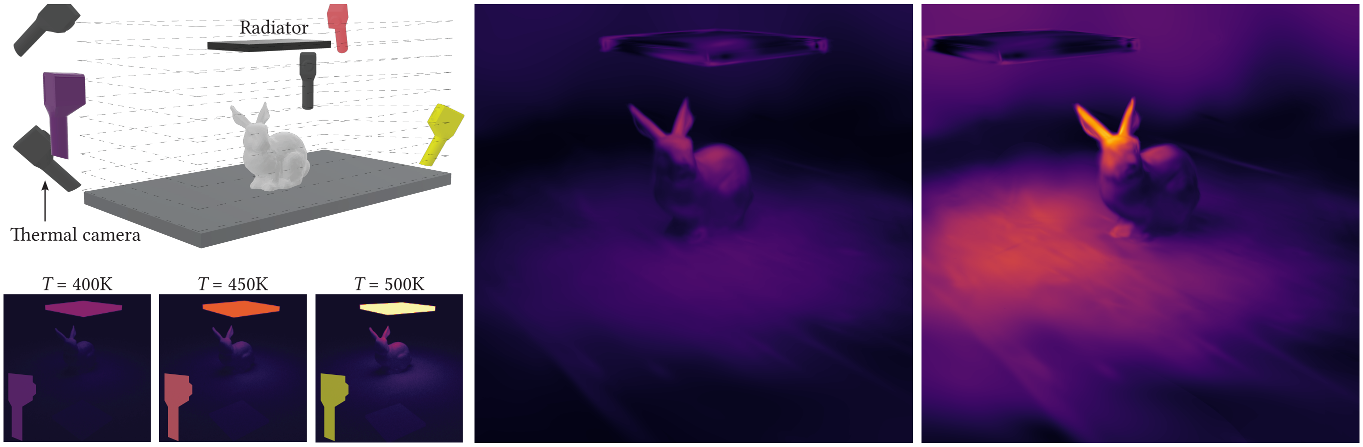
Inverse Radiative Transport for Infrared Scenes with Gaussian Primitives

LIU ZHENYUAN, ETH Zürich, Switzerland
BHARATH SESHADRI, ETH Zürich, Switzerland
GEORGE KOPANAS, Runway ML, United Kingdom
BERND BICKEL, ETH Zürich, Switzerland



Thermal capture setup

Predicted reflection

Predicted re-heating reflection

Fig. 1. Using our HiGS dataset comprised of 300 RGB-thermal image pairs taken at the radiator set to three different temperature, HiGS reconstructs the scene using 2D Gaussian primitives, analyzes the steady-state thermal radiative transport, and separates the reflection from the total radiation observed from the camera. It further enables *re-heating*: re-position the radiator and increase its temperature to 550K and emissivity to 0.9, and predict the reflections under the new heating condition.

Thermal imaging, as a promising approach for scalable and robust scene perception, is invaluable for many applications in various fields, such as architecture and building physics. Despite many recent works having demonstrated their capability to incorporate thermal images into radiance field methods, they typically do not explicitly model how radiation interacts and reflects within the scene before reaching the camera, which is essential for inferring thermal physics and properties of objects in a scene. Using Gaussian primitives as the scene representation, our method estimates surface temperature and material properties to generate infrared renderings that closely match the input images. Taking inspirations from radiosity and hemicube rasterization, our method decomposes the outgoing radiation from each Gaussian primitive into two parts: self-emission and reflection originating from other primitives and the environment. This formulation allows us to simulate radiation under novel heating conditions and to find the best-fit temperature and material parameters given thermal images. The method is verified using both synthetic and real capture datasets.

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

Authors' Contact Information: Liu Zhenyuan, ETH Zürich, Zürich, Switzerland, zhenyliu@ethz.ch; Bharath Seshadri, ETH Zürich, Zürich, Switzerland, seshadri@arch.ethz.ch; George Kopanas, Runway ML, London, United Kingdom, george@runwayml.com; Bernd Bickel, ETH Zürich, Zürich, Switzerland, bickelb@ethz.ch.

SA Conference Papers '25, Hong Kong, Hong Kong
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2137-3/2025/12
https://doi.org/10.1145/3757377.3763938

Additional Key Words and Phrases: radiance fields, radiative transport, thermal imaging

ACM Reference Format:
Liu Zhenyuan, Bharath Seshadri, George Kopanas, and Bernd Bickel. 2025. Inverse Radiative Transport for Infrared Scenes with Gaussian Primitives. In *SIGGRAPH Asia 2025 Conference Papers (SA Conference Papers '25), December 15–18, 2025, Hong Kong, Hong Kong*. ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3757377.3763938

1 Introduction

Thermal imaging, which leverages naturally emitted long-wave infrared (LWIR) radiation from objects, offers a promising approach for scalable scene perception, particularly under challenging environmental or illumination conditions. Infrared scene representations are also invaluable for many applications. For instance, in architectural and building physics applications, accurate thermal interpretations are crucial for detecting thermal anomalies and quantifying energy flows, as well as understanding the energy and comfort implications of design decisions.

Traditionally, the industry standard for analyzing thermal behavior involves manually modeling digital twins and physically simulating buildings and objects under various heat conditions. However, an emerging alternative seeks to circumvent this laborious manual 3D modeling process by reconstructing digital twins directly from photometric cameras and LiDAR sensors.

Playing a pivotal role in automating the modeling of objects and scenes, novel view synthesis and 3D scene representation techniques have garnered significant research interest. Methods such as Neural Radiance Fields (NeRFs) [Mildenhall et al. 2020] and 3D Gaussian Splatting (3DGS) [Kerbl et al. 2023] leverage image datasets and sensor information to generate high-fidelity results. While these foundational frameworks primarily focused on the visible spectrum, recent advancements have demonstrated their capability to incorporate thermal images alongside their visible-spectrum (also referred to as RGB) counterparts [Hassan et al. 2025; Lin et al. 2024; Lu et al. 2024; Xu et al. 2025]. This integration has necessitated addressing challenges such as the inherent lack of texture and lower resolution characteristic of thermal imagery. However, most existing methods in this domain are limited to predicting thermal images from novel viewpoints. They typically do not explicitly model how radiation interacts and reflects within the scene before reaching the camera. Crucially, however, this detailed radiative interaction information is essential for inferring the properties of the objects within a scene, including their emissivity.

To address this limitation, we develop *HiGS*, a novel steady-state radiation computation model. Inspired by the concept of radiosity, our model separates the total radiation observed by LWIR cameras into self-emission and reflection components. Building upon the existing Gaussian Splatting algorithm [Huang et al. 2024], we decompose the outgoing radiation from each Gaussian primitive into two distinct parts: self-emission and reflection originating from other Gaussian primitives and the environment. By employing the radiosity framework [Hadadan et al. 2021; Immel et al. 1986] coupled with our proposed rasterization-based hemicube technique [Cohen and Greenberg 1985], we compute the reflection at each Gaussian primitive based on the radiation contributions from other primitives within the scene. This formulation enables simulation of scene radiation under novel heating conditions (e.g., changes in self-emission or primitive positions) assuming the thermal conduction stay the same. Moreover, our formulation also allows us to solve the steady-state inverse radiative heat transport problem: inferring the temperature and material parameters that best fit given thermal images.

To summarize, we make the following contributions:

- A rasterization-based technique for computing reflection in a scene represented by self-emitting Gaussian primitives;
- An inverse radiative transfer method to infer material or surface properties of various objects in the scene, and radiation (thermal) energy exchanges between multiple objects;
- A multi-spectral (visible and thermal) novel-view synthesis dataset demonstrating radiative energy exchange between multiple objects.

We verify our method on both synthetic and real capture datasets. Our results show that we can reconstruct high-quality thermal 3D Gaussian primitives from multi-view images and achieve equal or better performance compared to other approaches. Finally, we demonstrate that our representation can be useful for downstream applications by exploring the thermal interactions in conjunction with the placement of objects and their materials.

2 Related Work

2.1 Radiance fields and inverse rendering

Radiance fields. Radiance field methods describe 3D scenes using a position- and direction- dependent color (or out-radiance) function $L_o(\mathbf{x}, \boldsymbol{\omega})$. Researchers have proposed a range of representations of L_o. Notable among these representations are neural radiance fields (NeRF) [Barron et al. 2022; Mildenhall et al. 2020] that use multilayer perceptron (MLP), Plenoxels [Fridovich-Keil et al. 2022] that use a grid and spherical harmonics, and 3D Gaussian Splatting (3DGS) [Kerbl et al. 2023] that rasterizes a set of 3D anisotropic Gaussian primitives. Subsequent variations of 3DGS include using 2D Gaussian disks to generate smoother surface reconstruction [Huang et al. 2024; Yu et al. 2024a] and ray-tracing Gaussian primitives [Condor et al. 2025; Moenne-Loccoz et al. 2024] for complex scenes and light transport parameters.

