans. Graph.*, 29(4):53:1–53:13, 2010. 1, 3, 4

[26] Adrian Jarabo, Carlos Aliaga, and Diego Gutierrez. A radiative transfer framework for spatially-correlated materials. *ACM Transactions on Graphics (TOG)*, 37(4):1–13, 2018. 1, 3

[27] Markus Kettunen, Eugene D'Eon, Jacopo Pantaleoni, and Jan Novák. An unbiased ray-marching transmittance estimator. *ACM Trans. Graph.*, 40(4), 2021. 3, 4

[28] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[29] Jan J Koenderink. *Solid shape*. MIT press, 1990. 3

[30] Peter Kutz, Ralf Habel, Yining Karl Li, and Jan Novák. Spectral and decomposition tracking for rendering heterogeneous

volumes. *ACM Transactions on Graphics (TOG)*, 36(4):1–16, 2017. 4

[31] Günter Last and Mathew Penrose. *Lectures on the Poisson process*. Cambridge University Press, 2017. 3

[32] Aviad Levis, Yoav Y Schechner, and Anthony B Davis. Multiple-scattering microphysics tomography. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6740–6749, 2017. 1

[33] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, page 163–169, 1987. 7

[34] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995. 4

[35] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 7, 3

[36] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 3, 7, 4, 5

[37] Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, Peter Hedman, Ricardo Martin-Brualla, and Jonathan T. Barron. Multi-NeRF: A Code Release for Mip-NeRF 360, Ref-NeRF, and RawNeRF, 2022. 2

[38] Michael I Mishchenko, Larry D Travis, and Andrew A Lacis. *Multiple scattering of light by particles: radiative transfer and coherent backscattering*. Cambridge University Press, 2006. 1, 3

[39] Jonathan T Moon, Bruce Walter, and Stephen R Marschner. Rendering discrete random media using precomputed scattering solutions. In *Proceedings of the 18th Eurographics conference on Rendering Techniques*, pages 231–242, 2007. 2

[40] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020. 7, 3

[41] Merlin Nimier-David, Thomas Müller, Alexander Keller, and Wenzel Jakob. Unbiased inverse volume rendering with differential trackers. *ACM Transactions on Graphics (TOG)*, 41(4):1–20, 2022. 4

[42] James R Norris. *Markov chains*. Number 2. Cambridge university press, 1998. 3, 6, 7, 8

[43] Jan Novák, Andrew Selle, and Wojciech Jarosz. Residual ratio tracking for estimating attenuation in participating media. *ACM Trans. Graph.*, 33(6):179–1, 2014. 3, 4

[44] Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. Monte carlo methods for volumetric light transport simulation. In *Computer Graphics Forum*, pages 551–576. Wiley Online Library, 2018. 1

[45] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. 1, 3, 7, 4

[46] Michael Oren and Shree K Nayar. Diffuse reflectance from rough surfaces. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 763–764. IEEE, 1993. 1, 2

[47] Stanley Osher and Ronald P Fedkiw. *Level set methods and dynamic implicit surfaces*. Springer New York, 2005. 4

[48] Rudolph W Preisendorfer. *Radiative transfer on discrete spaces*. Elsevier, 2014. 1, 2

[49] Ravi Ramamoorthi and Pat Hanrahan. A signal-processing framework for inverse rendering. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 117–128, 2001. 2

[50] Konstantinos Rematas, An Liu, Pratul P. Srinivasan, Jonathan T. Barron, Andrea Tagliasacchi, Thomas A. Funkhouser, and Vittorio Ferrari. Urban radiance fields. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12922–12932, 2021. 1

[51] Tim Salimans and Durk P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. *Advances in neural information processing systems*, 29, 2016. 5

[52] Silvia Sellán and Alec Jacobson. Stochastic Poisson surface reconstruction. *ACM Transactions on Graphics (TOG)*, 41 (6):1–12, 2022. 1, 5, 7, 2, 9

[53] Silvia Sellán and Alec Jacobson. Neural stochastic poisson surface reconstruction. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–9, 2023. 5, 8

[54] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David McAllister, and Angjoo Kanazawa. Nerfstudio: A modular framework for neural radiance field development. *arXiv preprint arXiv:2302.04264*, 2023. 2

[55] Shubham Tulsiani, Tinghui Zhou, Alexei A Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2626–2634, 2017. 7, 3, 4

[56] Ali Osman Ulusoy, Andreas Geiger, and Michael J Black. Towards probabilistic volumetric reconstruction using ray potentials. In *2015 International Conference on 3D Vision*, pages 10–18. IEEE, 2015. 4

[57] Eric Veach. *Robust Monte Carlo methods for light transport simulation*. Stanford University, 1998. 1

[58] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5481–5490. IEEE, 2022. 3, 2, 4

[59] Delio Vicini, Wenzel Jakob, and Anton Kaplanyan. A non-exponential transmittance model for volumetric scene rep-

resentations. *ACM Transactions on Graphics (TOG)*, 40(4): 1–16, 2021. 3, 8

[60] Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics conference on Rendering Techniques*, pages 195–206, 2007. 1, 2

[61] Jiepeng Wang, Peng Wang, Xiaoxiao Long, Christian Theobalt, Taku Komura, Lingjie Liu, and Wenping Wang. Neuris: Neural reconstruction of indoor scenes using normal priors. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, page 139–155, Berlin, Heidelberg, 2022. Springer-Verlag. 1

[62] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus codebase. https://github.com/Totoro97/NeuS, 2021. 6

[63] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 2, 3, 5, 6, 7, 4

[64] Zian Wang, Tianchang Shen, Merlin Nimier-David, Nicholas Sharp, Jun Gao, Alexander Keller, Sanja Fidler, Thomas Müller, and Zan Gojcic. Adaptive shells for efficient neural radiance field rendering. 42(6), 2023. 7, 9

[65] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. *Computer Vision and Pattern Recognition (CVPR)*, 2020. 7

[66] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815, 2021. 1, 2, 3, 5, 6, 7, 4

[67] Zehao Yu, Anpei Chen, Bozidar Antic, Songyou Peng Peng, Apratim Bhattacharyya, Michael Niemeyer, Siyu Tang, Torsten Sattler, and Andreas Geiger. Sdfstudio: A unified framework for surface reconstruction, 2022. 2

[68] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. MonoSDF: Exploring monocular geometric cues for neural implicit surface reconstruction. In *Advances in Neural Information Processing Systems*, 2022. 1

[69] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv:2010.07492*, 2020. 5

Objects as volumes: A stochastic geometry view of opaque solids

Supplementary Material

A. Volumetric representation of point clouds

In this section, we discuss how our theory relates to prior work on modeling point clouds, and how their combination enables volume rendering of point clouds.

Stochastic Poisson surface reconstruction (SPSR). Sellán and Jacobson [52] propose a stochastic model for opaque solids as excursion sets of Gaussian processes conditioned on point clouds. Using our notation, their model corresponds to Equations (24)–(25) as

$$\mathrm{v}_{\mathrm{SPSR}}(\boldsymbol{x}) = 1 - \mathrm{o}_{\mathrm{SPSR}}(\boldsymbol{x}) \equiv \Psi_{\mathrm{Gaussian}}(s(\boldsymbol{x})\,\mathrm{f}(\boldsymbol{x})), \quad (31)$$

where Ψ_{Gaussian} is the CDF for the zero-mean, unit-variance Gaussian distribution; and the mean implicit function $\mathrm{f}(\boldsymbol{x})$ and scale $s(\boldsymbol{x})$ are parameterized as functionals of a point cloud. For volume rendering, they use the occupancy $\mathrm{o}_{\mathrm{SPSR}}$ in place of the attenuation coefficient σ [52, Section 8.2]. Our theory disambiguates the occupancy and corresponding attenuation coefficient, which from Equation (16) (assuming perfectly anisotropic projected area) becomes:

$$\sigma_{\mathrm{SPSR}}(\boldsymbol{x}, \boldsymbol{\omega}) \equiv \frac{\|\nabla \mathrm{v}_{\mathrm{SPSR}}(\boldsymbol{x})\|}{\mathrm{v}_{\mathrm{SPSR}}(\boldsymbol{x})} \cdot |\boldsymbol{\omega} \cdot \boldsymbol{n}(\boldsymbol{x})|. \quad (32)$$

Volume rendering of point clouds. To highlight the importance of disambiguating vacancy and the attenuation coefficient, we use single-bounce volume rendering to synthesize images of a point cloud scene from SPSR [52] under point light illumination. In Figure 7, we compare the images resulting from using the attenuation coefficient our theory predicts in Equation (32), versus using the vacancy in Equation (31) in place of the attenuation coefficient. Using the attenuation coefficient results in images that more accurately convey the appearance of the opaque solid underling the point cloud.

