Computational Visual Media
https://doi.org/10.26599/CVM.XXXX

Research Article

Efficient Monte Carlo Rendering of Implicit-Shaped Volumetric Emitters

Jiawei Huang[1], Shaokun Zheng[2], Kun Xu[2], Yoshifumi Kitamura[3], and Jiaping Wang[1](✉)

Abstract We present a practical and unbiased sampling technique for volumetric light sources defined by signed distance functions (SDFs). SDFs can compactly represent complex procedural shapes and support Boolean operations for flexible composition. However, their use as emitters remains underexplored in photorealistic rendering due to the absence of efficient sampling strategies. Our key insight is to model the interior of an SDF as a uniform volume and to project volumetric samples onto the unit sphere centered at the shading point, yielding a solid-angle distribution that captures the emitter's relative spatial layout. We derive the directional probability densities for analytic primitives and generalize the formulation to arbitrary SDFs via Monte Carlo volume estimation. Leveraging robust sphere tracing, our method enables accurate and efficient sampling of SDF emitters without requiring explicit surface parameterization, voxelization, or precomputed tables. Compared to baseline approaches such as uniform directional sampling or surface approximations, our technique achieves significantly lower variance and runtime in next-event estimation, while broadening the expressive power of light sources in rendering.

Keywords Monte Carlo, rendering, signed distance function, next-event estimation, importance sampling

1 Introduction

In artistic and production rendering, light sources play a crucial role not only in illuminating scenes but also in conveying mood, form, and story. In physically-based rendering systems, light sources are typically modeled as emitters: objects that define how energy is radiated into the scene. While most systems support a variety of surface-based emitters such as points, polygons, or area meshes, systems offer far fewer options for volumetric emission. This limitation becomes especially apparent when artists aim to represent emitters with rich internal structure or nontrivial spatial form. In such cases, mesh emitters are often used as a workaround, approximating complex volumetric behavior through surfaces. However, this approach introduces significant overhead in sampling, storage, and shading complexity, limiting both performance and creative flexibility.

This disparity is not due to any fundamental weakness of volumetric emitters, but rather to the challenges they pose in sampling and integration. Existing volumetric models tend to rely on discretized voxel grids, which trade fidelity for tractability. By contrast, signed distance functions (SDFs) offer a compact, expressive alternative for procedural modeling—supporting smooth blends, Boolean operations, and structural edits within a unified representation. Despite their widespread use for geometry definition and scene evaluation, SDFs remain almost entirely unexplored as emitters in Monte Carlo rendering frameworks, owing to the lack of efficient and unbiased sampling strategies over their implicit interiors.

We introduce SDF emitters: a new class of volumetric emitters defined directly by signed distance functions. Our core idea is to treat the interior of an SDF-defined region as a uniform emissive volume and to derive a solid-angle distribution at the shading point through spatial-to-directional projection. This yields accurate and efficient importance sampling without requiring meshing, voxelization, or precomputed tables. Seamlessly integrating into existing Monte Carlo pipelines, SDF emitters expand the expressive range of light modeling—enabling high-fidelity, artist-controllable illumination with minimal overhead and substantial variance reduction.

By integrating SDF emitters seamlessly into existing Monte Carlo rendering pipelines, our method enhances the expressive

1 IDEA (International Digital Economy Academy), Shenzhen 518048, China. E-mail: huangjiawei@idea.edu.cn, jiapw.cg@gmail.com.

2 BNRist, Dept. of CS&T, Tsinghua University, Beijing 100084, China. E-mail: zsk20@mails.tsinghua.edu.cn, xukun@tsinghua.edu.cn.

3 RIEC, Tohoku University, Sendai 980-8577, Japan. E-mail: kitamura@riec.tohoku.ac.jp.

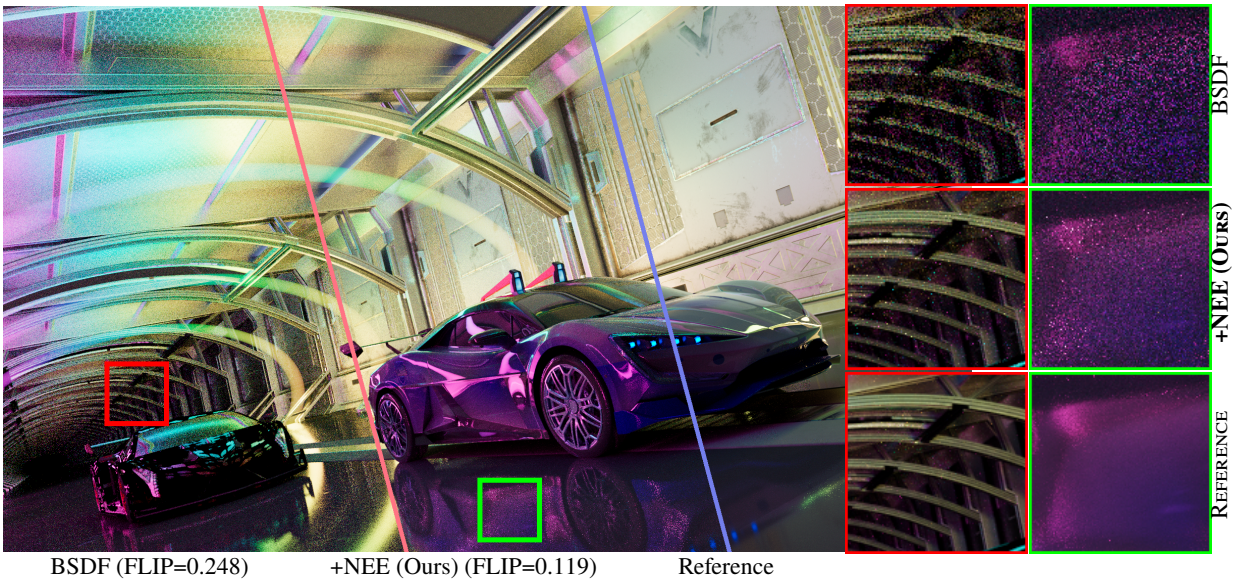


Fig. 1 A tunnel scene with SDF-defined capped-torus volumetric emitters. Left: BSDF sampling only. Center: Our solid-angle NEE with MIS. Right: Reference. All rendered at 256 spp.

power and flexibility of light source modeling, with only small variance through effective importance sampling in next-event estimation.

In summary, the main contributions of this paper are

- the introduction of SDF emitters as a new class of volumetric light sources, enabling implicit geometries to participate in lighting without meshing or discretization,
- the formulation of an unbiased importance sampling technique based on solid-angle distributions projected from volumetric samples, and
- a practical strategy for integrating SDF emitters into Monte Carlo rendering, leveraging robust sphere tracing for intersection and evaluation.

2 Related Work

2.1 Procedural and Implicit Geometry in Rendering

Procedural and implicit representations offer a compact and flexible means of describing complex geometry. Unlike traditional mesh-based models, these methods define shapes through functions or composition rules, allowing efficient storage, evaluation, and manipulation. Such representations are widely used for geometric modeling in rendering, particularly for structured or fractal-like objects.

Signed distance functions (SDFs) are a common choice in this domain. They describe geometry implicitly through a signed scalar field and support constructive operations like unions and intersections. SDFs are broadly used in procedural modeling, differentiable rendering [1], and neural geometry representations [2, 3], and can be parameterized through networks to yield continuous, high-fidelity fields [4, 5]. Recent work has also explored neural implicit surfaces for relightable appearance recovery [6], further demonstrating the versatility of implicit representations in rendering pipelines.

In most of these applications, SDFs serve as surface proxies for visibility testing, shading, or optimization; they are valued for their compactness and efficient intersection routines. In contrast, we explore their use as volumetric emitters defined over interior regions, and develop a solid-angle–based directional sampling technique locally adapted to each shading point, without requiring meshing or voxelization.

2.2 Light Source Modeling and Volumetric Emitters

Various strategies exist for modeling light sources in physically based rendering. Analytic emitters like points, spheres, or rectangles benefit from closed-form intersection and sampling [7–9], but their simplicity limits the range of visual effects. Mesh lights, composed of tessellated surfaces, offer greater flexibility but increase memory and sampling complexity, as they effectively act as large aggregates of triangle emitters.

Volumetric emitters, such as those found in fire or fog, are typically modeled using density grids or proxy geometry [10]. Sampling such media often involves integrating emission along rays, as in the works of Villemin *et al.* [11] and Simon *et al.* [12], which interpret emission in terms of directional contributions from volumetric regions. Heterogeneous volume rendering has also been advanced through biophysical models for subsurface scattering [13] and differentiable

regularization techniques for participating media [14].