These methods reconstruct radiance fields from a collection of images, resulting in photorealistic renderings while maintaining real-time rendering speed. Recent research work reports good novel view synthesis performance using NeRF and 3DGS approaches to thermal images [Chen et al. 2025; Lin et al. 2024; Lu et al. 2024]. However, a shortcoming of these thermal radiance field approaches is that the critical scene parameters (material and surface properties) are baked into the representation of L_o. More than simply visualization, thermal images provide insight into energy flows within a scene. The current radiance-field approaches for thermal images limit applications to only visualization. Inferring scene parameters, and editing scenes, for example, changing locations and materials of objects in thermal scenes to predict the resulting radiance field is not possible using current methods.

Inverse rendering of radiance fields. Inverse rendering techniques solve for unknown scene parameters that match the renderings to the given images. Inverse rendering methods for NeRFs separate the appearance into channels of the predefined material model and therefore enable material editing [Jin et al. 2023; Srinivasan et al. 2021; Xu et al. 2023; Zhang et al. 2021a,b]. To achieve a better separation of geometry from NeRFs, researchers have used meshes for visibility queries [Sarkar et al. 2023], or introduced signed distance functions to replace NeRF's density function [Wang et al. 2021; Yariv et al. 2021]. Some addressed light transport phenomena, including reflection [Ge et al. 2023; Liu et al. 2023], self-emission [Jeong et al. 2024], subsurface scattering [Zhang et al. 2023], transparency [Cai et al. 2024], and translucency [Wang et al. 2023].

Similarly, inverse rendering of Gaussian Splats is an emerging field of research. GS-IR [Liang et al. 2024], and Relightable3DGS [Gao et al. 2024] capture surface material models by introducing physics-based rendering properties into the 3DGS rasterizer. GaussianShader [Jiang et al. 2024] supports reflective surfaces by incorporating a surface shading model. Some recent works on Gaussian Splats also support relighting with challenging volumetric effects such as subsurface scattering using one-light-at-a-time datasets [Dihlmann et al. 2025; Zhenyuan et al. 2025].

Most of the inverse methods above do not explicitly model self-emission. However, in thermal scenes, every object above 0 Kelvin is a thermal radiation emitter [Eastman 1936]. ESR-NeRF [Jeong

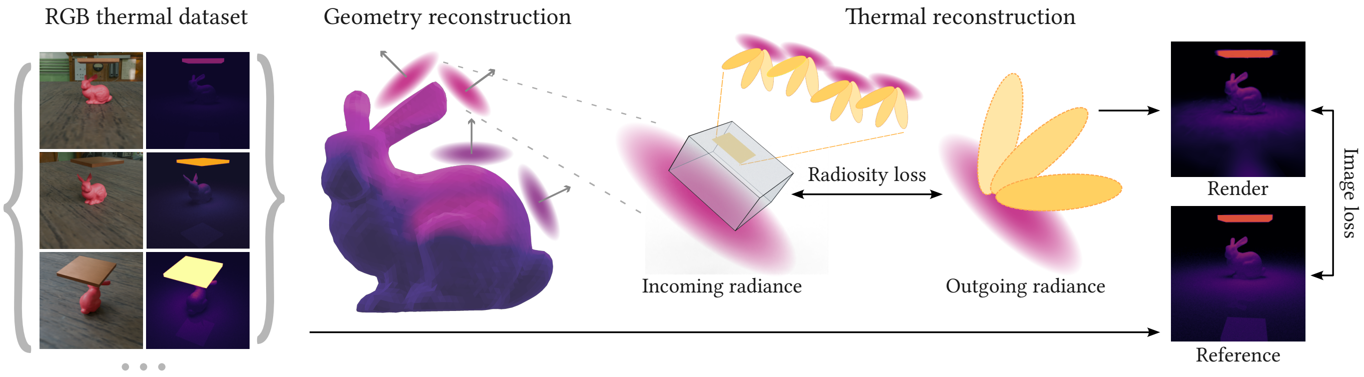


Fig. 2. **Overview**: HiGS takes as input dataset comprised of paired RGB thermal images. We first reconstruct a set of 2D Gaussian primitives that represent the geometry of the scene. Then, we compute the incoming radiance at each Gaussian primitive by rasterizing other primitives to the hemicube, and represent the outgoing radiance using Spherical Gaussian (SG) basis. The outgoing radiance and thermal properties are supervised by the image loss between the reference images and renderings, and a radiosity loss that promotes the separation of reflections.

et al. 2024] models the emission term explicitly but does not model the reflections which is one of the key focus of this work.

2.2 Radiative heat transfer

Radiosity. Initially developed to solve thermal radiative transport among discrete surfaces, radiosity was introduced to the computer graphics community to model light transport in diffuse-only scenes by Goral et al. [1984]. Later, radiosity methods have been extended to handle more complex, non-diffuse materials [Immel et al. 1986; Sillion et al. 1991]. To improve the computational efficiency of radiosity methods, various techniques have been proposed, such as adaptive meshing and refinement [Lischinski et al. 1992], and parallelizing view factor computation [Kramer et al. 2015]. Others used new radiosity approaches, such as removing explicit visibility computation [Dachsbacher et al. 2007], or use neural networks to approximate the solution to the radiosity equation without subdividing the surfaces [Hadadan et al. 2021].

Ray tracing and Monte Carlo methods. An alternative to the radiosity methods is ray tracing, commonly used in computer graphics to render photorealistic images for its superior scalability for detailed geometry and a wide range of materials [Pharr et al. 2023]. By raytracing temperature fields, radiative transport can be simulated without solving a global linear system as in radiosity methods [Pedro Aguerre and FernÃ¡ndez 2021]. Beyond rendering and radiative transport that this work focuses on, computer graphics community has also explored applying Monte Carlo methods to solve partial differential equations (PDEs), providing new ways to address thermal conduction and convection [Sawhney et al. 2023, 2022]. Bati et al. [2023] and Penazzi et al. [2024] further couple raytracers and Monte Carlo PDE solvers to simulate different modes of heat transfer in a unified framework. In thermal engineering, Monte Carlo methods are also widely used for modeling radiative transport at different scales and applications [Kang et al. 2005; Mazumder and Kersch 2000; Walters and Buckius 1994]. Advances in computer graphics have also introduced new tools for thermal engineering applications. For instance, Freude et al. [2023] adapted the concept of precomputed radiance transfer, a classical rendering technique, to develop an efficient thermal simulator.

Inverse radiative problems. Many thermal engineering problems in practice are *inverse*, meaning they require configurations to be computed which result in a particular thermal response. Common inverse radiative transport problems include estimating surface properties [Acosta et al. 2022; Pierre et al. 2022], temperature field reconstruction [Linhua et al. 1999; Liu et al. 2010], decoupling heat transfer mechanisms [Das et al. 2008; Lazard et al. 2000], geometry design [Abazid et al. 2016; Daun et al. 2003], and system efficiencies [Mann et al. 2022]. In thermal engineering, various inverse methods have been proposed for these problems, and we point the interested readers to a recent survey article [ErtÃ¼rk et al. 2023]. In computer graphics, differentiating Monte Carlo methods brings exciting new opportunities to solve these inverse problems with gradient-based optimization, such as differentiable raytracers [Nimier-David et al. 2020; Vicini et al. 2022] and MC PDE solvers [Miller et al. 2024; Yilmazer et al. 2024; Yu et al. 2024b]. In particular, Freude et al. [2025] devises a inverse radiative simulator to optimize the geometry of building facades.

3 Method

In this section, we review radiative heat transport theory (Section 3.1); discuss the calculation of the forward global radiative heat transport for Gaussian primitives via hemicubes and rasterization (Section 3.2); and the steps to solve the inverse radiative heat transport when using images captured under different heating conditions (Section 3.4). Figure 2 illustrates the pipeline.

3.1 Radiative heat transport theory

Emission. We consider the radiative transfer in the infrared band. According to the Stefan-Boltzmann law, objects at temperature T (in Kelvin) above absolute zero emit thermal radiation. An object's ability to emit thermal radiation is described by its *emissivity* $\varepsilon \in [0, 1]$. An ideal *black body* has $\varepsilon = 1$, and is non-reflective. An ideal *white body* has $\varepsilon = 0$, and is non-emissive. Objects in real life are usually gray bodies that have $0 < \varepsilon < 1$. In this work, we assume constant emissivity across the LWIR spectrum, although it is a wavelength-dependent property. For a point $\mathbf{x}$ on the surface of an object, the total thermal flux leaving $\mathbf{x}$ reads [Planck 1906, Eq. 78]

$$Q(\mathbf{x}) = \sigma\varepsilon(\mathbf{x})T(\mathbf{x})^4, \tag{1}$$

where σ is the Stefan-Boltzmann constant. The thermal flux can be also expressed as the integral of the emitting radiance $L_e(\mathbf{x}, \boldsymbol{\omega}_o)$ over all exiting directions $\boldsymbol{\omega}_o$ in the hemisphere $\mathcal{H}^2$ about the outward-pointing surface normal $\mathbf{n}$, $Q(\mathbf{x}) = \int_{\mathcal{H}^2} L_e(\mathbf{x}, \boldsymbol{\omega}_o) \cos\theta \, \mathrm{d}\boldsymbol{\omega}_o$, where θ is the angle between $\boldsymbol{\omega}_o$ and the surface normal $\mathbf{n}$.