We use the same scene for an additional experiment that highlights the importance of enforcing reciprocity in volumetric representations of opaque solids. We consider variant of Equation (32) that replaces the perfectly anisotropic projected area with the ReLU-based one from NeuS [63]:

$$\sigma_{\mathrm{SPSR,NeuS}}(\boldsymbol{x}, \boldsymbol{\omega}) \equiv$$

$$\frac{\|\nabla \mathrm{v}_{\mathrm{SPSR}}(\boldsymbol{x})\|}{\mathrm{v}_{\mathrm{SPSR}}(\boldsymbol{x})} \cdot \mathrm{ReLU}(-\boldsymbol{\omega} \cdot \boldsymbol{n}(\boldsymbol{x})). \quad (33)$$

We use the attenuation coefficients of Equations (32) and (33) to synthesize images with two volume rendering algorithms: volumetric path tracing (trace rays starting from the camera) and volumetric light tracing (trace rays starting from the light source) [57]. Under a physically-plausible, reciprocal volumetric representation, both algorithms should produce identical images. We show in Figure 7 that using our reciprocal attenuation coefficient in Equation (32) produces images that are a lot closer to each other (up to discretization and noise artifacts) than those using the non-reciprocal attenuation coefficient in Equation (33) that NeuS suggests.

B. Global illumination representation

We briefly discuss the global illumination term in the volume rendering equation (5). In *deterministic* light transport algorithms, global illumination appears as the outgoing radiance in the conservation of radiance equation. From the *reflection equation*, the outgoing radiance equals:

$$L_{\mathrm{o}}(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{o}}) =$$

$$\int_{\mathcal{H}^2(\boldsymbol{x})} f_{\mathrm{r}}(\boldsymbol{x}, \boldsymbol{n}(\boldsymbol{x}), \boldsymbol{\omega}_{\mathrm{o}}, \boldsymbol{\omega}_{\mathrm{i}}) L_{\mathrm{i}}(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{i}}) \langle \boldsymbol{\omega}_{\mathrm{i}}, \boldsymbol{n}(\boldsymbol{x}) \rangle \, \mathrm{d}\boldsymbol{\omega}_{\mathrm{i}}, \quad (34)$$

where $\mathcal{H}^2(\boldsymbol{x}) \equiv \{\boldsymbol{\omega} \in \mathcal{S}^2 : \boldsymbol{\omega} \cdot \boldsymbol{n}_{\boldsymbol{x}} \geq 0\}$ is the positive hemisphere and f_{r} is the bidirectional distribution reflection function (BRDF) at point $\boldsymbol{x}$. In *volumetric* light transport algorithms, global illumination appears as the conditional expectation of the outgoing radiance in the volume rendering equation (5). Dropping the distinction between expected and actual radiance to ease notation, from the *in-scattering equation* [23, Section 3.1], the outgoing radiance equals:

$$L_{\mathrm{o}}(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{o}}) = a(\boldsymbol{x}) \int_{\mathcal{S}^2} f_{\mathrm{p}}(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{o}}, \boldsymbol{\omega}_{\mathrm{i}}) L_{\mathrm{i}}(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{i}}) \, \mathrm{d}\boldsymbol{\omega}_{\mathrm{i}}, \quad (35)$$

where $a \in [0, 1]$ is the albedo and f_{p} is the phase function at point $\boldsymbol{x}$. Jakob et al. [25, Section 3] prove that reciprocal *exponential* transport requires that the attenuation coefficient σ and phase function f_{p} satisfy for all directions $\boldsymbol{\omega}_{\mathrm{o}}, \boldsymbol{\omega}_{\mathrm{i}}$:

$$\sigma(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{o}}) f_{\mathrm{p}}(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{o}}, \boldsymbol{\omega}_{\mathrm{i}}) = \sigma(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{i}}) f_{\mathrm{p}}(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{i}}, \boldsymbol{\omega}_{\mathrm{o}}). \quad (36)$$

Lastly, Heitz et al. [23, Section 5] show that, when the attenuation coefficient has the form of Equation (16) in Definition 5, the phase function f_{p} relates to the BRDF f_{r} of the underlying deterministic geometry as:

$$f_{\mathrm{p}}(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{o}}, \boldsymbol{\omega}_{\mathrm{i}}) = \frac{1}{\sigma_{\mathrm{D}}^{\perp}(\boldsymbol{x}, \boldsymbol{\omega}_{\mathrm{o}})}$$

$$\cdot \int_{\mathcal{S}^2} f_{\mathrm{r}}(\boldsymbol{x}, \boldsymbol{m}, \boldsymbol{\omega}_{\mathrm{o}}, \boldsymbol{\omega}_{\mathrm{i}}) |\boldsymbol{\omega}_{\mathrm{o}} \cdot \boldsymbol{m}| \, \mathrm{D}_{\boldsymbol{x}}(\boldsymbol{m}) \, \mathrm{d}\boldsymbol{m}. \quad (37)$$

Equation (37) expresses the phase function as the expected value of the *foreshortened* BRDF with respect to the distribution of normals D used also to define the projected area $\sigma_{\mathrm{D}}^{\perp}$ in Equation (14). We make three observations: 1. The phase function in Equation (37) satisfies the reciprocity relation in Equation (36). 2. Heitz et al. [23] derived Equation (37) in the context of microflake models for stochastic
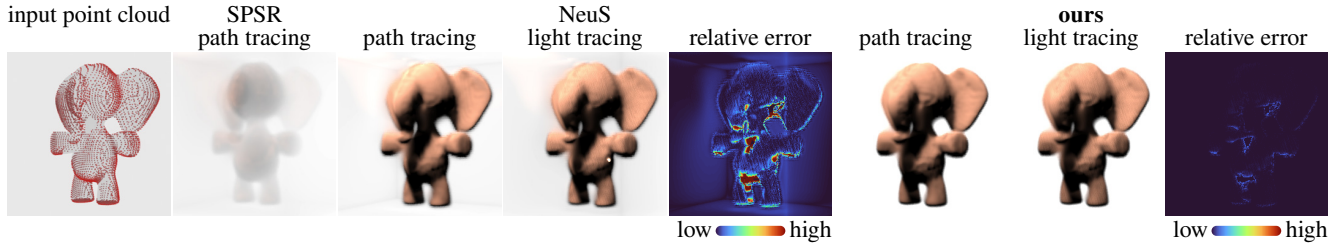
| input point cloud | SPSR path tracing | path tracing | NeuS light tracing | relative error | path tracing | **ours** light tracing | relative error |

Figure 7. Comparison of point-cloud volume renderings on a scene from SPSR [52]. Using the vacancy v_{SPSR} in place of the attenuation coefficient as SPSR suggests results in images that do not convey the appearance of an opaque solid. Using a non-reciprocal attenuation coefficient $\sigma_{\text{SPSR,NeuS}}$ as NeuS suggests results in significant differences between images from different rendering algorithms. Using the reciprocal attenuation coefficient σ_{SPSR} our theory suggests results in accurate images under both rendering algorithms.

microparticle geometry. The similarity between the attenuation coefficient in our theory (Definition 5) and that in microflake models allows us to apply Equation (37) also for stochastic solid geometry. 3. As in the microflake model for stochastic microparticle geometry, we can use different BRDFs f_r in Equation (37) (e.g., specular or Lambertian) to derive corresponding phase functions for the stochastic solid geometry. Then, different distribution of normals D correspond to different levels of "roughness" of the underlying geometry, in analogy with the distribution of normals in microfacet BRDF models (e.g., GGX [60] or Oren-Nayar [46] models for rough specular or Lambertian, respectively, reflectance). However, reciprocity requires using the same distribution of normals for both the phase function and attenuation coefficient, and thus jointly constrains geometry and global illumination in the volume rendering equation (5).