Our approach has a similar conceptual basis, constructing directional sampling distributions from volume integrals. Unlike grid-based formulations, however, we operate directly on implicit volumes defined by SDFs. This allows us to retain geometric continuity while avoiding the discretization and memory costs typically associated with volumetric grids.

2.3 Importance Sampling for Direct Lighting

Accurate sampling of light sources is critical for reducing variance in Monte Carlo rendering. Standard techniques like next-event estimation (NEE) and multiple importance sampling (MIS) [15] combine samples from light and BSDF distributions, but they can still struggle in the presence of occlusion, complex emission profiles, or large sets of lights. To enable NEE for surfaces behind refracting surfaces, manifold exploration and polynomial systems are leveraged [16–18].

To better align sampling with the integrand, many methods operate in projected solid angle space [7], particularly for structured emitters like disks and rectangles [8, 9]. Complementary advances in adaptive sampling and reconstruction further reduce variance in gradient-domain rendering [19]. Beyond geometric terms, product importance sampling strategies aim to sample the full product of illumination and material scattering. For instance, Diolatzis *et al.* [20] extended Linearly Transformed Cosines (LTC) [21] to guide paths by the product of light and BSDF. Our method complements these strategies by defining the normalized solid-angle distribution (p_L) for implicit volumetric emitters required by such frameworks, without requiring surface parameterization. This enables spatially adapted sampling for a broader class of emitter shapes and representations. In contrast to data-driven approaches [22] that often require pre-training, our technique remains purely algorithmic and lightweight, making it uniquely suitable for dynamic procedural content.

For scenes with many lights, hierarchical techniques like Lightcuts [23], cone trees [24], and light BVHs [25] improve scalability by clustering and pruning candidate emitters. More recently, sample reuse frameworks like ReSTIR [26] have introduced spatiotemporal resampling, amortizing sampling costs across space and time to enable real-time rendering of direct lighting in complex scenes. Extensions like ReSTIR GI [27] and ReSTIR PT [28] further generalize this idea to indirect illumination.

Our method immediately fits into existing direct lighting pipelines, and can work seamlessly with these many-light sampling frameworks.

3 Formulation of SDF Emitter Sampling

Let $\phi\colon \mathbb{R}^3 \to \mathbb{R}$ be a signed distance function (SDF) defining an implicit surface:

$$\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^3 \mid \phi(\mathbf{x}) = 0\}.$$

We define the corresponding solid region (i.e., the interior of the shape) as

$$\Omega = \{\mathbf{x} \in \mathbb{R}^3 \mid \phi(\mathbf{x}) \leqslant 0\}.$$

To model volumetric emission, we introduce the indicator function:

$$v(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in \Omega, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

This describes a volume with uniform density contained within the implicit boundary defined by the SDF.

We denote the total volume of the region by

$$V := \int_{\mathbb{R}^3} v(\mathbf{x})\, d\mathbf{x}, \tag{2}$$

which serves as a normalization factor throughout.

3.1 Volumetric Emission Model

We model the radiance emitted in a direction $\omega \in \mathbb{S}^2$ from a volumetric emitter by integrating an emissive density $\rho(\mathbf{x})$ along the ray from a shading point $\mathbf{x}_0$. The function $\rho : \mathbb{R}^3 \to \mathbb{R}$ defines the local radiative emission per unit length. The radiance is thus given by

$$L_e(\omega) = \int_0^\infty v(\mathbf{x}_0 + t\omega)\rho(\mathbf{x}_0 + t\omega)\, dt, \tag{3}$$

where the indicator function v restricts integration to the emitter interior Ω. This integral reduces to a sum over ray–volume intersection intervals $[r_i^-, r_i^+]$:

$$L_e(\omega) = \sum_i \int_{r_i^-}^{r_i^+} \rho(\mathbf{x}_0 + r\omega)\, dr. \tag{4}$$

In the case of uniform emission (*i.e.,* $\rho(\mathbf{x}) \equiv \text{const}$), we may normalize by total emitter volume V and set $\rho = 1/V$ to ensure unit total power. This yields the simplified form:

$$L_e(\omega) = \frac{1}{V} \sum_i (r_i^+ - r_i^-). \tag{5}$$

For general $\rho(\mathbf{x})$, we may estimate the radiance using Monte Carlo integration. Drawing one point $r^* \sim \mathcal{U}[r_i^-, r_i^+]$ per segment, where $\mathcal{U}$ denotes the uniform distribution, the unbiased estimator becomes:

$$\widehat{L}_e(\omega) = \sum_i (r_i^+ - r_i^-)\rho(\mathbf{x}_0 + r^*\omega). \tag{6}$$

When this is used in a Monte Carlo renderer, each sample's contribution must be weighted by the directional PDF $p(\omega)$,

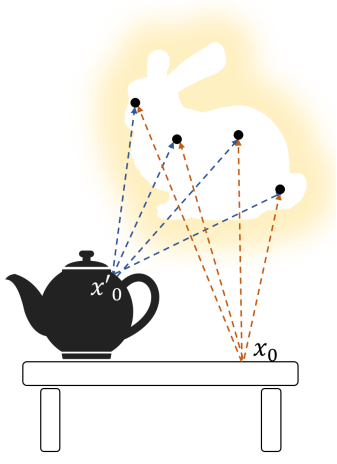


Fig. 2 A spatial-to-directional projection: sampling a point $\mathbf{x} \in \Omega$ inside the emitter (white bunny) and projecting toward the shading point $\mathbf{x}_0$ yields a direction $\omega = \mathbf{x} - \mathbf{x}_0/\|\mathbf{x} - \mathbf{x}_0\|$. This projection naturally adapts the sampling distribution to the emitter's geometry as seen from $\mathbf{x}_0$. At a different point $\mathbf{x}'_0$, the induced distribution differs due to perspective.

yielding a per-sample estimator:

$$\widehat{L}_e = \frac{1}{p(\omega)} \sum_i (r_i^+ - r_i^-)\rho(\mathbf{x}_0 + r^*\omega). \tag{7}$$

This formulation supports both constant and spatially varying emissive profiles within the same framework, enabling efficient importance sampling of arbitrarily complex volumetric emitters.

3.2 Projection-Based Directional Sampling

For SDF-defined volumes, directly generating a directionally adapted importance sample from a shading point is non-trivial due to the implicit nature of the geometry. However, it is often easier to sample a *spatial* point inside the volume using techniques like rejection sampling or analytic inversion.

Our key insight is to bridge this gap using a spherical projection from spatial samples to locally adapted directional ones (see Fig. 2). A point sampled uniformly inside the emitter induces a non-uniform distribution over directions at the shading point, which reflects the emitter's shape and relative position.

3.2.1 *The spatial-to-directional projection*

Let $\mathbf{x}_0 \in \mathbb{R}^3$ be the shading point. Given a point $\mathbf{x} \sim v(\mathbf{x})$ sampled uniformly within the volume V, the spatial distribution of $\mathbf{x}$ is

$$p(\mathbf{x}) = \frac{1}{V} v(\mathbf{x}) = \frac{1}{V}. \tag{8}$$

We define the corresponding direction by projecting to the unit sphere:

$$\omega = \frac{\mathbf{x} - \mathbf{x}_0}{\|\mathbf{x} - \mathbf{x}_0\|}, \quad \omega \in \mathbb{S}^2. \tag{9}$$

This defines a mapping from the volume to the unit sphere centered at $\mathbf{x}_0$, effectively converting uniform spatial samples into non-uniform direction samples that naturally account for the emitter's geometry.

Uniform sampling within V can be implemented through analytic inversion for primitive shapes or via rejection sampling for general SDFs. The resulting distribution over directions reflects both the geometry and spatial extent of the volume relative to the shading point.

3.2.2 *The induced directional PDF*

To derive the probability density function $p(\omega|\mathbf{x}_0)$ induced on the unit sphere, we perform a change of variables from Cartesian to spherical coordinates centered at $\mathbf{x}_0$:

$$\mathbf{x} = \mathbf{x}_0 + r\omega, \quad r \in [0, \infty), \quad \omega \in \mathbb{S}^2. \tag{10}$$

The volume element transforms as $d\mathbf{x} = r^2\, dr\, d\omega$, and the spatial PDF becomes:

$$f(r, \omega) = \frac{1}{V} v(\mathbf{x}_0 + r\omega) r^2. \tag{11}$$

Marginalizing over r gives the directional PDF:

$$p(\omega|\mathbf{x}_0) = \frac{1}{V} \int_0^\infty v(\mathbf{x}_0 + r\omega) r^2\, dr. \tag{12}$$

In practice, the volume V is often defined by a union of intervals $[r_i^-, r_i^+]$ along a ray. This leads to a discrete summation form:

$$p(\omega|\mathbf{x}_0) = \frac{1}{V} \sum_i \int_{r_i^-}^{r_i^+} r^2\, dr = \frac{1}{V} \sum_i \frac{(r_i^+)^3 - (r_i^-)^3}{3}. \tag{13}$$

This expression captures the contribution of each intersected segment of the emitter volume along the ray in direction ω.