If we assume uniform diffuse emission, the radiance exiting is constant over all directions from $\mathbf{x}$, and the emitted radiance is:

$$L_e(\mathbf{x}, \boldsymbol{\omega}_o) = Q(\mathbf{x})/\pi = \frac{\sigma}{\pi}\varepsilon(\mathbf{x})T(\mathbf{x})^4. \tag{2}$$

Reflection. When emitted radiation from one object hits the surface of another object, a part of the radiation is absorbed by the object, and assuming opaque surfaces, the remaining incident radiation is reflected.

In this work, we only consider objects and scenes at *steady state* i.e. heat balance is achieved and temperatures of the objects remain unchanged. By Kirchhoff's law of radiation, the *absorptivity*, or the ratio of absorbed radiation over the total irradiance, equals the emissivity. Therefore, $(1 - \varepsilon)$ of the incoming radiation is reflected off the surface. Then, the original Rendering Equation [Kajiya 1986] that describes reflection L_r and total outgoing radiation L_o can be written as follows [Bati et al. 2023, Eq. 8]:

$$L_o(\boldsymbol{\omega}_o) = \varepsilon L_e(\boldsymbol{\omega}_o) + \underbrace{(1-\varepsilon)\int_{\mathcal{H}^2} f(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) L_i(\boldsymbol{\omega}_i) \, \mathrm{d}\boldsymbol{\omega}_i^{\perp}}_{=:\, L_r(\boldsymbol{\omega}_o)}, \tag{3}$$

where $\mathrm{d}\boldsymbol{\omega}_i^{\perp}$ is the projected solid angle measure, f is the bidirectional reflectance distribution function (BRDF), L_i is the incident radiance arriving at $\mathbf{x}$. For notational brevity, we omit the dependence on $\mathbf{x}$ of the emissivity, BRDF, and radiance terms.

3.2 2D thermal Gaussian primitives

We develop our scene representation on top of 2D Gaussian splats (2DGS) proposed by Huang et al. [2024]. In 2DGS, the scene is represented by a set of 2D Gaussian functions embedded in three dimensions. Each primitive has a disk-like shape, and is characterized by the center of the Gaussian kernel $\mathbf{x}$, opacity o, and the RGB color function $\mathbf{c}(\boldsymbol{\omega}_o)$ represented using spherical harmonics. From the covariance matrix, the normal vector $\mathbf{n}$ of the primitive can also be determined up to a sign flip. We always orientate $\mathbf{n}$ so that it faces the camera, following 2DGS [Huang et al. 2024]. Given a camera $\boldsymbol{\omega}_o$, a set of primitives can be rasterized into an image by projecting the primitives onto the camera plane, sorted by their distance to the camera, in ascending order. For each pixel $\mathbf{p}$ of the render image I, we compute the blending weight $w_{g,\mathbf{p}}$ of each primitive g from front to back, to determine the pixel color $I_\mathbf{p}$ is determined as follows:

$$w_{g,\mathbf{p}} = o_i\hat{G}_g \prod_{j=1}^{g-1}(1 - o_j\hat{G}_j); I_\mathbf{p} = \sum_g w_{g,\mathbf{p}}\mathbf{c}_g(\boldsymbol{\omega}_o), \tag{4}$$

where $\hat{G}_i$ is the lower-bounded Gaussian function in 2DGS [Huang et al. 2024, Eq. 11]. The rasterization process is differentiable, and therefore we use an image-loss function between the renderings and the ground truth to find the best primitive parameters that represent the scene.

2DGS represent scenes by using RGB images, while yielding additional geometry information (e.g. surface normals). Chen et al. [2025] used a straightforward approach of adding an extra thermal channel to the color function per primitive for novel view synthesis of thermal scenes. However, this approach bakes global radiative transfer information in the training data into the radiance function, and as a result, makes it impossible to edit thermal scenes and propagate changes to other primitives e.g., re-positioning heat sources and/or sinks, or *reheating* objects to different temperatures. This shortcoming is analogous to the global illumination being baked into the color function causing difficulty for *re-lighting* tasks in visible-spectrum radiance field rendering.

To enable *re-heating*, we extend each Gaussian primitive with additional thermal-related parameters, namely emissivity ε, temperature T, diffuse coefficient d and roughness s. We parameterize the thermal radiance function $L_o(\boldsymbol{\omega}_o)$ by all primitives' ε, T, d and s. We then render thermal images in the same 2DGS rasterization process, with thermal radiance, instead of RGB color.

In radiative heat transfer, L_o is computed from the thermal parameters by Equation 3. While the emission term is relatively straightforward to compute, the reflection term requires recursively evaluating Equation 3, leading to a complex integral. In Monte-Carlo path tracing, this is typically done by sampling and evaluating one light path at a time. Recent works [Condor et al. 2025; Moenne-Loccoz et al. 2024] use ray-Gaussian intersection operators, providing a raytracing alternative to evaluate Equation 3 for Gaussian-represented scenes. In this work, we choose 2DGS rasterization for its speed advantage over the raytracing methods.

3.3 Reflection with hemicubes

Our aim is to compute reflected radiance L_o as defined in Equation 3, using a method compatible with rasterization. We draw inspiration from cube mapping, a common real-time rendering technique used to approximate environmental lighting and reflections. When light sources are assumed to be infinitely far away—such as in environment mapping—the incoming radiance at any point on an object can be approximated using a single cube map. The cube map consists of six images each representing incoming light from one of the six directions around the object.

However, the assumption of infinitely distant sources does not hold for thermal Gaussian splats. Each Gaussian primitive emits thermal radiation, and nearby primitives can significantly influence one another. As a result, the incoming radiance can vary greatly between different locations.

Consequently, we use a cube map *per Gaussian primitive* to compute the local incoming radiance $L_i(\boldsymbol{\omega}_i)$. Among possible ways of orienting a cube map at a Gaussian, we choose to orient the cube map such that one of the faces (the front face) points in the normal direction. If we consider a point on the surface of an object, radiance only arrives from the outside of the object. Therefore, we use the half of the cube map, or *hemicube* [Cohen and Greenberg 1985].

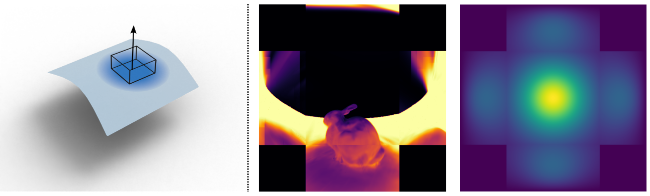


Fig. 3. Hemicubes are placed at Gaussian primitives orientated by the normal shown as black arrow (left), and hemicubes can be used to represent the incoming radiance L_i (middle) and the projected solid angle $\mathrm{d}\boldsymbol{\omega}^{\perp}$ (right).

These hemicubes can be viewed as a discretization of the hemisphere $\mathcal{H}^2$, with each pixel corresponding to a point $\boldsymbol{\omega} \in \mathcal{H}^2$. Then we can use the hemicubes to represent the incoming radiance $L_i(\boldsymbol{\omega}_i)$ at each primitive by rasterizing onto each side of the hemicube. For simplicity, we use the same intrinsic camera parameters for the rasterization, the 64x64 resolution for the faces, and the field of view of 90°, and we empirically discard Gaussian primitives that are too close to the hemicube (1% of the diagonal of the scene bounding box). For now, we pretend to know the outgoing radiance of other Gaussian primitives, then we can rasterize other primitives on the five faces of the hemicube by 2DGS rasterization procedures, as depicted in Figure 3. The value of each pixel represents the incoming radiance from the corresponding direction. We can also compute the projected solid angle measure of each pixel. Specifically, the cosine term being the dot product between the surface normal and the unit vector to each hemicube pixel's center, and the solid angle measure is the projected area of each pixel onto the unit sphere. Figure 3 shows an example of how L_i and $\mathrm{d}\boldsymbol{\omega}_i^{\perp}$ are represented by hemicubes.

Then, to compute the outgoing radiance in direction $\boldsymbol{\omega}_o$, we need to choose a BRDF model f. In the scope of this work, we consider a Cook-Torrance BRDF model [Cook and Torrance 1982]:

$$f(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = d f_d + (1 - d) f_s(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o), \tag{5}$$

where $d \in [0, 1]$ is the diffuse coefficient of the primitive, $f_d = 1/\pi$ is the diffuse component (Lambertian), and f_s is the specular component with GGX as the normal distribution function parametrized by the roughness s [Pharr et al. 2023, Sec. 9.6]. We use Schlick's approximation for the Fresnel term [Schlick 1994], with fixed Fresnel factor at normal incidence $f_\lambda = 0.8$. Theoretically, other BRDF models can also be used.