Integrated directional encoding. Neural rendering pipelines typically represent the global illumination in Equation (5) as a neural field with positional and directional inputs. This representation makes it difficult to exactly enforce Equations (35) and (37). Verbin et al. [58] showed how to approximately do so with an *integrated directional encoding* that computes moments of directional inputs with respect to the distribution of normals describing surface roughness. Equation (37) suggests that we can combine our volumetric representation with integrated directional encoding, by using the same distribution of normals D for both.[5] This approach links the geometry and global illumination terms in the volume rendering equation (5), as reciprocity requires.

In particular, Verbin et al. [58] use a distribution of normals D_x that equals the von-Mises Fisher distribution with center at the normal $n(x)$ and concentration parameter $\kappa(x) \equiv \alpha(x)/1-\alpha(x)$, where $\alpha(x) \in [0,1]$ is a spatially

varying anisotropy parameter.[6] Using Equation (14), the corresponding projected area equals:

$$\sigma_{\text{vMF}}^{\perp}(\boldsymbol{x},\boldsymbol{\omega}) \approx \frac{\alpha(\boldsymbol{x})+1}{2}|\boldsymbol{\omega}\cdot\boldsymbol{n}(\boldsymbol{x})|^{\alpha(\boldsymbol{x})} . \quad (38)$$

The proof of Equation (38) is exactly analogous to that by Han et al. [21, Section 7.2], and uses the same approximations as in the derivation of the equations for the integrated directional encoding [58, Equation (8), Appendix A]. It is worth comparing $\sigma_{\text{vMF}}^{\perp}$ in Equation (38) to $\sigma_{\text{mix}}^{\perp}$ and $\sigma_{\text{SGGX}}^{\perp}$ in Equations (17) and (45): all three projected area terms become fully anisotropic and isotropic at the limit values $\alpha = 1$ and $\alpha = 0$, respectively, and produce different interpolations between these extremes at intermediate values.

Novel-view synthesis experiments. We demonstrate the importance of linking representations for geometry and global illumination in the volume rendering equation Equation (5) through *proof-of-concept* experiments on novel-view synthesis. We use Ref-NeRF [58] as a baseline, and modify its architecture to use our attenuation coefficient in Equation (26) with projected area as in Equation (38) (Figure 8). Importantly, our modification uses the same roughness ρ (equivalently, anisotropy α) for both the projected area and integrated directional encoding. The qualitative and quantitative results in Figure 9 and Tables 4 and 5 show that our modifications improve normal and appearance predictions.

Implementation details. We briefly discuss important implementation details for our multi-view synthesis experiments. We build on the Ref-NeRF implementation in the Multi-NeRF codebase [37]. We modify the normal loss to compare with the level-set normals of the vacancy, $\nabla v/\|\nabla v\|$, instead of the attenuation coefficient, $\nabla \sigma/\|\nabla \sigma\|$—our theory shows that the two are *not* equal (Equation (16)) and the former should better represent the underlying geometry.

Multi-NeRF takes advantage of the large memory capacity of TPUs and uses large batch sizes during training. We had access to only GPUs with 16 GB of memory, so we had to reduce the batch size from the default 2^{14} to 2^9 rays per batch. We also followed a suggestion in the Github reposi-

---

[5]A caveat to this approach is that, whereas our Definition 5 and Equation (37) use D as a distribution of *normals*, the integrated directional encoding of Verbin et al. [58] uses it as a distribution of *reflected directions*. This discrepancy is analogous to the difference between the Blinn-Phong and Phong BRDFs, which randomize normals (or equivalently, half-way directions) versus reflected directions, respectively [10]. Ramamoorthi and Hanrahan [49, Appendix] show that, for the von-Mises Fisher distribution that Equation (38) and the integrated directional encoding use, this discrepancy introduces only a small approximation error.

[6]Verbin et al. [58] parameterize the von-Mises Fisher distribution using an *unbounded* roughness parameter $\rho \in [0,\infty)$, which we replace with our *bounded* anisotropy parameter $\alpha \equiv 1/\rho(\boldsymbol{x})+1$.
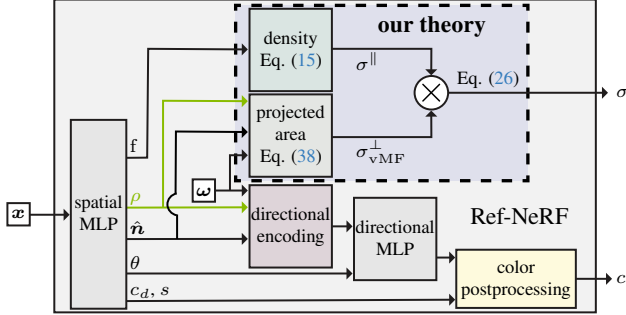
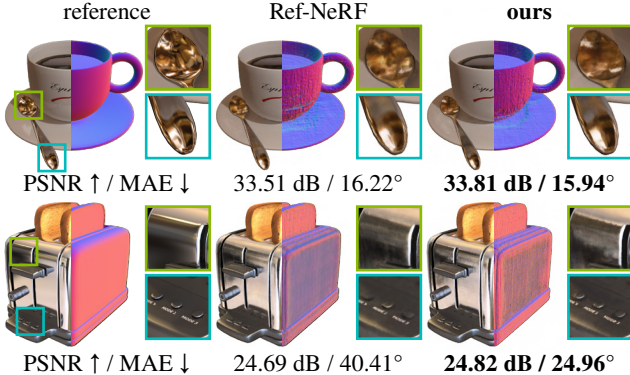Figure 8. Visualization of our modifications to Ref-NeRF [58].

PSNR ↑ / MAE ↓   33.51 dB / 16.22°   **33.81 dB / 15.94°**

PSNR ↑ / MAE ↓   24.69 dB / 40.41°   **24.82 dB / 24.96°**

Figure 9. Qualitative results for the Shiny Blender dataset.

Table 4. PSNR values on the Shiny Blender dataset.

Shiny Blender	Ref-NeRF	ours
TEAPOT	**44.07**	43.35
TOASTER	24.69	**24.82**
CAR	**29.07**	28.84
BALL	32.48	**32.79**
COFFEE	33.51	**33.81**
HELMET	29.31	**30.31**
mean	32.19	**32.32**
median	30.89	**31.55**

Table 5. Mean average error (MAE) values for learned normals on the Shiny Blender dataset.

Shiny Blender	Ref-NeRF	ours
TEAPOT	48.48	**42.86**
TOASTER	40.41	**24.96**
CAR	21.68	**18.06**
BALL	98.94	**38.18**
COFFEE	16.22	**15.94**
HELMET	50.62	**21.46**
mean	46.06	**26.91**
median	44.45	**23.21**

tory of Multi-NeRF to reduce the learning rate when using smaller batch sizes. Instead of scheduling the learning rate to decay from 2×10^{-3} to 2×10^{-5}, we use a schedule from 5×10^{-4} to 5×10^{-6}. These settings lead to worse

performance than what Ref-NeRF reports [58]. Therefore, for a fair comparison, we run Ref-NeRF with our training settings. We use the default $250 \,\text{k}$ iterations, which on an NVIDIA V100 Tensor Core GPU takes 18 hours.

C. Relationship to other prior work

Occupancy networks. Mescheder et al. [35] and Niemeyer et al. [40] propose an alternative to volume rendering for stochastic opaque solid models. They define an occupancy and vacancy model as in Equation (24) similar to NeuS,

$$\text{v}_{\text{ONet}}(\boldsymbol{x}) \equiv \Psi_{\text{logistic}}(s\,\text{f}(\boldsymbol{x})). \tag{39}$$

where the mean implicit function $\text{f}(\boldsymbol{x})$ is a (trainable) neural field. However, instead of converting this occupancy to an attenuation coefficient σ for volume rendering, they define what we term the *maximum a posteriori* (MAP) opaque solid $\mathcal{O}_{\text{MAP}} \equiv \left\{ \boldsymbol{x} \in \mathbb{R}^3 : \text{v}(x) \leq \text{1/2} \right\}$. This approach is equivalent to setting at each point $\boldsymbol{x}$ the random indicator function $\text{I}(\boldsymbol{x})$ equal to its MAP estimate $\hat{\text{I}}_{\text{MAP}}(\boldsymbol{x}) \equiv \mathbf{1}_{\{\text{v}(x) \leq \text{1/2}\}}$—hence the term MAP opaque solid.