To understand why our projection-based sampling is effective, consider the relationship between our derived PDF and the ideal PDF. For a uniform volumetric emitter, the ideal PDF is proportional to the radiance, which is the line integral of the volume indicator: $p^\star(\omega) \propto L(\omega) = \sum_i (r_i^+ - r_i^-)$.

Our method samples based on the volume projected onto the sphere, yielding a PDF proportional to the difference of cubes (Eq. 13): $p(\omega) \propto \sum_i ((r_i^+)^3 - (r_i^-)^3)$. Factorizing the cubic term for a single segment gives:

$$(r^+)^3 - (r^-)^3 = (r^+ - r^-)\left((r^+)^2 + r^+ r^- + (r^-)^2\right). \tag{14}$$

For an emitter at a distance D where the thickness is small relative to D (i.e., $r^\pm \approx D$), the second term approaches $3D^2$. Consequently, $p(\omega) \approx C(r^+ - r^-) \propto p^\star(\omega)$.

This implies that for distant emitters, our sampling distribution converges to the ideal proportional-to-radiance distribution. In the near field, the quadratic factor $((r^+)^2 + \dots)$ gently over-weights deeper parts of the volume, which is a safe heuristic that avoids variance spikes from under-sampling the emitter's core.

3.3 Integration Strategies for Common SDF Types

Having established the general projection-based framework, we now turn to the practical strategies used to compute volume integrals and ray intersections for different classes of SDFs. We focus on two representative cases: analytic primitives and general composite SDFs.

3.3.1 Analytic Sampling of Primitives

For common analytic SDF primitives such as *spheres*, *cubes*, and *tori*, both the total volume V and the ray–volume intersection intervals $[r_i^-, r_i^+]$ can be computed in closed form. These primitives serve as ideal testbeds for validating our method under controlled, low-variance conditions.

Sphere Consider a sphere of radius R centered at the origin, defined by the SDF

$$\phi(\mathbf{x}) = \|\mathbf{x}\| - R.$$

The volume is given by $V = \frac{4}{3}\pi R^3$, and a ray $\mathbf{x}_0 + r\omega$ intersects the volume by solving the quadratic

$$\|\mathbf{x}_0 + r\omega\|^2 = R^2,$$

which yields:

$$r^{\pm} = -(\omega\mathbf{x}_0) \pm \sqrt{(\omega\mathbf{x}_0)^2 - \|\mathbf{x}_0\|^2 + R^2}.$$

If a real solution exists, the ray intersects the sphere in a single segment $[r^-, r^+]$.

Importantly, *uniform volume sampling* is fully analytic. to sample a point uniformly inside the sphere, draw radius $r = Ru^{1/3}$, direction $\omega \sim \mathcal{U}(\mathbb{S}^2)$, and construct $\mathbf{x} = \mathbf{x}_{\text{center}} + r \cdot \omega$.

Torus For more complex primitives, such as a torus defined by

$$\phi(\mathbf{x}) = \left(\sqrt{x^2 + y^2} - R\right)^2 + z^2 - r^2,$$

the volume can also be computed analytically as $V = 2\pi^2 R r^2$. Ray–volume intersections are found by solving a quartic polynomial in r. While intersection and evaluation are exact, *uniform sampling within the torus volume* is nontrivial and typically requires rejection sampling.

Despite this, our projection-based method remains unbiased. The induced directional PDF can be computed via Eq. (13), and performs well even when rejection sampling is used for interior points.

3.3.2 Generalization to arbitrary SDFs

SDFs are capable of representing complex shapes formed via Boolean compositions, procedural modeling, or learned neural representations. While analytic formulae for V and ray intersections are not generally available in such cases, our framework remains applicable.

Volume Estimation The volume V can be estimated using Monte Carlo integration over a bounding region. Since $v(\mathbf{x}) \in \{0, 1\}$, the integration reduces to estimating the fraction of points inside the volume, yielding fast convergence.

Ray–Volume Intersections The intersection intervals $[r_i^-, r_i^+]$ can be computed using robust sphere tracing, which leverages the signed distance property to advance along rays while avoiding surface penetration.

Unbiasedness Estimating V with a Monte Carlo method introduces only a scalar normalization error:

$$\hat{L}_e(\omega) = \frac{V}{\hat{V}} L_e(\omega), \quad \frac{\hat{L}_e(\omega)}{L_e(\omega)} = \frac{V}{\hat{V}}. \tag{15}$$

This scalar factor does not affect the shape of the directional distribution, preserving the unbiased nature of the estimator and allowing seamless integration with other lighting strategies.

4 Practical Implementation Details

Our SDF emitter is designed for seamless integration into existing Monte Carlo path tracers. We implemented the method in a proprietary GPU production renderer, and also provide the implementation using LuisaCompute's DSL [29], available in the supplemental material. Below, we highlight key aspects of the implementation, with a focus on robustness, and compatibility with common rendering pipelines.

4.1 Directional-Safe Stepping Inside Volumes

In standard sphere tracing [30], ray advance is governed by the signed distance value $\phi(\mathbf{x})$, which guarantees a conservative step size while outside the implicit surface: when $\phi(\mathbf{x}) > 0$, the ray can safely advance by $\Delta t = \phi(\mathbf{x})$ without risk of penetrating geometry. However, this guarantee does not hold once the ray enters the interior of a solid volume.

Inside the volume, where $\phi(\mathbf{x}) < 0$, the value $|\phi(\mathbf{x})|$ continues to represent the minimal distance to the surface, but not necessarily along the ray direction. More precisely, the surface point minimizing $\|\mathbf{x} - \mathbf{y}\|$ such that $\phi(\mathbf{y}) = 0$ does not in general lie along the ray direction ω. As a result, stepping by $|\phi(\mathbf{x})|$ in direction ω may overshoot the nearest intersection, particularly for grazing incidence, leading to missed surface crossings and systematic undercounting of entry and exit events. A naïve remedy of simply using $|\phi(\mathbf{x})|$

inside the solid tacitly assumes that the field is a perfect unit-Lipschitz distance in every direction. However, in practice, this assumption rarely holds once constructive-solid-geometry, smooth blends, or voxel-reconstructed SDFs are involved, so the overshoot problem reappears.

To prevent such overstepping, we adopt a *directional-safe* stepping rule: while inside the volume ($\phi < 0$), we restrict the step size to

$$\Delta t = \min(-\phi(\mathbf{x}), \delta_{\max}),$$

where $\delta_{\max}$ is a user-defined maximum step size, chosen to be no greater than the thickness of the smallest volumetric feature to be resolved. This clamp ensures that even in worst-case directional misalignment scenarios, the ray progresses conservatively without skipping over narrow structures or internal components.

This modification has no effect outside the volume, where the standard step $\Delta t = \phi$ remains valid. In combination with root bracketing and bisection refinement, this directionally-safe rule yields a traversal algorithm that is both robust and unbiased, capable of accurately recovering all ray–volume intersections even in complex or fragmented geometries.

To our knowledge, this interior-specific correction to signed distance stepping has not been previously formalized in the literature. While similar ideas have been heuristically employed in real-time applications and shader code, we present it here as a principled and necessary component of accurate volumetric integration.

Choosing $\delta_{\max}$ in practice A practical guideline is to set $\delta_{\max}$ to the thickness of the thinnest volumetric feature the user wishes to resolve. For analytic primitives whose minimum wall thickness is known (e.g., the tube radius r of a torus), $\delta_{\max} = r$ suffices. For general or composite SDFs where the feature size is not known a priori, we recommend $\delta_{\max} \approx d_{\mathrm{AABB}}/K$, where d_{AABB} is the diagonal length of the emitter's axis-aligned bounding box and K is a user-chosen resolution factor; in our experiments, $K \in [20, 100]$ covers the range from coarse shapes to fine procedural detail. Setting $\delta_{\max}$ too large risks overshooting thin structures, while setting it too small increases traversal cost without improving image quality. In all scenes presented in this paper, a single value of $\delta_{\max} = 0.05$ (in normalized emitter-local coordinates) was used without per-shape tuning, and proved sufficient for all tested geometries.

We utilize a single tight axis-aligned bounding box (AABB) for each emitter. Once the ray brackets the volume interface during sphere tracing, we perform a few iterations of bisection to refine the intersection root to high precision, ensuring robust entry/exit determination.