Then, we will discuss how to decide the outgoing radiance of a Gaussian primitive. One strategy would be to evaluate the thermal rendering equation recursively, in a similar fashion to solving the recursive integral using multiple bounces in path tracing. In our hemicube rasterization framework, computation of the incoming radiance could be performed by first computing all the Gaussian primitives that are visible to the target Gaussian and then recursively evaluate the outgoing radiance from each visible Gaussian primitive to the target primitive. However, this strategy is prohibitively expensive, as each time we need to evaluate L_o of all the tens of thousands of visible primitives. To make the computation of the reflection more tractable, we use the idea of radiance cache [MÃ¼ller et al. 2021]. Instead of computing the reflection for each primitive g recursively, we now use a set of spherical Gaussian basis functions to represent the reflection at g, denoted by $L_{r,g}$. Each primitive is now associated with m spherical Gaussian basis functions (in our case, $m = 4$). Each spherical Gaussian has the shape of a lobe and is defined as $G(\boldsymbol{\omega}_i; \boldsymbol{\mu}, \alpha, \lambda) = \alpha e^{\lambda(\mu \cdot \omega_i - 1)}$, where $\mu \in \mathcal{S}^2$ is the lobe axis, $\alpha \in \mathbb{R}_+$ is the amplitude of the lobe, and $\lambda \in \mathbb{R}_+$ is the sharpness. We use $m = 4$ such spherical Gaussians to represent the reflected radiance for each Gaussian primitive g:

$$\tilde{L}_{r,g}(\boldsymbol{\omega}_o) = \sum_{i}^{m} G(\boldsymbol{\omega}_o; \boldsymbol{\mu}_{i,g}, \alpha_{i,g}, \lambda_{i,g}) \ . \tag{6}$$

Now, the problem becomes how to find the Spherical Gaussian parameters, namely, $\boldsymbol{\mu}_{i,g}$, $\alpha_{i,g}$ and $\lambda_{i,g}$ that best represent $L_{r,g}$. Here we take a radiosity approach inspired by Hadadan et al. [2023]. At each Gaussian, given an outgoing direction, the reflection represented by the radiance cache should equal the reflection computed via hemicube rasterization as described. Then we can use this relation to build a reflection residue term:

$$r(\boldsymbol{\omega}_o) = \tilde{L}_r(\boldsymbol{\omega}_o) - (1 - \varepsilon) \int_{\omega_i} L_i(\boldsymbol{\omega}_i) f(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \mathrm{d}\boldsymbol{\omega}_i^{\perp} . \tag{7}$$

$L_i(\boldsymbol{\omega}_i)$ is computed by hemicube rasterization with the color of each Gaussian primitive being $\tilde{L}_r(-\boldsymbol{\omega}_i)$. To find the best SG parameters given L_e, ε, d and s, we can minimize the sum of the L2 norm of residue at all primitives,

$$\min_{\mu,\alpha,\lambda} \mathcal{L}_r = \sum_g \left| r_g \right|^2 , \tag{8}$$

where $|r|^2 = \int_{\mathcal{H}^2} r^2(\boldsymbol{\omega}) \mathrm{d}\boldsymbol{\omega}$. Then, the final outgoing radiance for each primitive is the sum of self-emission and reflection from the cache: $L_o(\boldsymbol{\omega}_o) = \varepsilon L_e(\boldsymbol{\omega}_o) + \tilde{L}_r(\boldsymbol{\omega}_o)$.

3.4 Optimization

After solving the forward problem, namely, find the radiance L_o given the scene descriptions including geometry (spatial distribution of the Gaussian primitives) and materials (L_e and ε, d and s), we will explain how to solve the inverse problem we are interested in: how to infer geometry and thermal properties given observations of L_o in the form of images $\mathbf{I}^*$ taken by thermal cameras. In our settings of using Gaussian primitives as the scene representation, we take a two-phase approach: in the first phase, optimize for the geometry parameters of the primitives; and in the second phase, fix the geometry, and recover the primitives' thermal properties and separate reflections and self-emissions.

Geometry reconstruction. The goal of the first phase is to find the geometry-related parameters of our Gaussian primitives, including position $\mathbf{x}$, covariance Σ, opacity α, normals $\mathbf{n}$. To find these parameters, we use the loss functions and optimization process presented in 2DGS [Huang et al. 2024, Sec. 5]. For enhancing the quality of the geometry reconstruction, we use paired RGB and LWIR images for training. LWIR images usually lack background textures if they are viewed in linear space, which is necessary for our radiation physics formulation. In this case, RGB images provide the detailed textures that lead to better geometry reconstruction results with better aligned surface normals and fewer floaters. To improve reconstruction quality with thermal input, we use the following scheduling strategy: training begins with supervision from RGB images only, and thermal image supervision is gradually introduced over time. More specifically, in each training iteration we compute two 2DGS loss terms, one for RGB images and one for thermal images. Next, the two loss terms are summed: the RGB loss term is weighted by a constant factor of 1, while the weight of the thermal loss term increases linearly from 0 to 1 between iterations 10,000 and 20,000 This approach addresses the lack of texture in the thermal images, which otherwise leads to premature removal of many Gaussian primitives during early training and ultimately results in many floaters.

Thermal reconstruction. In Equation 8, we derive the residue term $\mathcal{L}_r$ that drives the radiance cache to account for global illumination effects such as reflection given the scene description. On top of that, by adding image supervision, we can further promote the Gaussian primitives and the radiance cache to render images $\mathbf{I}$ that are similar to the reference images $\mathbf{I}^*$. Specifically, we use the image loss proposed by Kerbl et al. [2023],

$$\mathcal{L}_I = \mathcal{L}_1(\Gamma(\mathbf{I}) - \Gamma(\mathbf{I}^*)) + \mathcal{L}_{\text{D-SSIM}}(\Gamma(\mathbf{I}) - \Gamma(\mathbf{I}^*)), \tag{9}$$

where $\Gamma(\cdot) = (\cdot)^{1/2.2}$ is the gamma function that empirically increases the intensity of the reflection shown in the images. Minimizing the image loss promotes the radiance cache to converge to generate renderings that look similar to the training images. The image loss and the residue loss are combined to form our final loss function,

$$\mathcal{L} = w_I \mathcal{L}_I + w_r \mathcal{L}_r, \tag{10}$$

where w_I and w_r are the weights for the corresponding loss term. The optimizable parameters are the SG cache parameters ($\boldsymbol{\mu}_{i,g}$, $\alpha_{i,g}$ and $\lambda_{i,g}$) and the material parameters: emissivity ε, temperature T, roughness s and diffuse coefficient d.

Sampling $\mathcal{L}_r$. In our loss function, $\mathcal{L}_I$ is defined over the whole training dataset, and $\mathcal{L}_r$ is defined over all the Gaussian primitives. To make the optimization tractable, we use a sampling strategy to minimize $\mathcal{L}_I$ iteratively using gradient-based methods. During each iteration, we sample one training camera for stochastic evaluation of $\mathcal{L}$. Then, we sample a batch (size 128) of Gaussian primitives and a direction $\boldsymbol{\omega}_o$ for each primitive to calculate $\mathcal{L}_r$. A straightforward strategy is to uniformly sample the primitives and directions. However, we notice the uniform sampling strategy leads to significantly slow convergence. As a result, we prioritize primitives that are most visible to the training camera we have sampled in this iteration. Specifically, we project the primitives onto the training camera, and then compute their rendering weights $w_{g,\mathrm{p}}$ by Equation 4. Then we sum the weights over all the pixels in the training camera to obtain the sampling weight $w_g = \sum_{\mathrm{p}} w_{g,\mathrm{p}}$ for each g. Then, we sample the primitives with a probability proportional to the sampling weights. After selecting the primitives, we use the direction from each primitive g to the camera as $\boldsymbol{\omega}_o$ for the evaluation of $r_g(\boldsymbol{\omega}_o)$. Algorithm 1 provides the pseudocode of the training iteration.

With the stated sampling strategy, we use Adam [Kingma and Ba 2015] to minimize $\mathcal{L}$ for 300,000 iterations.

Algorithm 1 TrainingIteration

```
Require: Training image $I^*$, camera $\mathbf{c}$, batch size $b$:
  Evaluate rendering weight $w_g = \sum_{\mathrm{p}} w_{g,\mathrm{p}}$          ▷ Eq. 4
  for each primitive $g$ do
      Evaluate $L_o(\boldsymbol{\omega}_{g\to c})$          ▷ Direction from c to $g$
      Render $I$. Evaluate $\mathcal{L}_I$          ▷ Eq. 4 & 9
  end for
  Sample $b$ primitives, weighted by $w_i$
  for each sampled primitive $g$ do
      for each other primitive $k$ do
          Evaluate $L_o(\boldsymbol{\omega}_{k\to g})$          ▷ Direction from $k$ to $g$
      end for
      Evaluate $r(\boldsymbol{\omega}_{g\to c})$, $\mathcal{L}_r$          ▷ Eq. 7 & 8
  end for
  Evaluate $\mathcal{L}$          ▷ Eq. 10
  BackPropagate from $\mathcal{L}$
```

4 Dataset

Our HiGS dataset comprises 4 synthetic scenes and 1 real-world scene. The scenes demonstrate the emission and reflection of thermal radiation. Each scene contains a singular heat source that emits radiation and one or more objects that reflect the radiation from the heat source.