Mescheder et al. [35] proceed to render $\mathcal{O}_{\text{MAP}}$ using *deterministic* intersection queries and rendering algorithms. This approach is analogous to setting, during deterministic rendering, the free-flight distance $t^*_{\boldsymbol{x},\boldsymbol{\omega}}$ equal to its MAP estimate $t^*_{\boldsymbol{x},\boldsymbol{\omega},\text{MAP}} \equiv \text{argmax}_{t \in [0,+\infty)}\, \text{p}^{\text{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$. By contrast, volume rendering (Equation (5)) computes the expectation over the free-flight distance $t^*_{\boldsymbol{x},\boldsymbol{\omega}}$. By using point estimation instead of expectation, the MAP approach discards uncertainty information available in $\text{o}(\boldsymbol{x})$ about the scene. Additionally, recent work has demonstrated the benefits of volumetric representations for novel-view synthesis [7, 36, 58] and 3D reconstruction [45, 63, 66], motivating our focus on volumetric representations.

Discretization approaches. Another approach for deriving volumetric representations for opaque solids is to discretize a ray and define the free-flight distribution using discrete occlusion events [8, 45, 55]. We specialize this discussion to exponential transport. Along each ray, these approaches are discrete-time approximations to the continuous-time binary Markov process $\text{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$ underlying our model [42]. Alternatively, but equivalently, they correspond to a piecewise-constant approximation to the attenuation coefficient σ associated with the indicator function, as we explain below.

We consider a segment $[t_0, t_N]$ along a ray $\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$, which we break into N segments $[t_n, t_{n+1}]$, $n = 0, \ldots, N-1$, such that $t_0 < t_1 < \cdots < t_N$. (These segments can have unequal lengths.) The probability that a ray will *not* terminate within segment $[t_n, t_{n+1}]$ equals:

$$\text{V}_n \equiv \text{T}_{\boldsymbol{x},\boldsymbol{\omega}}(t_n, t_{n+1}) \tag{40}$$

$$= \exp\left\{ -\int_{t_n}^{t_{n+1}} \sigma(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(s), \boldsymbol{\omega})\, \text{d}s \right\}, \tag{41}$$

where we used Definition 1 and expanded the notation for transmittance in Equations (1) and (6). The quantity V_n is a discrete analogue of the vacancy v of the continuous-time Markov process $I_{x,\omega}(t)$: it equals the probability that $I_{x,\omega}(t) = 0$ for all $t \in [t_n, t_{n+1}]$, and thus all points along the ray segment are vacant. We also define a discrete occupancy as $O_n \equiv 1 - V_n$. With this notation, assuming Markovianity, the probability that a ray terminates at segment n after traveling through the previous segments—i.e., the discrete analogue of the free-flight distribution—equals:

$$P_n^{\text{ff}} \equiv O_n \prod_{m=0}^{n-1} V_m = (1 - V_n) \prod_{m=0}^{n-1} V_m . \quad (42)$$

Lastly, if we approximate the attenuation coefficient on the segment $[t_n, t_{n+1}]$ as a constant $\bar{\sigma}_n$, then from Equation (40) the discrete vacancy becomes:

$$V_n = \exp\{-\bar{\sigma}_n \cdot |t_{n+1} - t_n|\}. \quad (43)$$

Equations (42)–(43) correspond exactly to the model for ray tracing through discretized opaque geometry in previous literature [8, 45, 55, 56]. As we discuss in Appendix D, they are also equivalent to the common procedure for numerically *approximating* the free-flight distribution when volume rendering continuous opaque geometry [34, 36].

Importantly, most previous discretization-based approaches model not the (piecewise-linear) attenuation coefficient $\bar{\sigma}$ in Equation (43), but directly the discrete vacancy V_n (or occupancy O_n). They do so using either a grid [8, 55, 56], or a neural field [45]—i.e., $V_n := \text{neural\_net}(r_{x,\omega}(t_n))$.

Unfortunately, directly modeling the discrete vacancy in this way does not account for the ray-segment length term $|t_{n+1} - t_n|$ in Equation (43), which is problematic when considering segments of varying length (e.g., in multi-resolution approaches, or when ray discretization is non-uniform). As an example, we consider the discrete free-flight distribution P_n^{ff} in Equation (42) for the same ray $r_{x,\omega}(t)$, first at discretization resolution Δt, then at a finer resolution $\Delta t/2$. In the latter case, P_n^{ff} will equal the product of twice as many V_n terms as in the first case. If these terms are directly provided by, e.g., a neural field that does not account for the change in ray-segment length as resolution changes, then increasing discretization resolution results in a much lower value for the discrete free-flight distribution of the same ray. By contrast, we can elide this problem by using the neural field to output the attenuation coefficient $\bar{\sigma}_n$, then computing the vacancy terms V_n through Equation (40) while accounting for varying ray-segment lengths.

D. Implementation details

We discuss implementation details for our 3D reconstruction experiments. Our volumetric neural rendering pipeline builds on top of the NeuS pipeline [63].

Quadrature algorithm. An important algorithmic compo-

nent of volumetric rendering pipelines is the numerical integration (i.e., quadrature) along a ray of the free-flight distribution and expected radiance terms in the volume rendering equation (5). Quadrature algorithms work in two stages: first, they sample locations along the ray; second, they approximate the free-flight distribution and expected radiance along the ray segments between samples. Several quadrature algorithms have appeared in both graphics [18, 27, 30, 41, 43] and vision [7, 36, 45, 58, 63, 66]. In particular, the volumetric representations in prior work we discussed in Section 4 all use their own quadrature algorithms. Unfortunately, this makes it difficult to experimentally compare different volumetric representations on equal terms: empirically, we have found that performance differences between different representations are often primarily due to the different underlying quadrature algorithms, rather than the different attenuation coefficient expressions. Therefore, to ensure a fair and informative comparison, in our experiments we use for all volumetric representations the same quadrature algorithm, which combines a sampling technique inspired by Oechsle et al. [45] with the numerical approximation by Max [34] (Equations (42)–(43)). We have found that this quadrature algorithm performs well for all representations we tested.

The sampling component of our quadrature algorithm, which we visualize in Figure 10, works as follows: We first intersect a ray with a bounding sphere of the scene. Then, we divide the ray into 1024 segments, evaluate the mean implicit function f at the endpoints of each segment, and check for sign changes between the endpoints. The free-flight distribution p^{ff} is concentrated within the segment containing the first sign change of f, and Oechsle et al. [45] suggest placing most samples in this segment. We follow their suggestion and sample a total of 64 points in three sets—one-third of the sample points within the segment with the first sign change, another third in the part of the ray before this segment, and the remaining third in the part of the ray after. If there is no segment with a sign change, we allocate the entire set of 64 samples along the entire ray. For each of the three sample sets, we select the sample locations within their corresponding ray intervals using the equidistant sampling comb approach of Kettunen et al. [27]: we deterministically place the samples at equidistant intervals, then shift them all by the same uniformly-sampled random offset.

Network architecture. Our pipeline mimics NeuS [63], using two multi-layer perceptrons (MLPs) to encode the mean implicit function and volume emission, with the addition of another small MLP to encode the anisotropy parameter.

The mean implicit function MLP has 8 hidden layers with 256 neurons per layer and SoftPlus activations with $\beta = 100$. A skip connection combines the output of the fourth layer with the original input. It takes as input the spatial position, and outputs the mean implicit function value and a 256-dimensional feature vector.
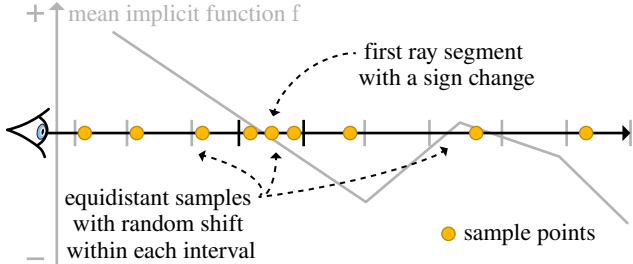
Figure 10. Visualization of our importance sampling algorithm, whoch concentrates points within the first ray segment at whose endpoints the mean implicit function f has opposite signs.