4.2 Controllable Emitter Parameters

In digital content creation (DCC) pipelines, a *light source* typically refers to an artist-facing object with parameters like *intensity* and *transform*, abstracting the underlying emitter model. To ensure our SDF emitters integrate naturally into this paradigm, we support these controls directly.

Intensity We normalize emission by the volume V of the SDF-defined region. The user-defined intensity then acts as a scalar multiplier on the sampled radiance, preserving consistent brightness across emitters of different shapes or scales. For non-uniform emission, we scale each segment's contribution by the evaluated local color before normalization.

Transform We support arbitrary diagonal scaling, rotation, and translation. Let the world-space transform be represented by:

$$\mathbf{x} = \mathbf{R}\,\mathbf{S}\,\mathbf{x}' + \mathbf{t},$$

where $\mathbf{S}$ is a diagonal scale matrix, $\mathbf{R}$ is a rotation matrix, and $\mathbf{t}$ is a translation vector. The transformed SDF is evaluated using:

$$\phi'(\mathbf{x}) = \lambda_{\min}(\mathbf{S}) \cdot \phi\left(\mathbf{S}^{-1}\mathbf{R}^{-1}(\mathbf{x} - \mathbf{t})\right), \tag{16}$$

where $\lambda_{\min}(\mathbf{S})$ is the minimum singular value of $\mathbf{S}$, used conservatively to ensure a valid signed distance bound for safe sphere tracing.

During sampling and PDF evaluation, all geometry is interpreted in local (pre-transform) space. Volume sampling is performed in the unit emitter domain, then mapped to world space using the full transform. The directional PDF is corrected by the Jacobian determinant of the spatial transform:

$$p'(\omega) = \frac{1}{|\det(\mathbf{S})|}\, p(\omega'), \tag{17}$$

where ω' is the direction computed in the emitter's local frame.

This formulation supports non-uniform scaling and maintains physical correctness in sampling and evaluation, while exposing artist-friendly parameters at the light source level.

5 Experiments and Results

We performed a series of experiments to evaluate our method. All experiments were conducted on a PC equipped with an Intel i7-9700K CPU, 56GB DDR4 RAM, and an NVIDIA RTX 4070 GPU. Unless otherwise specified, all images were rendered at a resolution of 1024×1024. Comparisons are

always reported in an equal-samples per pixel (spp) manner, as the computational overhead of our method remains moderate in varying scenarios.

5.1 Variance Reduction

We thoroughly evaluated the variance reduction capabilities of our proposed SDF emitter sampling technique across a diverse set of volumetric light sources and 3D scenes. Our experiments demonstrate that by correctly sampling the induced solid-angle distribution, our method consistently achieves significantly lower variance compared to traditional approaches, leading to higher-fidelity rendering at equivalent sampling budgets. We performed the same rendering tasks (direct lighting only, to isolate the variance introduced by emitter sampling, while a production-level comparison with global illumination is shown in Fig. 1) with different approaches. Each approach used the same 64 spp budget. To avoid overexposure, we suppressed emission directly visible to the camera. We report error measured as mean average percentage error (MAPE) and FLIP [31]. We categorize our comparisons into two groups as explained below.

5.1.1 Analytic and Simple Implicit Shapes

For this group, we evaluated volumetric emitters defined by analytic SDFs (e.g., sphere, torus, cube). These shapes allow for a direct comparison to baselines where alternative sampling strategies might theoretically be feasible. Here, we report tests with a torus, which is slightly more complex than other analytic primitives. We compared three distinct sampling methods for direct lighting:

- *Uniform Directional Sampling* (Baseline 1): in this fundamental baseline, samples were drawn uniformly over the unit sphere from the shading point. This approach is highly general but completely ignores the light source's specific geometry and extent, leading to high variance, especially for small or distant emitters.
- *Emitter Surface Sampling* (Baseline 2): for shapes where a parametric surface exists (or can be easily derived, e.g., the boundary of a sphere or cube), we sampled points uniformly over the emitter's surface area. While common for surface lights, this method is fundamentally suboptimal for volumetric emitters as it only considers emission from the boundary, not the interior. It also requires explicit surface parameterization, which is often unavailable for implicit shapes.
- *Our SDF Emitter Sampling Method* (Analytic): our proposed technique, which sampled directly from the analytically derived solid-angle distribution induced by the volumetric SDF.

Across both analytic test scenes our solid–angle sampler attained the lowest error without extra rendering time. For Dragon, MAPE drops from 0.699 (uniform) and 0.078 (surface) to **0.024**, with FLIP falling from 0.448 and 0.054 to **0.021**; the Room scene exhibits the same pattern. Because Monte-Carlo error scales as $1/\sqrt{N}$, these improvements resulted in more than 2× faster convergence than surface sampling and two orders of magnitude faster than uniform sampling, yet the wall-clock cost remained essentially unchanged (4–6 s per render).

5.1.2 Complex and Composite Implicit Shapes

This group focuses on volumetric emitters defined by complex or composite SDFs (e.g., Boolean operations on primitives like a fractal shape or a 3D logo). For these shapes, direct surface parameterization or efficient surface sampling is generally intractable. Here, our method provides the crucial ability to sample directly from the volume. We compared three distinct approaches:

- *Uniform Directional Sampling* (Naive Baseline): as in Group 1, this acted as the most general fallback, where rays were sampled uniformly over the sphere from the shading point, and contributions were evaluated only if they intersected the emitter. This method made no attempt to target the light source and thus served as a lower bound on efficiency.
- *Bounding Volume-Based Sampling* (Approximate, Our Method): In this baseline, we applied our proposed sampling framework not to the exact complex SDF, but to a simple bounding volume (e.g., an axis-aligned bounding box or sphere) that enclosed the emitter. While this provided a form of importance sampling, it overestimated the emissive region, leading to wasted samples and higher variance compared to using the precise SDF. This demonstrated the benefit of using the exact SDF query capabilities.
- *Our SDF Emitter Sampling Method* (Precise): Our full approach, which robustly identified ray-volume intersections via sphere tracing within the precise composite SDF and sampled from the resulting accurate solid-angle distribution.

The results are shown in Fig. 3. For composite emitters the advantage of sampling the *exact* SDF is even more pronounced, because any mismatch between proposal and true shape wastes samples on empty space. In the Kitchen scene the uniform baseline results in a MAPE of 0.945, bounding–box importance sampling trims this to 0.079, but our precise sampler reaches **0.013**: a 6× improvement over the bound approach, and more than 70× over uniform, with FLIP