Synthetic scenes. We provide four synthetic scenes, and each of them has three pre-set heat-source temperature conditions. For each heating condition, we render 100 paired RGB and thermal images, resulting in 300 image pairs for the whole scene. The cameras are posed evenly over a hemisphere, pointing at the center of the scene for consistency. Some examples of the RGB-thermal image pairs can be seen in Figure 2. We model our scene and generate the RGB renderings in *Blender* 4.4. The thermal simulation is performed in *Stardis 0.16.1* [Bati et al. 2023], which accounts for radiative heat transfer between surfaces. We limit conductive and disable convective heat transfer to disambiguate between the three modes of heat transfer. We perform the simulation at steady state. In each thermal scene, we pick one object as the main heat source and vary its surface temperature (in the form of Dirichlet boundary condition) to create three different heating scenarios, where the main heat source temperature increases from 300K to 400K in fixed increments. These variations allow us to observe how surface temperature affects emitted radiation and its spatial distribution. In the following, we will give more specific description of each scene.

(1) WarmBunny: This scene uses the *Stanford Bunny* geometry with a constant Dirichlet surface boundary condition (temperature = 350K). The ambient temperature is set to 300K. The table surface follows a Robin boundary condition (emissivity = 0.5, specularity = 0.1).
(2) Radiator: This scene builds on the WarmBunny setup. The bunny is assigned a Robin boundary condition (emissivity = 0.5, specularity = 0.1) and is initialized at ambient temperature. The emissivity of the bunny surface is chosen to produce more distinct reflections. A rectangular heat source (radiator) is placed above the bunny and set to a Dirichlet boundary condition (temperature = 350K).

(3) Torus: This configuration is similar to the Radiator scene, except that the heat source is changed to a torus surrounding the bunny instead of a rectangular radiator. This variation introduces complex self-occlusion and shadowing effects.
(4) Human: This scene is a variation of the Radiator in an indoor architectural environment setting. The radiator is suspended above the human and desk. The human (emissivity = 0.9, specularity = 0.01) and desk (emissivity = 0.5, specularity = 0.0) are assigned Robin boundary conditions.

Real-world scene. We capture 350 hemispherical view points for a real-world scene comprising a heat source (ceramic teapot) and objects (two polymer 3D printed parts) placed on a wooden tabletop. We repeat the image capture process for three different heating conditions, resulting in three distinct temperature radiance fields. We achieve this by gradually increasing the input power to the teapot which radiates heat to the surrounding objects. We allow steady-state conditions to be achieved (>2 hours after the increase in input power) to minimize temperature variations in the scene before capturing the images.

The scene is captured by a FLIR T540 thermal camera (uncooled micro bolometer, spectral range 7.5-14 μm, 464x348 pixel resolution, 14" focal length, automatic laser-assisted focus, accuracy 0.2°C). From the thermal camera, raw temperature data is extracted to reconstruct thermal images. RGB images are also taken with the thermal images with the same image resolution and lens.

The novel view synthesis and inverse rendering results rely on the accuracy of the camera registration. We used *colmap* [Schonberger and Frahm 2016] to estimate camera parameters from the visible-spectrum images. Although studies demonstrating novel view synthesis of a thermal scene could use thermal images to estimate camera parameters [Lin et al. 2024; Lu et al. 2024], we use RGB images for better accuracy and consistency across heating scenarios. More specifically, we have three heating scenarios for each scene, and therefore visually inconsistent thermal images. The inconsistency leads to challenges if we estimate the exact same camera parameters using all the thermal images. On the contrary, the visible-spectrum images contain little visible changes across heating scenarios, and lead to better camera registration results. Additionally, with roughly 3 times the number of images, the estimation of sparse point cloud generated and camera parameters are improved.

5 Results

Implementation details. We implement our algorithm with PyTorch and `gsplat` [Ye et al. 2025]. We use `gsplat` for their 2DGS rasterization and the 3DGS-MCMC [Kheradmand et al. 2025] densification algorithm. We cap the number of primitives to 200,000 for all scenes. All of our experiments are conducted on an NVIDIA RTX 6000Ada.

Timings. The training time of our method depends on the number of primitives, iterations, and the batch size for evaluating the radiosity loss. For all of our experiment, we train with 200k primitives, 300k iterations, and use batch size 128, and the training takes around 14 hours. The rendering of a scene from a novel view runs at 192.47 FPS. Given a novel heating condition, reheating a scene by computing reflection with hemicubes takes 140.12 seconds for a scene of 200k primitives.

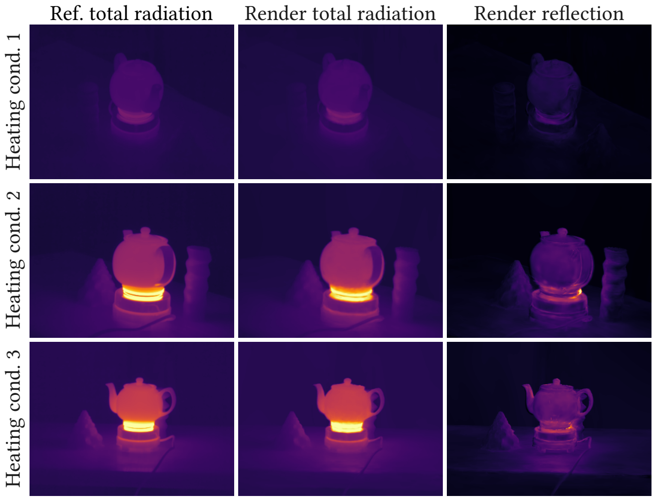


Fig. 4. **Training view reflection**: for each heating condition in the Teapot scene, we show the reference total radiation image; the rendering of total radiation; and the rendering of reflection, all after gamma mapping.

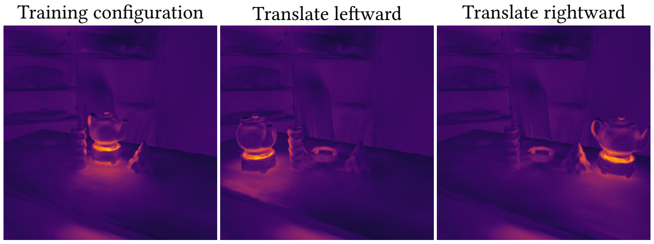


Fig. 5. **Heat source re-positioned**: after training, we move the teapot to new positions and predict the reflections.

5.1 Re-heating

After optimizing each scene using our algorithm, our method plausibly factors the reflection out from the total radiation, as shown in Figure 4 (real capture scene) and Figure 6 (synthetic scenes). The quantitative metrics evaluated on the training images are reported in Table 1. Samples of visualization of the predicted material parameters are provided in Figure 9.

We proceed to perform edits to the heat source objects in the scene and observe the reflection change. In this section, we showcase two types of edits: 1) repositioning an object, and 2) change the heat source's surface temperature. To test our method's effectiveness in this regard, we applied it to four scenes: three synthetic and one real capture. It is important to note that after editing, a scene might no longer be in thermal steady state. Since our model does not account for transient changes or heat transfer modes other than radiation, object temperatures that would typically change due to, for example, conduction, will remain unaltered in the edited scenes. Nonetheless, our experiments demonstrate plausible reflections using our method.