The volume emission MLP has 4 hidden layers with 256 neurons per layer and ReLU activations. It takes as input the spatial position, viewing direction, unit normal of the isosurface of the vacancy function, and the 256-dimensional feature vector from the mean implicit function MLP.

The anisotropy MLP has 1 hidden layer with 256 neurons and sigmoid activation. It takes as input the 256-dimensional feature vector from the mean implicit function MLP.

All positional and directional inputs to the MLPs use frequency encoding [36], with 6 frequencies for positional and 4 frequencies for directional inputs. As in NeuS, all linear layers use weight normalization [51] to stabilize training.

Background radiance field. To enable surface reconstruction on unbounded scenes, such as those in the BlendedMVS dataset, we follow the approach of NeuS [63] and VolSDF [66]: We perform surface reconstruction only inside a bounding sphere, and treat its exterior as a neural radiance field. We use the *inverted sphere parameterization* from NeRF++ [69] for the exterior of the sphere. To learn the background radiance field, we sample along each ray an additional 32 points on the exterior of the bounding sphere, uniformly with respect to inverse distance in the range $[0, 1/R)$, where $R = 3$ is the bounding sphere radius. As in VolSDF, we do not learn a background radiance field for the DTU scenes.

Regularization and initialization. We follow NeuS [63] and use eikonal regularization [20], and geometric initialization [4] for the mean implicit function f.

Training details. Our training protocol is the same as in NeuS [63]. We train using the ADAM optimizer [28] for $300\,\mathrm{k}$ iterations with 512 rays per batch. During the first $5\,\mathrm{k}$ iterations, we control the learning rate with a linear warmup from 0 to 5×10^{-4}; for the remaining iterations, we use a cosine-decay schedule until the learning rate reaches a minimum of 2.5×10^{-5}. Training on an NVIDIA V100 Tensor Core GPU training takes 11 hours without the background radiance field, and 12 hours with it.

E. Additional quantitative results

We provide complete versions of Tables 2–3 in Tables 6–7 (for Table 2) and Tables 8–9 (for Table 3).

In Table 9, we evaluate an additional option for the dis-

Table 6. Chamfer distances on the DTU dataset.

DTU	VolSDF	NeuS	**ours**
24	2.25	3.57	**1.92**
37	3.19	4.02	**2.35**
40	**1.94**	1.99	1.96
55	1.61	1.71	**1.11**
63	1.87	2.04	**1.83**
65	2.12	2.37	**2.01**
69	1.61	1.70	**1.30**
83	1.60	2.33	**1.53**
97	2.18	2.38	**1.62**
105	1.73	3.17	**1.50**
106	0.94	1.07	**0.71**
110	1.91	**1.09**	1.56
114	1.51	1.16	**0.83**
118	**1.36**	1.37	1.52
122	**1.74**	1.83	1.78
mean	1.84	2.17	**1.57**
median	1.74	1.99	**1.56**

tribution of normals: We use the *SGGX distribution* that Heitz et al. propose for stochastic microparticle geometry—specifically its *surface-like* version [23, Equations (13)–(14)]. Using our notation and a reparameterization in terms of a (spatially varying) anisotropy parameter $\alpha(\boldsymbol{x}) \in [0, 1]$, this distribution and corresponding projected area equal:

$$\mathrm{D}_{\boldsymbol{x},\mathrm{SGGX}}(\boldsymbol{m}) \equiv \frac{1}{\mathrm{D}(\alpha(\boldsymbol{x}))\Big(1 - \alpha(\boldsymbol{x})^2 |\boldsymbol{m} \cdot \boldsymbol{n}(\boldsymbol{x})|^2\Big)^2}, \quad (44)$$

$$\sigma^{\perp}_{\mathrm{SGGX}}(\boldsymbol{x}, \boldsymbol{\omega}) \equiv$$

$$\frac{1}{\Sigma(\alpha(\boldsymbol{x}))}\sqrt{\alpha(\boldsymbol{x})^2 |\boldsymbol{\omega} \cdot \boldsymbol{n}(\boldsymbol{x})|^2 + (1 - \alpha(\boldsymbol{x})^2)}, \quad (45)$$

with normalization coefficients

$$\mathrm{D}(\alpha) \equiv \frac{\alpha + \mathrm{arctanh}(\alpha) - \alpha^2 \, \mathrm{arctanh}(\alpha)}{2\alpha\pi}, \quad (46)$$

$$\Sigma(\alpha) \equiv 1 + \left(\frac{1}{\alpha} - \alpha\right) \mathrm{arcsinh}\left(\frac{\alpha}{\sqrt{1 - \alpha^2}}\right). \quad (47)$$

Comparing the SGGX projected area of Equation (45) to the mixture one in Equation (17), we see that—the normalization coefficients aside—they behave similarly: $\sigma^{\perp}_{\mathrm{SGGX}}$ is a sum of squared terms, whereas $\sigma^{\perp}_{\mathrm{mix}}$ is a sum of linear terms. At the limits $\alpha = 1$ and $\alpha = 0$, they both reduce to fully anisotropic and isotropic, respectively. Heitz et al. [23] provide detailed justification for the use of the SGGX distribution in the context of stochastic microparticle geometry. In our context of stochastic solid geometry, the linear mixture performed better in our ablation study (Table 9), so we used it for the rest of our experiments.

Table 7. Chamfer distances on the NeRF Realistic Synthetic dataset.

NeRF RS	VolSDF	NeuS	**ours**
CHAIR	**0.0105**	0.0182	0.0196
LEGO	0.0354	0.0555	**0.0156**
DRUMS	0.6081	0.9897	**0.1769**
FICUS	**0.0279**	0.0796	0.0436
HOTDOG	0.1654	0.2104	**0.1295**
MATERIALS	0.0231	0.0444	**0.015**
MIC	0.9141	**0.1214**	0.4344
SHIP	0.2351	0.0901	**0.0711**
mean	0.252	0.201	**0.113**
median	0.100	0.085	**0.057**

Table 8. Chamfer distances on the DTU dataset when using different implicit function distributions Ψ for the density $\sigma^{\parallel}$.

Ψ model	logistic	Laplace	**Gaussian**
24	2.73	**1.92**	1.99
37	3.56	3.65	**3.08**
40	**1.94**	2.32	2.28
55	1.77	1.65	**1.64**
63	1.86	1.76	1.76
65	2.67	2.60	**2.45**
69	1.73	1.58	**1.31**
83	1.85	1.94	**1.69**
97	**1.82**	2.13	1.83
105	1.90	2.04	**1.74**
106	1.09	0.98	0.98
110	1.98	1.92	**1.76**
114	1.29	1.43	**0.96**
118	**1.39**	1.54	1.67
122	2.11	1.89	**1.59**
mean	1.98	1.96	**1.78**
median	1.86	1.92	**1.74**

Table 9. Chamfer distances on the DTU dataset when using different distributions of normals D for the projected area $\sigma^{\perp}$.

D model	delta (ReLU)	delta	mixture (const.)	**mixture (var.)**	SGGX (var.)
24	3.57	2.73	2.43	2.16	**2.10**
37	4.02	3.56	4.16	3.40	**3.32**
40	1.99	1.94	1.94	**1.76**	1.83
55	1.71	1.77	1.85	**1.43**	1.64
63	2.04	1.86	1.85	**1.60**	1.80
65	2.37	2.67	2.19	**1.97**	2.34
69	1.70	1.73	1.57	**1.54**	1.43
83	2.33	1.85	1.79	1.55	**1.49**
97	2.38	**1.82**	2.25	1.91	2.20
105	3.17	1.90	1.85	**1.53**	1.82
106	1.07	1.09	0.99	1.32	**0.89**
110	1.90	1.98	1.89	**1.59**	1.79
114	1.16	1.29	1.37	1.26	**1.15**
118	1.37	1.39	1.75	**1.31**	1.35
122	1.83	2.11	**1.73**	1.85	1.95
mean	2.17	1.98	1.97	**1.75**	1.81
median	1.99	1.86	1.85	**1.59**	1.80

F. Proofs

We provide proofs for our theory in Sections 3 and 4.