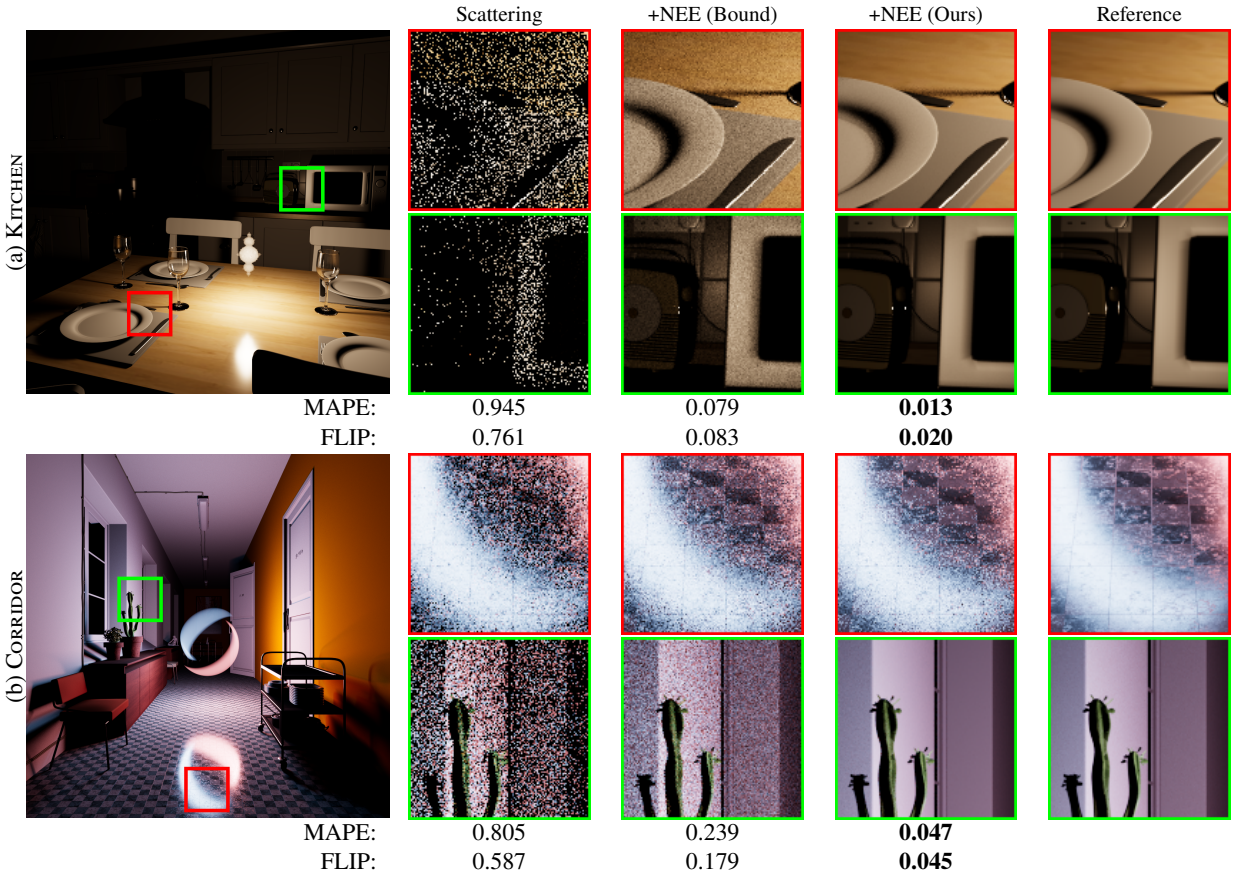


Fig. 3 Comparison of different sampling techniques for complex SDF emitters (a: fractal-shaped lamp, b: logo in room). Left to right: BSDF sampling only, added next-event estimation (NEE) using bounding volume approximation, NEE using our precise SDF emitter sampling, and high-sample-count reference. Our method (third column) achieves significantly lower noise and perceptual error (FLIP) than both the naive uniform and bounding volume-based approaches, demonstrating its effectiveness in targeting complex emission volumes.

dropping from 0.761 and 0.083 to **0.020**. The Corridor scene shows the same pattern. Crucially, evaluating the true SDF adds only moderate runtime overhead (less than 30% for all methods), so the extra geometric fidelity translates directly to substantial variance reduction at negligible cost.

5.1.3 Non-uniform Emission

As discussed, our method naturally extends to support non-uniform volumetric emission without introducing bias. In Fig. 4, we evaluate its variance performance under two types of non-uniformity. In Fig. 4(a), the emitter uses spatially varying color with uniform intensity, while in Fig. 4(b), the intensity falls off with distance to the surface. For comparison, we also render the same scene with uniform emission.

We observed an increase in error metrics (approx. 50% higher MAPE) for non-uniform emitters compared to their uniform counterparts. This behavior is expected: our sampling PDF is derived solely from the spatial geometry of the emitter (the indicator function $v(\mathbf{x})$) and assumes a constant emission density. When the emission $\rho(\mathbf{x})$ varies spatially, especially with high-frequency noise or strong contrast, the integrand $L_e(\omega) = \int v\rho$ diverges from our geometric PDF $p(\omega) \propto \int v$.

The resulting mismatch in weights leads to increased variance. However, because our method still robustly identifies the valid solid angle subtended by the volume, it remains significantly more efficient than uniform sphere sampling, which fails to capture the emitter's boundary entirely.

5.2 Volume Estimation Accuracy

To evaluate the accuracy and practical cost of our Monte Carlo volume estimation procedure, we tested a variety of representative implicit shapes with increasing complexity:

- A torus, for which the exact volume is known analytically;
- A union of two disjoint spheres;
- A procedurally defined fractal shape;
- A stylized 3D logo.

For each shape, we estimated the enclosed volume by uniformly sampling points within a tight bounding box and applying the binary indicator function described in § 3. We

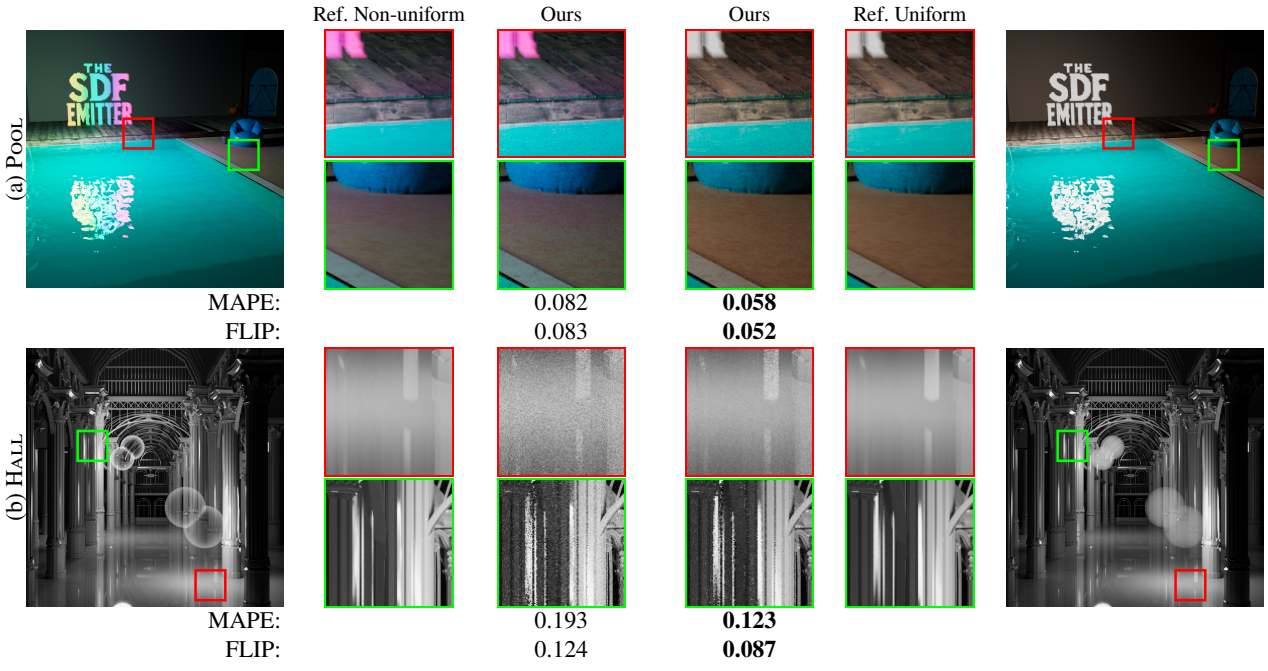


Fig. 4 Comparison of rendering results using an SDF emitter with non-uniform (left) or uniform (right) volumetric emission. In each scene, we render two variants: one with uniform intensity and one with spatially varying emission. (a): an SDF extruded from a 2D SDF map, with modulated emission color; (b): two Boolean combined spheres, with distance-based intensity falloff. Despite the added complexity of non-uniformity, our method displays only a modest increase in error for the same sampling budget.