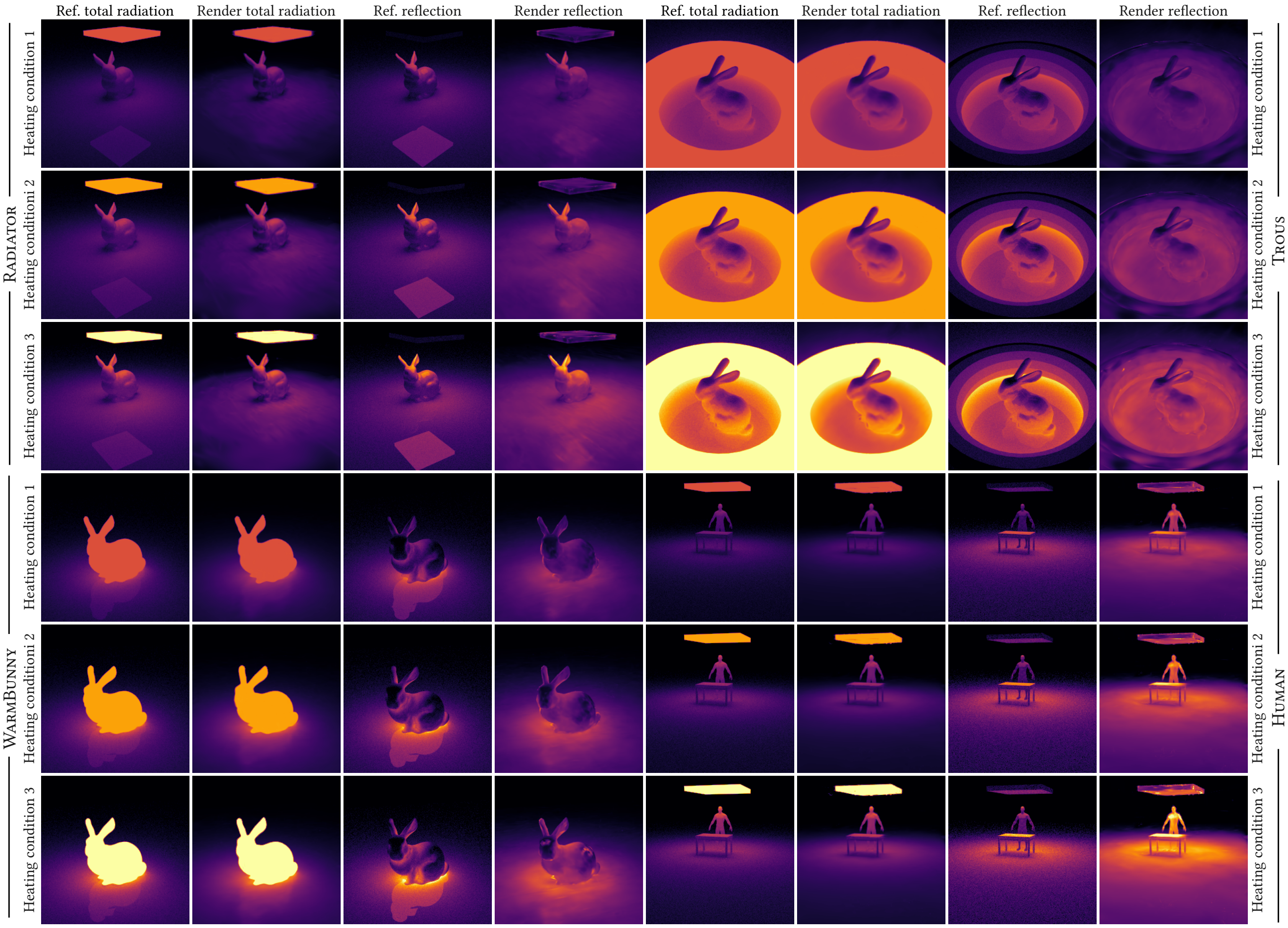


Fig. 6. **Training view reflection:** for each heating condition in each synthetic scene, we show the reference total radiation image; the rendering of total radiation; the reference reflection; the rendering of reflection, all after gamma mapping.

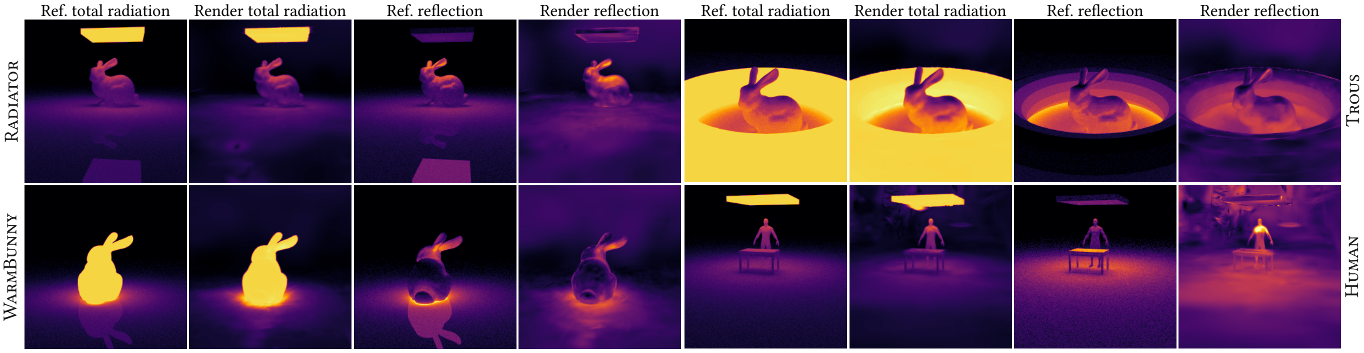


Fig. 7. **Temperature change**: after optimizing for each scene, we change the surface temperature of the main heat source object in each scene to a new value that is not in the training data, and predict the reflection and total radiation. We show the reference and rendered image of reflection, and reference and rendered image of total radiation, all after gamma mapping.

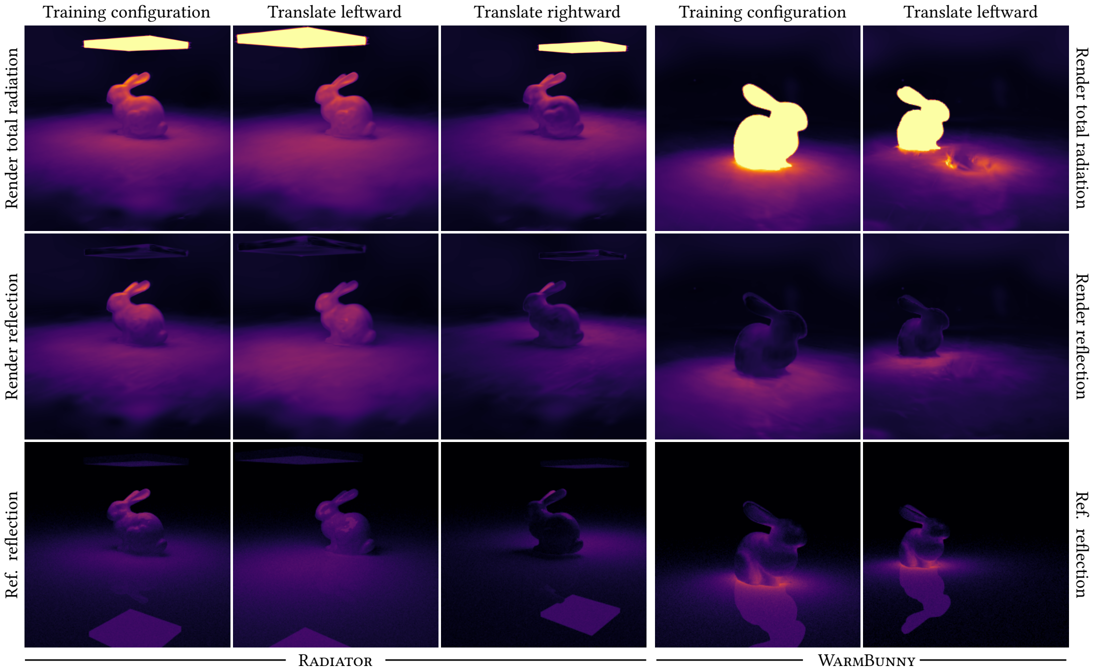


Fig. 8. **Heat source re-positioned**: we move the main heat source object to new positions, and predict the reflection. In WarmBunny, after re-positioning, the original position of the bunny still emits a noticeable amount of radiation. This is mainly because the scene after edit is not in thermal steady state and conduction is baked into the model.

The challenging yet significant task of integrating additional heat transfer modes will be a focus of future work.

Temperature change. We test our model's performance of predicting radiation condition in *novel heating condition* that is not part of the training dataset. We select all the Gaussian primitives that belong to an object in each scene using object masks obtained from *segment anything* [Kirillov et al. 2023], and then change the temperatures of these primitives to a predefined value. Under this new heating condition, we compute the reflection which we further compare with the reference images we generated by prescribing the same temperature change in Stardis. The comparison between our reflection, reference image, and our rendering are shown in Figure 7. Our reflection calculation is based on the positions and orientations of the Gaussian primitives, which do not change in the thermal reconstruction phase of our training pipeline. As a result, a suboptimal spatial distribution of primitives might limit the accuracy of the predicted reflection. For example, in the Human scene, many primitives float in the scene, and hamper the model's prediction of reflection when the radiator is set to a new temperature. The missing sharp reflection in the Bunny scene is also affected by an undesired distribution of the primitives: the primitives near the periphery are large, leading to difficulties in expressing sharp features.

Re-positioned heat sources. We also demonstrate our method's capability of computing reflections after we reposition heat source objects on two synthetic scenes and the real capture scene. In Figure 8, we move the heater around the bunny, and show the distribution of reflection and total radiation given the new heater position. Similarly, we test the model with the WarmBunny scene, and we can observe the reflection moving consistently while the bunny is being re-positioned. In the rendering of the total radiation of the re-positioned scene, there is still a noticeable amount of radiation emitting from the original position. As we discussed previously, this is because the scene after editing is not in thermal steady state. We also perform a similar experiment on the real capture scene, Teapot, as can be seen in Figure 5. Plausible reflections are observed on the cylinder and the cone as we move the teapot to the new positions.

5.2 Comparison

We compare our method with Thermo-NeRF [Hassan et al. 2025], a technique for novel view synthesis that utilizes visible-spectrum and thermal images without modeling reflection separation. We evaluated the view synthesis performance of both methods using our Teapot scene, quantitatively assessing rendering quality using training views and thermal conditions. The metrics presented in Table 2 clearly indicate our method's superior performance in reproducing the training images.

To the best of our knowledge, the closest work to analyze the reflection and self-emission of radiance fields is ESR-NeRF [Jeong et al. 2024]. Their method is designed for visible spectrum data, instead of infrared scenes, and requires a dataset of two subsets: emissive source on, and completely off. Our datasets usually do not include images with emissive sources completely off.

Table 1. The quantitative metrics of of the renderings of total radiation, and the renderings of reflection for each scene in our dataset. Reflection MAE (Mean Average Error) of Teapot is not available as no reflection references in real capture scenes. Average values are computed over all scenes with available metrics.