F.1. Exponential transport in opaque solids

We begin with the proof of Theorem 4. Our proof requires background on continuous-time discrete-space Markov processes, for which we refer to Norris [42] as one of many excellent textbooks on this subject matter. We provide pointers to specific parts throughout the proof.

To sketch the proof: The free-flight distribution $\mathrm{p}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$ is the probability density that, starting from $\mathrm{I}(\boldsymbol{x}) = 0$, the first $0 \to 1$ transition of the indicator function along the ray (i.e., first intersection) happens at distance t. For this distance to be an exponential random variable, the counting process of $0 \to 1$ transitions must be a *Poisson process*. Then, the continuous-time discrete-space random process

$\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$ must be Markovian,[7] and Equation (12) follows from the Kolmogorov equations [42] after enforcing reciprocity and reversibility.

Proof. We extend Definition 1 to define the *conditional transmittance*:

$$\overline{\mathrm{T}}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv \mathrm{Pr}_{\mathcal{O}}\{t^{*}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \geq t \mid \mathrm{I}(\boldsymbol{x}) = 0\}, \quad (48)$$

the *conditional free-flight distribution*:

$$\overline{\mathrm{p}}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv -\frac{\mathrm{d}\,\overline{\mathrm{T}}_{\boldsymbol{x},\boldsymbol{\omega}}}{\mathrm{d}t}(t), \quad (49)$$

and the *conditional attenuation coefficient*:

$$\overline{\sigma}(\boldsymbol{x}, \boldsymbol{\omega}) \equiv \overline{\mathrm{p}}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}(0). \quad (50)$$

Under the assumption of exponential transport, the transmittance equals the conditional transmittance:

$$\mathrm{T}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \stackrel{\mathrm{exponential}}{=} \overline{\mathrm{T}}_{\boldsymbol{x},\boldsymbol{\omega}}(t). \quad (51)$$

and likewise for the free-flight distribution and attenuation coefficient [9, Supplementary Section 1, Equation (68)]. Thus, we work with the conditional quantities for this proof.

First jump time. The restriction $\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}$ of the indicator function I along the ray $\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}$ is a

---

[7]Readers familiar with the derivation of exponential transport for stochastic microparticle geometry will recognize this argument: In that setting, the free-flight distribution describes the distance to the first collision with a microparticle. If microparticle locations are independent, then counting collisions is a Poisson process, and the distance to the first collision is an exponential random variable. In the stochastic solid geometry setting, we replace collisions with $0 \to 1$ transitions, and the assumption of independent particle locations with the Markovianity assumption of Equation (11).

continuous-time discrete-space (binary) stochastic process. Then, the conditional free-flight distance $t^*_{\boldsymbol{x},\boldsymbol{\omega}} \equiv \max\{t \in [0,\infty) : \mathrm{V}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = 1 \,|\, \mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(0) = 0\}$ equals the *first jump time* $\mathrm{J}_{\boldsymbol{x},\boldsymbol{\omega}} \equiv \min\{t \in [0,\infty) : \mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = 1 \,|\, \mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(0) = 0\}$ of this process; that is, the distance travelled along the ray until the *first* $0 \to 1$ transition of $\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}$, or equivalently until we first move from outside to inside the opaque solid [42, Section 2.2]. Consequently, the conditional free-flight distribution $\overline{\mathrm{p}}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}$ equals the probability density function of $\mathrm{J}_{\boldsymbol{x},\boldsymbol{\omega}}$.

Markov property. From Equation (7), exponential transport requires that the conditional free-flight distance $t^*_{\boldsymbol{x},\boldsymbol{\omega}}$, and thus the first jump time $\mathrm{J}_{\boldsymbol{x},\boldsymbol{\omega}}$, be exponential random variables. This is equivalent to the process $\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}$ being *Markovian*, that is, satisfying Equation (11) [42, Section 2.6].

Transition coefficients and generator matrix. The Markov property implies that we can associate with the process $\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}$ a *generator matrix* [42, Section 2.8]:

$$\mathbf{Q}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv \begin{bmatrix} -\sigma^{01}_{\boldsymbol{x},\boldsymbol{\omega}}(t) & \sigma^{01}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \\ \sigma^{10}_{\boldsymbol{x},\boldsymbol{\omega}}(t) & -\sigma^{10}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \end{bmatrix}, \qquad (52)$$

where $\sigma^{01}_{\boldsymbol{x},\boldsymbol{\omega}}$ and $\sigma^{10}_{\boldsymbol{x},\boldsymbol{\omega}}$ are nonnegative *transition rates*. The transition rate $\sigma^{01}_{\boldsymbol{x},\boldsymbol{\omega}}$ is the rate of the exponential distribution of the first jump time $\mathrm{J}_{\boldsymbol{x},\boldsymbol{\omega}}$, and thus we can relate it to the conditional free-flight distribution as:

$$\overline{\mathrm{p}}^{\mathrm{ff}}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \sigma^{01}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \exp\left(-\int_0^t \sigma^{01}_{\boldsymbol{x},\boldsymbol{\omega}}(s)\,\mathrm{d}s\right). \qquad (53)$$

Analogously, the transition rate $\sigma^{10}_{\boldsymbol{x},\boldsymbol{\omega}}$ is the rate of the exponential distribution of the first jump time for $1 \to 0$ transitions (moving from inside to outside the opaque solid). Comparing Equations (7) and (53), we see that the transition rate $\sigma^{01}_{\boldsymbol{x},\boldsymbol{\omega}}$ is the restriction of the attenuation coefficient σ along the ray $\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}$. Even though we need only $\sigma^{01}_{\boldsymbol{x},\boldsymbol{\omega}}$ and not $\sigma^{10}_{\boldsymbol{x},\boldsymbol{\omega}}$ for exponential transport, our proof characterizes both.

Kolmogorov equations. To proceed, we will use the *transition probabilities* of the process $\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}$ for all $i,j \in \{0,1\}$:

$$\overline{\mathrm{P}}^{ij}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv \Pr(\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = j \,|\, \mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = i \,|\,), \qquad (54)$$

and the corresponding *transition probability matrix*:

$$\overline{\mathbf{P}}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv \begin{bmatrix} \overline{\mathrm{P}}^{00}_{\boldsymbol{x},\boldsymbol{\omega}}(t) & \overline{\mathrm{P}}^{01}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \\ \overline{\mathrm{P}}^{10}_{\boldsymbol{x},\boldsymbol{\omega}}(t) & \overline{\mathrm{P}}^{11}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \end{bmatrix}. \qquad (55)$$

Because $\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}$ is Markovian, this matrix satisfies the *Kolmogorov forward equation* [42, Theorem 2.8.2]:

$$\frac{\mathrm{d}\overline{\mathbf{P}}_{\boldsymbol{x},\boldsymbol{\omega}}}{\mathrm{d}t}(t) = \overline{\mathbf{P}}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \cdot \mathbf{Q}_{\boldsymbol{x},\boldsymbol{\omega}}(t). \qquad (56)$$

To proceed, we will derive a version of Equation (56) that uses the unconditional *probability matrix*:

$$\mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \equiv \begin{bmatrix} \Pr(\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = 0) \\ \Pr(\mathrm{I}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = 1) \end{bmatrix} = \begin{bmatrix} \mathrm{v}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \\ \mathrm{o}_{\boldsymbol{x},\boldsymbol{\omega}}(t) \end{bmatrix}. \qquad (57)$$

Using marginalization, we can relate $\mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}$ and $\overline{\mathbf{P}}_{\boldsymbol{x},\boldsymbol{\omega}}$ as:

$$\mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \overline{\mathbf{P}}_{\boldsymbol{x},\boldsymbol{\omega}}(t)^\top \cdot \mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}(0). \qquad (58)$$