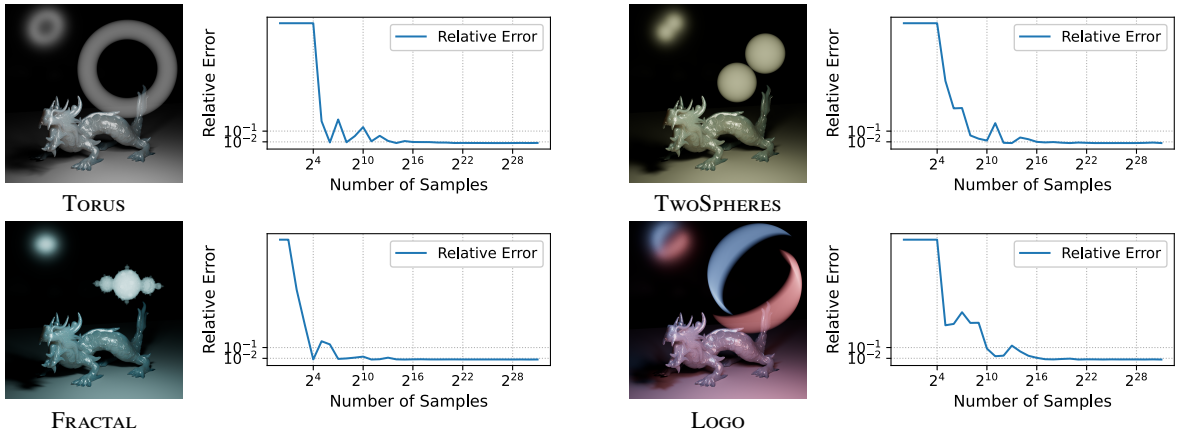
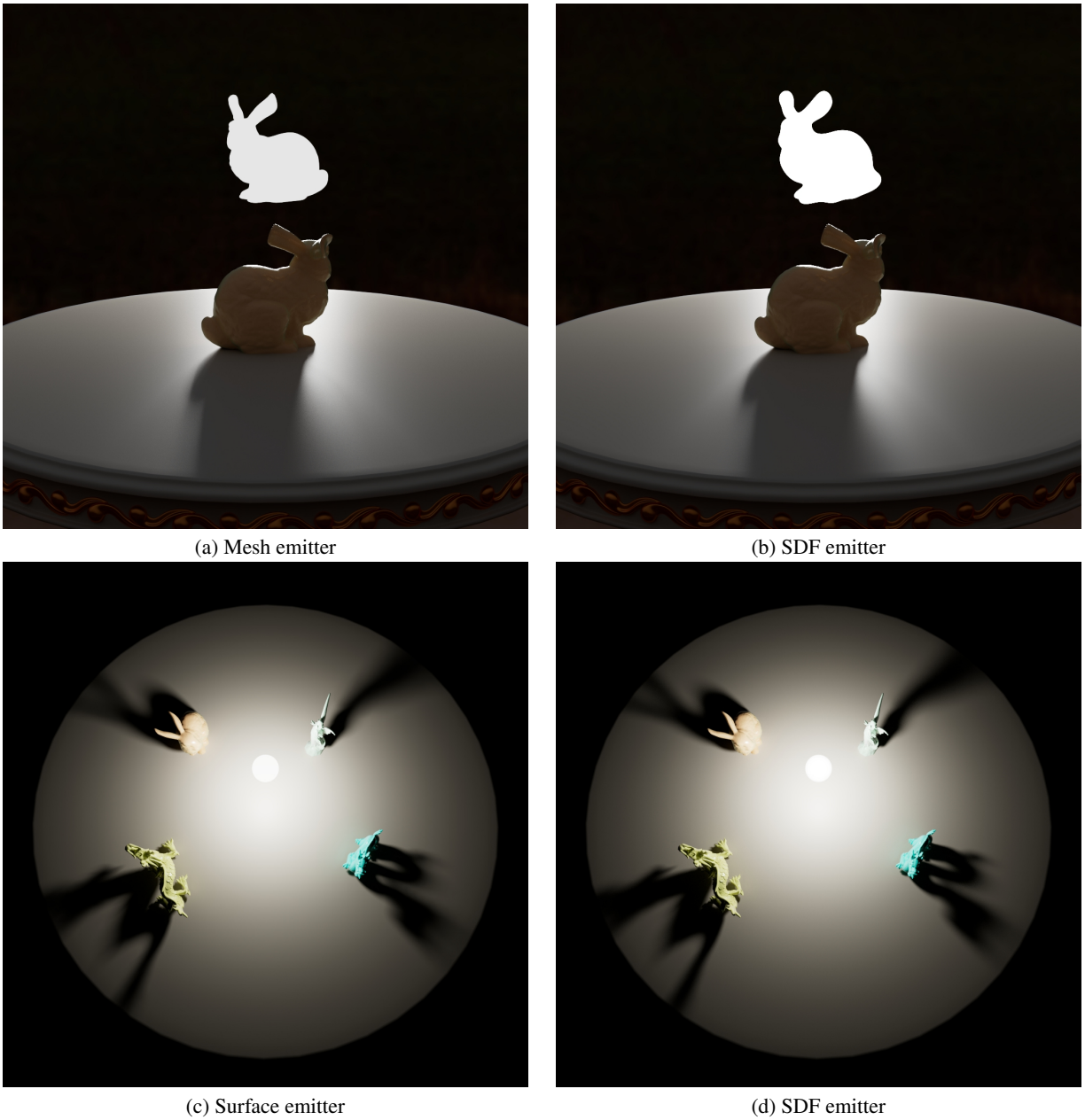


Fig. 5 Convergence plots of volume estimation for four representative implicit shapes. For each case, the relative error is plotted against the number of samples used during uniform point sampling inside a bounding box. For Torus and TwoSpheres we calculated the analytical volume as reference, while the reference for the others was calculated via Monte Carlo estimation using a significant number of samples. Although the estimation procedure used up to 2^{31} samples, we observe that all examples converged to less than 1% error by approximately 2^{18} samples. Our implementation calculated 2^{31} samples within 2 milliseconds, achieving an 0.1% error. This rapid convergence demonstrates the efficiency and practical viability of our volume normalization strategy, even for highly complex or fragmented implicit geometries.

recorded the estimated volume as a function of sample count, and report wall-clock timings for all evaluations.

Fig. 5 presents the convergence behavior across the four shapes. For the Torus and TwoSpheres cases, we compare to exact analytic volumes. For the Fractal and Logo cases (SDF courtesy https://www.shadertoy.com/view/4sl3zn), we estimated ground-truth values using 2^{40} samples. Despite the high sample ceiling, all shapes exhibited rapid convergence, with relative errors falling below 0.1% with approximately 2^{18} samples. This level of precision is sufficient to preserve perceptual consistency in rendered brightness across all tested scenes.

To assess runtime overhead, we also measured the total estimation time using 2^{31} samples. For all shapes, the GPU



(a) Mesh emitter (b) SDF emitter

(c) Surface emitter (d) SDF emitter

Fig. 6 Comparison of surface and volumetric SDF emitters. (a, b) A bunny-shaped emitter represented as a triangle mesh versus an SDF. The SDF is a SIREN network [4] with dozens of weights. Both produce similar appearance. (c, d) A spherical emitter represented as an area emitter and as a volumetric SDF emitter. Despite identical geometry, the shadow sharpness differs: the volumetric SDF produces a more focused shadow due to spatially varying emission along different ray paths. This highlights the expressive lighting behavior unique to volumetric emitters. Readers are referred to the supplemental material for a clearer comparison.

execution time remained below 2 milliseconds. This preprocessing cost is on par with alias table construction for environment lights, and can be amortized across frames or shared across emitter instances. As a result, our method enables fast and accurate volume normalization with negligible impact on overall performance.

5.3 Lighting Results

5.3.1 Compactness of SDF emitters

Signed–distance functions (SDFs) encode geometry implicitly, so a single analytic formula, or a tiny neural network with only a few dozen weights, can describe shapes that would require thousands of mesh triangles. Fig. 6(a,b) juxtapose a bunny emitter stored as a polygon mesh versus an SDF (SDF courtesy https://www.shadertoy.com/view/3lyyWc). The ren-

dered appearance is very similar, yet the SDF representation is two orders of magnitude smaller in memory, incurs no importance sampling overhead (alias table or tree-based representation), and can be edited procedurally at virtually no extra cost.

5.3.2 *Volumetric versus surface emission*

Fig. 6(c,d) use a simpler luminous sphere to reveal a fundamental behavioral difference between *surface* and *volumetric* lights. For the surface mesh light, radiance is uniform across the emitting area; therefore every point on the projected disk contributes equally and the resulting shadow shows a classic, evenly blurred penumbra. By contrast, our SDF emitter is filled with uniform *volumetric* emission. A shading ray aimed through the centre of the sphere travels a longer path inside the volume than one skimming the rim, so the effective radiant power varies with direction as the line-integral length $L(\omega)$. Consequently the solid-angle distribution is *denser* toward the centre, yielding a sharper umbra and a penumbra that fades more rapidly (see Fig. 6(d)). This directional modulation is captured automatically by our sampling PDF and incurs no additional cost——yet it produces shadow silhouettes that cannot be replicated by merely adjusting the intensity of a surface light.

Although SDFs could be converted to surface meshes prior to rendering, retaining the implicit representation during rendering provides practical benefits. Sampling SDF emitters avoids the construction and traversal of complex mesh-based light sampling structures and is often faster than sampling heavy mesh lights with comparable geometric detail. Moreover, SDFs naturally support volumetric emission and continuous shape manipulations. In practice, this yields lighting effects such as directionally modulated shadows and volumetric glows that surface emitters cannot reproduce by intensity reweighting alone, while remaining compact, procedural, and straightforward to integrate.

6 Discussion

6.1 Surface versus Volumetric Emission Sampling

For diffuse area lights, sampling uniformly over the emitter's solid angle [7, 9] leads to constant Monte Carlo weights. That is because the geometry term $G = \cos\theta/\|\mathbf{x} - \mathbf{x}_0\|^2$ appears both in the radiance expression and in the Jacobian from solid angle to surface area. The result is fast convergence with even a few samples. However, in an emissive volume, radiance in direction ω depends on how far that ray travels through the emitter:

$$\frac{d\Phi}{d\omega} \propto L(\omega), \quad L(\omega) = \sum_i (r_i^+ - r_i^-).$$

This means the optimal PDF should be proportional to $L(\omega)$. But uniform-solid-angle sampling does not account for this, causing high variance especially for thin or distant volumes. We sample a point uniformly inside the volume and project it to a direction. This yields a PDF (Eq. (13)) that increases with the cube of segment length, closely tracking $L(\omega)$ in most practical cases. When the emitter is far from the surface, our PDF approaches the ideal $p^\star(\omega) \propto L(\omega)$. Even nearby, it rejects directions that miss the volume and still favors longer paths, reducing variance. This explains the consistent speedups in Fig. 7.