Metric	Radiator	Torus	WarmBunny	Human	Teapot	Average
Total radiation PSNR	35.162	37.381	35.678	31.852	38.496	35.713
Total radiation SSIM	0.915	0.949	0.867	0.836	0.982	0.909
Total radiation LPIPS	0.020	0.057	0.155	0.221	0.019	0.094
Reflection MAE ($\times 10^{-3}$)	5.189	18.129	5.123	9.530	No Ref.	9.492

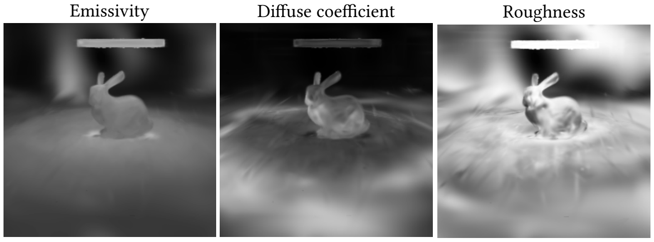


Fig. 9. **Material parameters**: predicted emissivity ε, diffuse coefficient d, and roughness s of Radiator scene.

Table 2. We compare training view synthesis results with Thermo-NeRF [Hassan et al. 2025] on our Teapot scene; ↑ means higher is better, and ↓ means lower is better.

Method	**PSNR** ↑	**SSIM** ↑	**LPIPS** ↓
Ours	38.496	0.982	0.019
Thermo-NeRF	28.032	0.942	0.085

6 Conclusions

Summary. In this work, we present an inverse radiative transfer method to compute the energy exchange given thermal images, and a dataset that features paired RGB thermal images captured at different heating scenarios. We propose to use Gaussian primitives together with hemicubes, a rasterization-based technique, to compute the self-emission and reflection of the Gaussian primitives given thermal images. To the best of our knowledge, this is the first work that separates reflection from total radiation using Gaussian primitives. We demonstrate the effectiveness of our method by showing applications such as re-positioning and re-heating using our dataset.

Discussions on extension to visible-spectrum scenes. Our inverse thermal radiative transfer method is designed for and tested on thermal scenes. The core of the algorithm, the hemicube-represented reflection and spherical Gaussian cache, is readily adaptable to RGB scenes. The adaptation requires minor modifications to the material models, primarily replacing emissivity with albedo. Another challenge might arise in self-emission modeling: unlike infrared scenes where nearly all objects self-emit, visible-spectrum scenes typically receive light from the environment and a few light sources.

Limitations and future work. We only consider steady-state radiative transport in the scope of this work. There are many other important aspects of thermal simulation in real life that are out of the scope of this work. And we believe that it would open many promising directions for future improvements to our methods. For example, incorporating transient simulation and other modes of heat transport, i.e., conduction and convection, into the system could enable steady-state simulation *after* the scene being edited. The separation of the geometry and thermal reconstruction in our training strategy limits the model's accuracy when the primitives are not optimally representing the object, (e.g., floaters, noisy normals). Jointly optimizing the thermal physics and geometry could improve the robustness of our method and better represent the sharp reflections. We estimate the incoming radiance using hemicubes which need to rasterize five different faces separately. With recent advances in the rasterization of Gaussian primitives supporting distorted camera [Wu et al. 2025], the computation of incoming radiance could be made more efficient. Another direction of future work is to scale up the method to larger scenes such as indoor capture or outdoor building capture, which might inspire new approaches to interpret energy flows in urban-scale thermal 3D scenes to drive design decisions for architectural renovations.

Acknowledgments

Bharath Seshadri's position is funded by the Albert Lück Stiftung via the ETH Zürich Foundation for the project "Extended Reality for Inspection, Assembly, Operations for net-zero carbon infrastructure" The authors thank the anonymous reviewers for their valuable feedback; Dominik Stoll for his help with the figures; Jeremy Chew for many insightful discussions; Stanford University Computer Graphics Laboratory for the *Stanford Bunny* model.

References

Alakel Mohammad Abazid, Aref Lakhal, and Alfred K. Louis. 2016. Inverse design of anti-reflection coatings using the nonlinear approximate inverse. *Inverse Problems in Science and Engineering* (2016), 917–935. https://doi.org/10.1080/17415977.2015.1077446

Greg Acosta, Andrew Reicks, Miguel Moreno, Alireza Borjali, Craig Zuhlke, and Mohammad Ghashami. 2022. Emissivity prediction of functionalized surfaces using artificial intelligence. *Journal of Quantitative Spectroscopy and Radiative Transfer* (2022), 108325. https://doi.org/10.1016/j.jqsrt.2022.108325

Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. 2022. Mip-NeRF 360: Unbounded Anti-Aliased Neural Radiance Fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 5460–5469. https://doi.org/10.1109/CVPR52688.2022.00539

MÃ©gane Bati, StÃ©phane Blanco, Christophe Coustet, Vincent Eymet, Vincent Forest, Richard Fournier, Jacques Gautrais, Nicolas Mellado, Mathias Paulin, and Benjamin Piaud. 2023. Coupling Conduction, Convection and Radiative Transfer in a Single Path-Space: Application to Infrared Rendering. *ACM Transactions on Graphics* (2023), 1–20. https://doi.org/10.1145/3592121

Yuxiang Cai, Jiaxiong Qiu, Zhong Li, and Bo Ren. 2024. NeuralTO: Neural Reconstruction and View Synthesis of Translucent Objects. *ACM Transactions on Graphics* (2024), 50:1–50:14. https://doi.org/10.1145/3658186

Qian Chen, Shihao Shu, and Xiangzhi Bai. 2025. Thermal3D-GS: Physics-Induced 3D Gaussians forÂ Thermal Infrared Novel-View Synthesis. In *Computer Vision â€“ ECCV 2024*. Springer Nature Switzerland, 253–269. https://doi.org/10.1007/978-3-031-73383-3_15

Michael F. Cohen and Donald P. Greenberg. 1985. The hemi-cube: a radiosity solution for complex environments. *ACM SIGGRAPH Computer Graphics* (1985), 31–40. https://doi.org/10.1145/325165.325171

Jorge Condor, Sebastien Speierer, Lukas Bode, Aljaz Bozic, Simon Green, Piotr Didyk, and Adrian Jarabo. 2025. Don't Splat your Gaussians: Volumetric Ray-Traced Primitives for Modeling and Rendering Scattering and Emissive Media. *ACM Transactions on Graphics* (2025), 10:1–10:17. https://doi.org/10.1145/3711853

R. L. Cook and K. E. Torrance. 1982. A Reflectance Model for Computer Graphics. *ACM Transactions on Graphics* (1982), 7–24. https://doi.org/10.1145/357290.357293

Carsten Dachsbacher, Marc Stamminger, George Drettakis, and FrÃ©do Durand. 2007. Implicit visibility and antiradiance for interactive global illumination. *ACM Transactions on Graphics* (2007), 61–es. https://doi.org/10.1145/1276377.1276453

Ranjan Das, Subhash C. Mishra, M. Ajith, and R. Uppaluri. 2008. An inverse analysis of a transient 2-D conductionâ€“radiation problem using the lattice Boltzmann method and the finite volume method coupled with the genetic algorithm. *Journal of Quantitative Spectroscopy and Radiative Transfer* (2008), 2060–2077. https://doi.org/10.1016/j.jqsrt.2008.01.011

K.J. Daun, J.R. Howell, and D.P. Morton. 2003. Design of radiant enclosures using inverse and non-linear programming techniques. *Inverse Problems in Engineering* (2003), 541–560. https://doi.org/10.1080/1068276031000086796

Jan-Niklas Dihlmann, Arjun Majumdar, Andreas Engelhardt, Raphael Braun, and Hendrik P.A. Lensch. 2025. Subsurface scattering for 3D Gaussian splatting. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*. Curran Associates Inc., 121765–121789. https://doi.org/10.5555/3737916.3741786

E. D. Eastman. 1936. The Third Law of Thermodynamics. *Chemical Reviews* (1936), 257–275. https://doi.org/10.1021/cr60060a002

Hakan ErtÃ¼rk, Kyle Daun, Francis H. R. FranÃ§a, Shima Hajimirza, and John R. Howell. 2023. Inverse Methods in Thermal Radiation Analysis and Experiment. *ASME Journal of Heat and Mass Transfer* (2023). https://doi.org/10.1115/1.4056371

C. Freude, D. Hahn, F. Rist, L. Lipp, and M. Wimmer. 2023. Precomputed Radiative Heat Transport for Efficient Thermal Simulation. *Computer Graphics Forum* (2023), e14957. https://doi.org/10.1111/cgf.14957

C. Freude, L. Lipp, M. Zezulka, F. Rist, M. Wimmer, and D. Hahn. 2025. Inverse Simulation of Radiative Thermal Transport. *Computer Graphics Forum* (2025), e70048. https://doi.org/10.1111/cgf.70048

Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. 2022. Plenoxels: Radiance Fields without Neural Networks. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5491–5500. https://doi.org/10.1109/CVPR52688.2022.00542

Jian Gao, Chun Gu, Youtian Lin, Zhihao Li, Hao Zhu, Xun Cao, Li Zhang, and Yao Yao. 2024. Relightable 3D Gaussians: Realistic Point Cloud Relighting withÂ BRDF Decomposition andÂ Ray Tracing. In *Computer Vision â€“ ECCV 2024: 18th European Conference*. Springer-Verlag, 73–89. https://doi.org/10.1007/978-3-031-72995-9_5