Differentiating with respect to distance, we have:

$$\frac{\mathrm{d}\mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}}{\mathrm{d}t}(t) = \frac{\mathrm{d}\overline{\mathbf{P}}_{\boldsymbol{x},\boldsymbol{\omega}}}{\mathrm{d}t}(t)^\top \cdot \mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}(0). \qquad (59)$$

Using the Kolmogorov forward equation (56):

$$\frac{\mathrm{d}\mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}}{\mathrm{d}t}(t) = \mathbf{Q}_{\boldsymbol{x},\boldsymbol{\omega}}(t)^\top \cdot \overline{\mathbf{P}}_{\boldsymbol{x},\boldsymbol{\omega}}(t)^\top \cdot \mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}(0). \qquad (60)$$

Lastly, using Equation (58), we arrive at what we term the *unconditional Kolmogorov equation*:

$$\frac{\mathrm{d}\mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}}{\mathrm{d}t}(t) = \mathbf{Q}_{\boldsymbol{x},\boldsymbol{\omega}}(t)^\top \cdot \mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}(t). \qquad (61)$$

Relating ray to local quantities. An important property of the unconditional Kolmogorov equation (61) is that it is *local*, that is, it uses quantities defined at only one point, $\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t)$. In particular, even though our derivation so far has considered a ray starting at point $\boldsymbol{x}$ and traveling at direction $\boldsymbol{\omega}$, Equation (61) requires knowledge of only the direction of the ray not its history (origin $\boldsymbol{x}$ or distance t).

We will use this property to rewrite Equation (61) in terms of *local* quantities. We associate with every point $\boldsymbol{x} \in \mathbb{R}^3$ and direction $\boldsymbol{\omega} \in \mathcal{S}^2$ the *local* transition rates $\sigma^{01}(\boldsymbol{x},\boldsymbol{\omega})$ and $\sigma^{10}(\boldsymbol{x},\boldsymbol{\omega})$, generator matrix

$$\mathbf{Q}(\boldsymbol{x},\boldsymbol{\omega}) \equiv \begin{bmatrix} -\sigma^{01}(\boldsymbol{x},\boldsymbol{\omega}) & \sigma^{01}(\boldsymbol{x},\boldsymbol{\omega}) \\ \sigma^{10}(\boldsymbol{x},\boldsymbol{\omega}) & -\sigma^{10}(\boldsymbol{x},\boldsymbol{\omega}) \end{bmatrix}, \qquad (62)$$

and unconditional probability matrix

$$\mathbf{P}(\boldsymbol{x}) \equiv \begin{bmatrix} \mathrm{v}(\boldsymbol{x}) \\ \mathrm{o}(\boldsymbol{x}) \end{bmatrix}. \qquad (63)$$

It follows that we can relate ray to local quantities as:

$$\sigma^{01}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \sigma^{01}(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t),\boldsymbol{\omega}), \qquad (64)$$

$$\sigma^{10}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \sigma^{10}(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t),\boldsymbol{\omega}), \qquad (65)$$

$$\mathbf{Q}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \mathbf{Q}(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t),\boldsymbol{\omega}), \qquad (66)$$

$$\mathbf{P}_{\boldsymbol{x},\boldsymbol{\omega}}(t) = \mathbf{P}(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}(t)). \qquad (67)$$

From Equations (7), (53) and (64), we recognize the local transition rate $\sigma^{01}(\boldsymbol{x},\boldsymbol{\omega})$ as the attenuation coefficient $\sigma(\boldsymbol{x},\boldsymbol{\omega})$. Additionally, for any quantity $(*)$:

$$\frac{\mathrm{d}(*)_{\boldsymbol{x},\boldsymbol{\omega}}}{\mathrm{d}t}(t) = \boldsymbol{\omega} \cdot \nabla_{\boldsymbol{x}}(*)(\boldsymbol{r}_{\boldsymbol{x},\boldsymbol{\omega}}). \qquad (68)$$

Local form of the Kolmogorov equation. Using the correspondences between ray and local quantities in Equations (64)–(68), we can rewrite the unconditional Kolmogorov equation Equation (61) in *local form*:

$$\boldsymbol{\omega} \cdot \nabla_{\boldsymbol{x}}\mathbf{P}(\boldsymbol{x}) = \mathbf{Q}(\boldsymbol{x},\boldsymbol{\omega})^\top \cdot \mathbf{P}(\boldsymbol{x}). \qquad (69)$$

Expanding $\mathbf{Q}$ and $\mathbf{P}$ from Equations (62) and (63), and using $\mathrm{o}(\boldsymbol{x}) = 1 - \mathrm{v}(\boldsymbol{x})$, we arrive at a scalar equation that we term the *local form of the unconditional Kolmogorov equation*:

$$\boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) =$$
$$- \operatorname{v}(\boldsymbol{x}) \, \sigma^{01}(\boldsymbol{x}, \boldsymbol{\omega}) + (1 - \operatorname{v}(\boldsymbol{x})) \, \sigma^{10}(\boldsymbol{x}, \boldsymbol{\omega}). \quad (70)$$

Exponential transport requires that Equation (70) hold for rays passing through point $\boldsymbol{x}$ along any direction $\boldsymbol{\omega}$. In particular, if we consider a ray in direction $-\boldsymbol{\omega}$, we arrive at:

$$- \boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) =$$
$$- \operatorname{v}(\boldsymbol{x}) \, \sigma^{01}(\boldsymbol{x}, -\boldsymbol{\omega}) + (1 - \operatorname{v}(\boldsymbol{x})) \, \sigma^{10}(\boldsymbol{x}, -\boldsymbol{\omega}). \quad (71)$$

Enforcing reciprocity. At any point $\boldsymbol{x}$ and direction $\boldsymbol{\omega}$, Equations (70) and (71) provide two linear constraints on the four transition rates $\sigma^{01}(\boldsymbol{x}, \boldsymbol{\omega}), \sigma^{01}(\boldsymbol{x}, -\boldsymbol{\omega}), \sigma^{10}(\boldsymbol{x}, \boldsymbol{\omega}), \sigma^{10}(\boldsymbol{x}, -\boldsymbol{\omega})$ as functions of the vacancy $\operatorname{v}(\boldsymbol{\omega})$ and its directional derivative. Enforcing reciprocity provides an additional linear constraint through the requirement of Equation (8):

$$\sigma^{01}(\boldsymbol{x}, \boldsymbol{\omega}) = \sigma^{01}(\boldsymbol{x}, -\boldsymbol{\omega}). \quad (72)$$

Equations (70)–(72) form an *underdetermined* linear system on the four transition rates that, after requiring nonnegativity for the transition rates, admits the family of solutions:

$$\begin{bmatrix} \sigma^{01}(\boldsymbol{x}, \boldsymbol{\omega}) \\ \sigma^{01}(\boldsymbol{x}, -\boldsymbol{\omega}) \\ \sigma^{10}(\boldsymbol{x}, \boldsymbol{\omega}) \\ \sigma^{10}(\boldsymbol{x}, -\boldsymbol{\omega}) \end{bmatrix} =$$

$$\begin{cases} \begin{bmatrix} \frac{\boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) + \tau(\boldsymbol{x}, \boldsymbol{\omega})(1 - \operatorname{v}(\boldsymbol{x}))}{\operatorname{v}(\boldsymbol{x})} \\ \frac{\boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) + \tau(\boldsymbol{x}, \boldsymbol{\omega})(1 - \operatorname{v}(\boldsymbol{x}))}{\operatorname{v}(\boldsymbol{x})} \\ \frac{2\boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) + \tau(\boldsymbol{x}, \boldsymbol{\omega})(1 - \operatorname{v}(\boldsymbol{x}))}{1 - \operatorname{v}(\boldsymbol{x})} \\ \tau(\boldsymbol{x}, \boldsymbol{\omega}) \end{bmatrix}, & \text{if } \boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) \geq 0, \\[2em] \begin{bmatrix} \frac{-\boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) + \tau(\boldsymbol{x}, \boldsymbol{\omega})(1 - \operatorname{v}(\boldsymbol{x}))}{\operatorname{v}(\boldsymbol{x})} \\ \frac{-\boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) + \tau(\boldsymbol{x}, \boldsymbol{\omega})(1 - \operatorname{v}(\boldsymbol{x}))}{\operatorname{v}(\boldsymbol{x})} \\ \tau(\boldsymbol{x}, \boldsymbol{\omega}) \\ \frac{-2\boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) + \tau(\boldsymbol{x}, \boldsymbol{\omega})(1 - \operatorname{v}(\boldsymbol{x}))}{1 - \operatorname{v}(\boldsymbol{x})} \end{bmatrix}, & \text{if } \boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x}) < 0, \end{cases} \quad (73)$$