6.2 Limitations and Failures

While our method performs robustly for a wide variety of emitter geometries, certain classes of implicit shapes and scenarios pose challenges worth highlighting.

6.2.1 *High-Frequency or Noisy SDFs*

In Fig. 8(a), we tested a procedurally modulated implicit surface with high-frequency perturbations. Such SDFs require very small sphere tracing steps to avoid overshooting during ray–volume intersection, resulting in increased traversal cost. In our current implementation, this leads to noticeably higher rendering times, although image quality remains consistent. We note that hierarchical SDF acceleration (e.g., grid-based min/max trees) could substantially alleviate this issue.

6.2.2 *Low Volume-to-Bounding-Box Ratio*

Emitters with sparse or thin structures incur high sampling overheads. In Fig. 8(c), a box-frame SDF fills only a small fraction of its bounding volume, causing a low acceptance rate in rejection sampling and thus higher computational cost. Tight bounding boxes and stratified sampling reduce this effect, but a more efficient sampling method remains an open problem.

6.2.3 *Lack of Surface Parameterization*

As with most implicit representations, SDFs do not provide inherent 2D surface parameterization. However, in a production rendering context, artists typically rely on *UV* coordinates to drive shader graphs for procedural textures, emission maps, or surface variation. Without *UV* coordinates, applying fine-grained material control to the emitter surface becomes difficult, or requires ad hoc workarounds, such as baking textures into volumes.

7 Conclusions and Future Work

We have introduced SDF emitters, a new class of light source for physically-based rendering, along with a practical method for efficiently rendering with them. By turning spatial volume

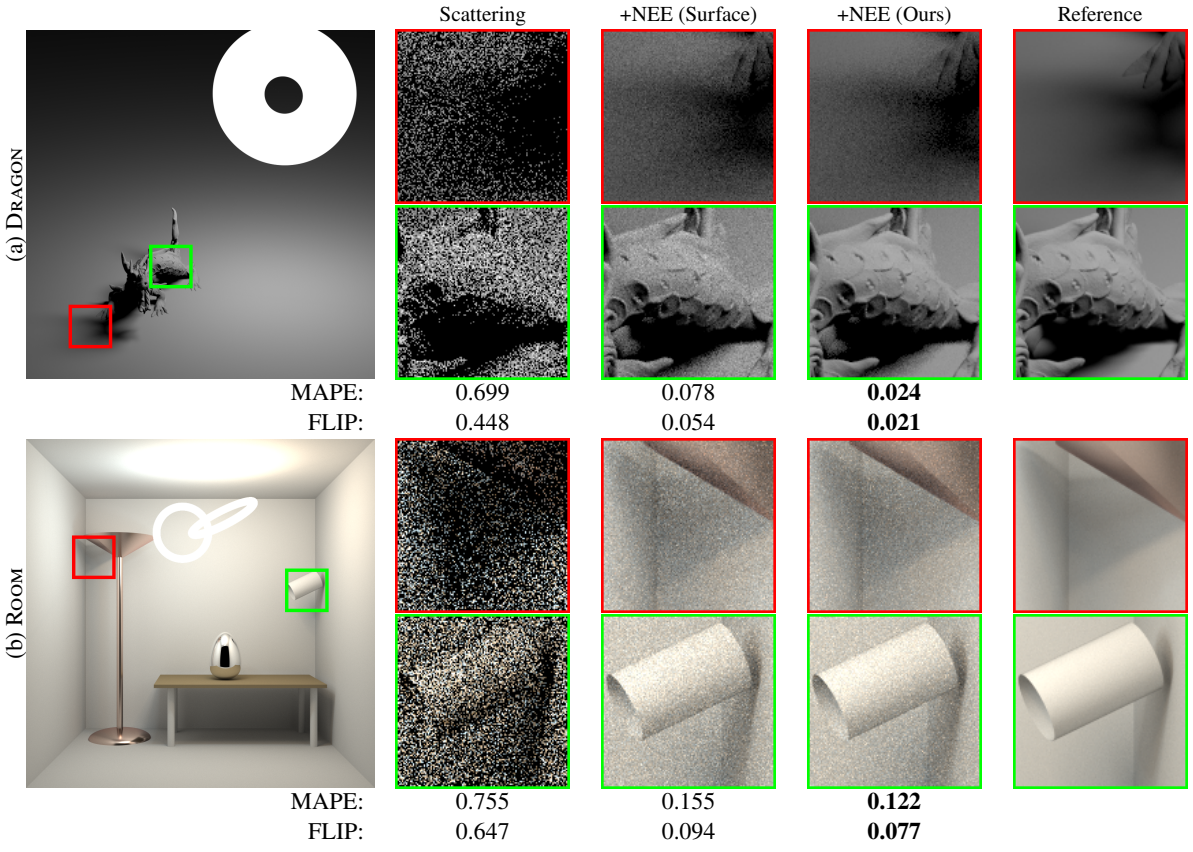


Fig. 7 Comparison of rendering quality for analytical SDF emitters, using three sampling strategies: uniform scattering (baseline), next-event estimation (NEE) with surface-area sampling, and NEE with our solid-angle sampling. NEE methods are combined with scattering via multiple importance sampling (MIS). The naive approach samples directions by uniformly sampling the SDF surface and projecting to solid angle, which ignores the interior structure. In contrast, our method enables analytically exact directional sampling over the volume. We report MAPE and FLIP (lower is better).

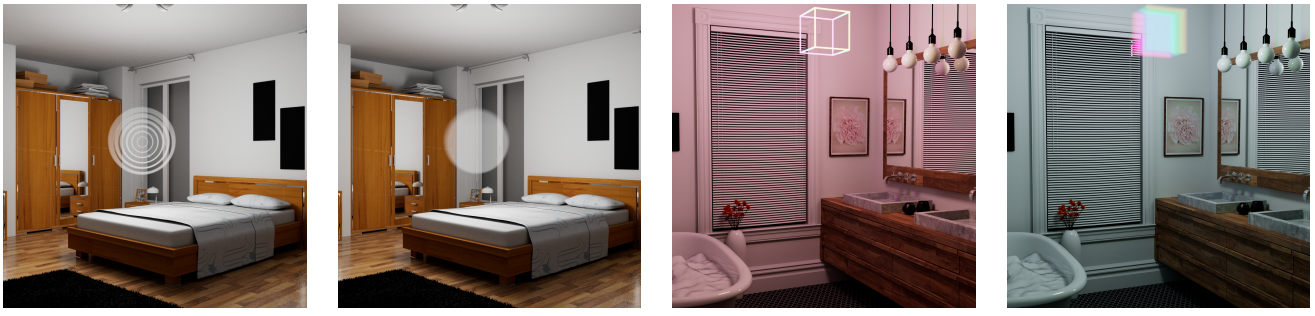


Fig. 8 Limitations of our method in challenging SDF scenarios. (a–b): High-frequency perturbations in the SDF (a) require much finer sphere tracing steps than a smooth surface (b), leading to a significant increase in rendering time (60 s vs. 12 s). (c–d): Emitters with low volume-to-bounding-box ratios, such as a thin frame (c), suffer from inefficient rejection sampling compared to solid volumes (d), doubling the computation time (20 s vs. 10 s).

sampling into solid-angle importance sampling, our approach enables complex SDF-based lights to be used directly in path tracing——without meshing, discretization, or precomputed tables. The result is both unbiased and efficient, with substantial variance reduction in scenes where conventional strategies struggle.

There is room to extend this work in several directions. One is cosine-weighted sampling, which would better match typical BRDFs near surfaces. Another is incorporating solid-angle–to–projected-area corrections to account for emitter orientation. Both are standard ideas in light sampling, but adapting them to arbitrary SDFs is nontrivial, and not essential

for most lighting tasks. Our focus has been on building a general, reliable foundation. Optimizations can follow.

References

[1] Yariv L, Gu J, Kasten Y, Lipman Y. Volume Rendering of Neural Implicit Surfaces. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, 2021, 4805–4815.

[2] Mescheder L, Oechsle M, Niemeyer M, Nowozin S, Geiger A. Occupancy Networks: Learning 3D Reconstruction in Function Space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, 4460–4470, doi:10.1109/CVPR.2019.00459.

[3] Wang P, Liu L, Liu Y, Theobalt C, Komura T, Wang W. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, 2021, 27171–27183.

[4] Sitzmann V, Martel JN, Bergman AW, Lindell DB, Wetzstein G. Implicit Neural Representations with Periodic Activation Functions. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020, 7462–7473.

[5] Jiang Y, Ji D, Han Z, Zwicker M. SDFDiff: Differentiable Rendering of Signed Distance Fields for 3D Shape Optimization. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020: 1251–1261.

[6] Mao S, Wu C, Shen Z, Wang Y, Wu D, Zhang L. NeuS-PIR: Learning Relightable Neural Surface Using Pre-Integrated Rendering. *Computational Visual Media*, 2025, 11(4): 727–744, doi:10.26599/CVM.2025.9450493.

[7] Arvo J. Stratified Sampling of Spherical Triangles. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1995, 437–438, doi:10.1145/218380.218500.

[8] Gamito MN. Solid Angle Sampling of Disk and Cylinder Lights. *Computer Graphics Forum*, 2016, 35(4): 25–36, doi:10.1111/cgf.12946.

[9] Ureña C, Fajardo M, King A. An Area-Preserving Parametrization for Spherical Rectangles. *Computer Graphics Forum*, 2013, 32(4): 59–66, doi:10.1111/cgf.12151.

[10] Crassin C, Neyret F, Lefebvre S, Eisemann E. GigaVoxels: Ray-Guided Streaming for Efficient and Detailed Voxel Rendering. In *Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games*, 2009, 15–22, doi:10.1145/1507149.1507152.

[11] Villemin R, Hery C. Practical Illumination from Flames. *Journal of Computer Graphics Techniques*, 2013, 2(2): 142–155.

[12] Simon F, Hanika J, Zirr T, Dachsbacher C. Line Integration for Rendering Heterogeneous Emissive Volumes. *Computer Graphics Forum*, 2017, 36(4): 101–110, doi:10.1111/cgf.13228.

[13] Wang Q, Luan F, Dai Y, Huo Y, Bao H, Wang R. A Biophysical-Based Skin Model for Heterogeneous Volume Rendering. *Computational Visual Media*, 2025, 11(2): 289–303, doi:10.26599/CVM.2025.9450360.

[14] Wu W, Wang B, Hašan M, Zhang L, Jin Z, Yan LQ. Efficient Participating Media Rendering with Differentiable Regularization. *Computational Visual Media*, 2024, 10(5): 937–948, doi:10.1007/s41095-023-0372-2.

[15] Veach E. Robust Monte Carlo Methods for Light Transport Simulation. Ph.D. thesis, Stanford University, 1997, doi:10.5555/927297.

[16] Hanika J, Droske M, Fascione L. Manifold Next Event Estimation. *Computer Graphics Forum*, 2015, 34(4): 87–97, doi:10.1111/cgf.12681.

[17] Zeltner T, Georgiev I, Jakob W. Specular Manifold Sampling for Rendering High-Frequency Caustics and Glints. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, 2020, 39(4): 1–15, doi:10.1145/3386569.3392408.

[18] Fan Z, Guo J, Wang Y, Xiao T, Zhang H, Zhou C, Chen Z, Hong P, Guo Y, Yan LQ. Specular Polynomials. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, 2024, 43(4): 1–13, doi:10.1145/3658132.

[19] Liang Y, Liu T, Huo Y, Wang R, Bao H. Adaptive Sampling and Reconstruction for Gradient-Domain Rendering. *Computational Visual Media*, 2024, 10(5): 885–902, doi:10.1007/s41095-023-0361-5.

[20] Diolatzis S, Gruson A, Jakob W, Nowrouzezahrai D, Drettakis G. Practical Product Path Guiding Using Linearly Transformed Cosines. *Computer Graphics Forum (Proceedings of Eurographics Symposium on Rendering)*, 2020, 39(4): 23–33.

[21] Heitz E, Dupuy J, Hill S, Neubelt D. Real-Time Polygonal-Light Shading with Linearly Transformed Cosines. *ACM Transactions on Graphics*, 2016, 35(4): 1–8, doi:10.1145/2897824.2925895.

[22] Zhu J, Bai Y, Xu Z, Bako S, Velázquez-Armendáriz E, Wang L, Sen P, Hašan M, Yan LQ. Neural Complex Luminaires: Representation and Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, 2021, 40(4): 1–12, doi:10.1145/3450626.3459798.

[23] Walter B, Fernandez S, Arbree A, Bala K, Donikian M, Greenberg DP. Lightcuts: A Scalable Approach to Illumination. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, 2005, 24(3): 1098–1107, doi:10.1145/1073204.1073318.

[24] Conty Estevez A, Kulla C. Importance Sampling of Many Lights with Adaptive Tree Splitting. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 2018, 1(2): 1–17, doi:10.1145/3233305.

[25] Moreau P, Pharr M, Clarberg P. Dynamic Many-Light Sampling for Real-Time Ray Tracing. In *Proceedings of High Performance Graphics 2019*, Eurographics Association, 2019, 21–26, doi:10.2312/hpg.20191191.

[26] Bitterli B, Wyman C, Pharr M, Shirley P, Lefohn A, Jarosz W. Spatiotemporal Reservoir Resampling for Real-Time Ray Tracing with Dynamic Direct Lighting. *ACM Transactions on*

Graphics (Proceedings of SIGGRAPH), 2020, 39(4): 1–17, doi:10.1145/3386569.3392481.

[27] Ouyang Y, Liu S, Kettunen M, Pharr M, Pantaleoni J. ReSTIR GI: Path Resampling for Real-Time Path Tracing. *Computer Graphics Forum*, 2021, 40(8): 17–29, doi:10.1111/cgf.14378.

[28] Lin D, Kettunen M, Bitterli B, Pantaleoni J, Yuksel C, Wyman C. Generalized Resampled Importance Sampling: Foundations of ReSTIR. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, 2022, 41(4): 1–23, doi:10.1145/3528223.3530158.

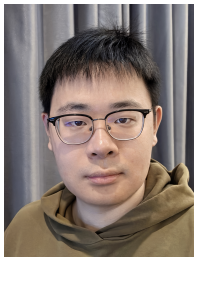
[29] Zheng S, Zhou Z, Chen X, Yan D, Zhang C, Geng Y, Gu Y, Xu K. LuisaRender: A High-Performance Rendering Framework with Layered and Unified Interfaces on Stream Architectures. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, 2022, 41(6): 1–19, doi:10.1145/3550454.3555463.

[30] Hart JC. Sphere Tracing: A Geometric Method for the Antialiased Ray Tracing of Implicit Surfaces. *The Visual Computer*, 1996, 12(10): 527–545.

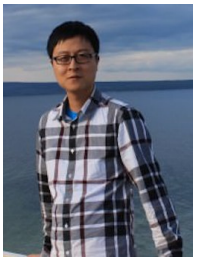
[31] Andersson P, Nilsson J, Shirley P, Akenine-Möller T. Visualizing Errors in Rendered High Dynamic Range Images. In *Eurographics 2021 – Short Papers*, The Eurographics Association, 2021, 25–28, doi:10.2312/egs.20211015.

Author biography

Jiawei Huang received a B.Eng. degree from Xiamen University, China, and M.S. and Ph.D. degrees in information science from Tohoku University, Sendai, Japan. He is currently an Associate Researcher at the International Digital Economy Academy (IDEA), where he focuses on rendering research and the development of a GPU-based production renderer. His research interests include physically based rendering, real-time graphics, and 3D motion capture.



Shaokun Zheng received his B.S. and Ph.D. degrees in Computer Science from Tsinghua University. His research interests are in sampling and reconstruction methods, high-performance rendering frameworks, and graphics programming languages.



Kun Xu is an Associate Professor in the Department of Computer Science and Technology, Tsinghua University, where he received his bachelor's and doctoral degrees in 2005 and 2009, respectively. His research interests include realistic rendering and image/video editing.

Yoshifumi Kitamura is a Professor in the Research Institute of Electrical Communication, Tohoku University, where he has also served as Deputy Director (2018–2025) and Director of the Interdisciplinary ICT Research Center for Cyber and Real Spaces since 2023. His research interests include interactive content design, human-computer interaction, 3D user interfaces, and telecommunication with nonverbal information. Previously, he was an Associate Professor at Osaka University and a researcher at ATR Communication Systems Research Laboratories. He received his B.Sc. (1985), M.Sc. (1987), and Ph.D. (1996) from Osaka University.

Jiaping Wang received his Ph.D. from the Institute of Computing Technology, Chinese Academy of Sciences in 2007. He previously served as a researcher at Microsoft Research Asia, where he focused on realistic rendering and appearance modeling. Since 2013, his research interests have shifted to distributed computing systems, high-performance GPU computing, and blockchain technology. He is currently affiliated IDEA.