where $\tau(\boldsymbol{x}, \boldsymbol{\omega}) = \tau(\boldsymbol{x}, -\boldsymbol{\omega}) \geq 0$ is a free variable. We can verify this solution by computing the range and null space of the linear system of Equations (70)–(72). Focusing on the attenuation coefficient $\sigma \equiv \sigma^{01}$, we can write succinctly:

$$\sigma(\boldsymbol{x}, \boldsymbol{\omega}) = \sigma(\boldsymbol{x}, -\boldsymbol{\omega}) =$$
$$\frac{|\boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x})| + \tau(\boldsymbol{x}, \boldsymbol{\omega})(1 - \operatorname{v}(\boldsymbol{x}))}{\operatorname{v}(\boldsymbol{x})}. \quad (74)$$

Enforcing reversibility. Equation (74) includes the free variable τ whose value we cannot resolve by enforcing only exponential transport (i.e., exponential free-flight distributions) and reciprocity. We can determine the value of τ by considering the following: For any two points $\boldsymbol{x}$ and $\boldsymbol{y}$, Equation (70) allows computing the vacancy v of one point from

that of the other, by integrating along the linear segment connecting the two points. As the two vacancies correspond to probabilities of the same underlying random field I, we should arrive at consistent results whether we integrate in the direction from $\boldsymbol{x}$ towards $\boldsymbol{y}$, or in the reverse direction from $\boldsymbol{y}$ towards $\boldsymbol{x}$. Concretely, using the ray notation:

1. In one direction, we start at $\boldsymbol{r}_{\boldsymbol{x}, \boldsymbol{x} \to \boldsymbol{y}}(0) = \boldsymbol{x}$ with initial condition $\operatorname{v}_{\boldsymbol{x}, \boldsymbol{x} \to \boldsymbol{y}}(0) = \operatorname{v}(\boldsymbol{x})$, and use Equation (70) to integrate $\operatorname{v}_{\boldsymbol{x}, \boldsymbol{x} \to \boldsymbol{y}}(t)$ with respect to t along the ray until we reach $\boldsymbol{r}_{\boldsymbol{x}, \boldsymbol{x} \to \boldsymbol{y}}(\|\boldsymbol{y} - \boldsymbol{x}\|) = \boldsymbol{y}$. The integration result should be $\operatorname{v}_{\boldsymbol{x}, \boldsymbol{x} \to \boldsymbol{y}}(\|\boldsymbol{y} - \boldsymbol{x}\|) = \operatorname{v}(\boldsymbol{y})$.

2. In the reverse direction, we start at $\boldsymbol{r}_{\boldsymbol{y}, \boldsymbol{y} \to \boldsymbol{x}}(0) = \boldsymbol{y}$ with initial condition $\operatorname{v}_{\boldsymbol{y}, \boldsymbol{y} \to \boldsymbol{x}}(0) = \operatorname{v}(\boldsymbol{y})$, and use Equation (70) to integrate $\operatorname{v}_{\boldsymbol{y}, \boldsymbol{y} \to \boldsymbol{x}}(t)$ with respect to t along the ray until we reach $\boldsymbol{r}_{\boldsymbol{y}, \boldsymbol{y} \to \boldsymbol{x}}(\|\boldsymbol{y} - \boldsymbol{x}\|) = \boldsymbol{x}$. The integration result should be $\operatorname{v}_{\boldsymbol{y}, \boldsymbol{y} \to \boldsymbol{x}}(\|\boldsymbol{y} - \boldsymbol{x}\|) = \operatorname{v}(\boldsymbol{x})$.

We term this property *reversibility*,[8] as it corresponds to enforcing consistency between the process $\operatorname{I}_{\boldsymbol{x}, \boldsymbol{\omega}}(t)$ and its directional reverse $\operatorname{I}_{\boldsymbol{x}, -\boldsymbol{\omega}}(t)$. Reversibility along all rays is necessary for the underlying indicator function I to represent a solid, as otherwise views of this solid from different rays would have inconsistent geometry. From Equation (70), reversibility holds only if the right-hand side of this equation is proportional to the vacancy $\operatorname{v}(\boldsymbol{x})$. In turn, from Equation (73), this is only true if $\tau(\boldsymbol{x}, \boldsymbol{\omega}) \propto \operatorname{v}(\boldsymbol{x})/1 - \operatorname{v}(\boldsymbol{x})$. Setting $\tau(\boldsymbol{x}, \boldsymbol{\omega}) := 0$ produces the *minimal* transition rates that guarantee reversibility. Then, Equation (74) simplifies to:

$$\sigma(\boldsymbol{x}, \boldsymbol{\omega}) = \sigma(\boldsymbol{x}, -\boldsymbol{\omega}) = \frac{|\boldsymbol{\omega} \cdot \nabla \operatorname{v}(\boldsymbol{x})|}{\operatorname{v}(\boldsymbol{x})}. \quad (75)$$

This concludes the proof. $\square$

F.2. Stochastic implicit geometry

We now prove Proposition 7.

Proof. Equation (25) follows from the definitions of the vacancy function and the indicator function relative to the implicit function in Definition 6:

$$\operatorname{v}(\boldsymbol{x}) = \Pr(\operatorname{I}(\boldsymbol{x}) = 0) \quad (76)$$
$$= \Pr(\operatorname{G}(\boldsymbol{x}) > 0) \quad (77)$$
$$= 1 - \operatorname{cdf}_{\operatorname{G}(\boldsymbol{x})}(0) \quad (78)$$
$$= 1 - \Psi(-s \operatorname{f}(\boldsymbol{x})) \quad (79)$$
$$= \Psi(s \operatorname{f}(\boldsymbol{x})), \quad (80)$$

where in the last step, we used the assumption that Ψ is symmetric. Equation (24) is a consequence of the property $\operatorname{o}(\boldsymbol{x}) = 1 - \operatorname{v}(\boldsymbol{x})$. Lastly, Equation (26) follows by combining Equations (15) and (25), and the property that the PDF ψ is the derivative of the CDF Ψ. This concludes the proof. $\square$

---

[8]This property is different from the usual notion of time reversibility in continuous-time Markov processes [42, Section 3.7].

We close this subsection by adapting Proposition 7 to the case of a spatially varying scale $s(\boldsymbol{x})$. This case is required for adaptive shell representations [64] and stochastic Poisson surface reconstruction [52] (Appendix A). Equations (24) and (25) remain unchanged:

$$o(\boldsymbol{x}) = \Psi(-s(\boldsymbol{x})\,f(\boldsymbol{x})), \tag{81}$$

$$v(\boldsymbol{x}) = \Psi(s(\boldsymbol{x})\,f(\boldsymbol{x})). \tag{82}$$

Combining Equations (15) and (82) results in:

$$\sigma^{\parallel}(\boldsymbol{x}) = \frac{\psi(s(\boldsymbol{x})\,f(\boldsymbol{x}))\,\|s(\boldsymbol{x})\,\nabla\,f(\boldsymbol{x}) + \nabla s(\boldsymbol{x})\,f(\boldsymbol{x})\|}{\Psi(s(\boldsymbol{x})\,f(\boldsymbol{x}))}. \tag{83}$$

F.3. Density for logistic distribution

Lastly, we prove Equation (28). This equation follows easily from Equation (26) using the properties of the logistic CDF:

$$\psi_{\text{logistic}}(s) \equiv \frac{d\,\Psi_{\text{logistic}}}{ds}(s) \tag{84}$$

$$= \Psi_{\text{logistic}}(s) \cdot (1 - \Psi_{\text{logistic}}(s)) \tag{85}$$

$$= \Psi_{\text{logistic}}(s) \cdot \Psi_{\text{logistic}}(-s). \tag{86}$$