Wenhang Ge, Tao Hu, Haoyu Zhao, Shu Liu, and Ying-Cong Chen. 2023. Ref-NeuS: Ambiguity-Reduced Neural Implicit Surface Learning for Multi-View Reconstruction with Reflection. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 4228–4237. https://doi.org/10.1109/ICCV51070.2023.00392

Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. 1984. Modeling the interaction of light between diffuse surfaces. *ACM SIGGRAPH Computer Graphics* (1984), 213–222. https://doi.org/10.1145/964965.808601

Saeed Hadadan, Shuhong Chen, and Matthias Zwicker. 2021. Neural radiosity. *ACM Transactions on Graphics* (2021), 236:1–236:11. https://doi.org/10.1145/3478513.3480569

Saeed Hadadan, Geng Lin, Jan NovÃ¡k, Fabrice Rousselle, and Matthias Zwicker. 2023. Inverse Global Illumination using a Neural Radiometric Prior. In *ACM SIGGRAPH 2023 Conference Proceedings*. ACM, 1–11. https://doi.org/10.1145/3588432.3591553

Mariam Hassan, Florent Forest, Olga Fink, and Malcolm Mielle. 2025. ThermoNeRF: A multimodal Neural Radiance Field for joint RGB-thermal novel view synthesis of building facades. *Adv. Eng. Inform.* (2025). https://doi.org/10.1016/j.aei.2025.103345

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2024. 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. In *ACM SIGGRAPH 2024 Conference Papers*. ACM, 1–11. https://doi.org/10.1145/3641519.3657428

David S. Immel, Michael F. Cohen, and Donald P. Greenberg. 1986. A radiosity method for non-diffuse environments. *ACM SIGGRAPH Computer Graphics* (1986), 133–142. https://doi.org/10.1145/15886.15901

Jinseo Jeong, Junseo Koo, Qimeng Zhang, and Gunhee Kim. 2024. ESR-NeRF: Emissive Source Reconstruction Using LDR Multi-View Images. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4598–4609. https://doi.org/10.1109/CVPR52733.2024.00440

Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. 2024. GaussianShader: 3D Gaussian Splatting with Shading Functions for Reflective Surfaces. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5322–5332. https://doi.org/10.1109/CVPR52733.2024.00509

Haian Jin, Isabella Liu, Peijia Xu, Xiaoshuai Zhang, Songfang Han, Sai Bi, Xiaowei Zhou, Zexiang Xu, and Hao Su. 2023. TensoIR: Tensorial Inverse Rendering. (2023). https://doi.org/10.1109/CVPR52729.2023.00024

James T. Kajiya. 1986. The rendering equation. *ACM SIGGRAPH Computer Graphics* (1986), 143–150. https://doi.org/10.1145/15886.15902

Byungkwon Kang, Insung Ihm, and Chandrajit Bajaj. 2005. Extending the photon mapping method for realistic rendering of hot gaseous fluids. *Computer Animation and Virtual Worlds* (2005), 353–363. https://doi.org/10.1002/cav.94

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkuehler, and George Drettakis. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics* (2023), 139:1–139:14. https://doi.org/10.1145/3592433

Shakiba Kheradmand, Daniel Rebain, Gopal Sharma, Weiwei Sun, Yang-Che Tseng, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. 2025. 3D Gaussian splatting as Markov Chain Monte Carlo. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*. Curran Associates Inc., Red Hook, NY, USA, 80965–80986.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr DollÃ¡r, and Ross Girshick. 2023. Segment Anything. https://arxiv.org/abs/2304.02643

Stephan Kramer, Ralf Gritzki, Alf Perschk, Markus RÃ¶sler, and Clemens Felsmann. 2015. Numerical simulation of radiative heat transfer in indoor environments on programmable graphics hardware. *International Journal of Thermal Sciences* (2015), 345–354. https://doi.org/10.1016/j.ijthermalsci.2015.02.008

Myriam Lazard, StÃ©phane AndrÃ©, Denis Maillet, and Alain Degiovanni. 2000. Radiative and conductive heat transfer: a coupled model for parameter estimation. *High Temperatures-High Pressures* (2000), 9–17. https://doi.org/10.1068/htwu311

Zhihao Liang, Qi Zhang, Ying Feng, Ying Shan, and Kui Jia. 2024. GS-IR: 3D Gaussian Splatting for Inverse Rendering. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 21644–21653. https://doi.org/10.1109/CVPR52733.2024.02045

Yvette Y. Lin, Xin-Yi Pan, Sara Fridovich-Keil, and Gordon Wetzstein. 2024. ThermalNeRF: Thermal Radiance Fields. In *2024 IEEE International Conference on Computational Photography (ICCP)*. 1–12. https://doi.org/10.1109/ICCP61108.2024.10644336

Liu Linhua, Tan Heping, and Yu Qizheng. 1999. Inverse radiation problem of temperature field in three-dimensional rectangular furnaces. *International Communications in Heat and Mass Transfer* (1999), 239–248. https://doi.org/10.1016/S0735-1933(99)00010-X

D. Lischinski, F. Tampieri, and D.P. Greenberg. 1992. Discontinuity meshing for accurate radiosity. *IEEE Computer Graphics and Applications* (1992), 25–39. https://doi.org/10.1109/38.163622

D. Liu, J. H. Yan, F. Wang, Q. X. Huang, Y. Chi, and K. F. Cen. 2010. Inverse radiation analysis of simultaneous estimation of temperature field and radiative properties in a two-dimensional participating medium. *International Journal of Heat and Mass Transfer* (2010), 4474–4481. https://doi.org/10.1016/j.ijheatmasstransfer.2010.06.046

Yuan Liu, Peng Wang, Cheng Lin, Xiaoxiao Long, Jiepeng Wang, Lingjie Liu, Taku Komura, and Wenping Wang. 2023. NeRO: Neural Geometry and BRDF Reconstruction of Reflective Objects from Multiview Images. *ACM Transactions on Graphics* (2023), 1–22. https://doi.org/10.1145/3592134

Rongfeng Lu, Hangyu Chen, Zunjie Zhu, Yuhang Qin, Ming Lu, Le Zhang, Chenggang Yan, and Anke Xue. 2024. ThermalGaussian: Thermal 3D Gaussian Splatting. In *International Conference on Representation Learning 2025 (ICLR)*. https://proceedings.iclr.cc/paper_files/paper/2025/hash/03bdba50e3741ac5e3eaa0e55423587e-Abstract-Conference.html

Sean Mann, Eric Fadel, Samuel S. Schoenholz, Ekin D. Cubuk, Steven G. Johnson, and Giuseppe Romano. 2022. *∂PV*: An end-to-end differentiable solar-cell simulator. *Computer Physics Communications* (2022), 108232. https://doi.org/10.1016/j.cpc.2021.108232

Sandip Mazumder and Alfred Kersch. 2000. A Fast Monte Carlo Scheme for Thermal Radiation in Semiconductor Processing Applications. *Numerical Heat Transfer, Part B: Fundamentals* (2000), 185–199. https://doi.org/10.1080/104077900275486

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *Computer Vision â€“ ECCV 2020*. https://doi.org/10.1007/978-3-030-58452-8_24

Bailey Miller, Rohan Sawhney, Keenan Crane, and Ioannis Gkioulekas. 2024. Differential Walk on Spheres. *ACM Transactions on Graphics* (2024), 174:1–174:18. https://doi.org/10.1145/3687913

Nicolas Moenne-Loccoz, Ashkan Mirzaei, Or Perel, Riccardo de Lutio, Janick Martinez Esturo, Gavriel State, Sanja Fidler, Nicholas Sharp, and Zan Gojcic. 2024. 3D Gaussian Ray Tracing: Fast Tracing of Particle Scenes. *ACM Transactions on Graphics* (2024), 232:1–232:19. https://doi.org/10.1145/3687934

Thomas MÃ¼ller, Fabrice Rousselle, Jan NovÃ¡k, and Alexander Keller. 2021. Real-time neural radiance caching for path tracing. *ACM Transactions on Graphics* (2021), 36:1–36:16. https://doi.org/10.1145/3450626.3459812

Merlin Nimier-David, SÃ©bastien Speierer, BenoÃ®t Ruiz, and Wenzel Jakob. 2020. Radiative backpropagation: an adjoint method for lightning-fast differentiable rendering. *ACM Transactions on Graphics* (2020), 146:146:1–146:146:15. https://doi.org/10.1145/3386569.3392406

Jose Pedro Aguerre and Eduardo FernÃ¡ndez. 2021. ThRend: a ray tracing module for infrared rendering of urban scenes. IBPSA, 1588–1595. https://doi.org/10.26868/25222708.2021.30435

L. Penazzi, S. Blanco, C. Caliot, C. Coustet, M. El Hafi, R. Fournier, J. Gautrais, A. Golijanek-Jędrzejczyk, and M. Sans. 2024. Path integrals formulations leading to propagator evaluation for coupled linear physics in large geometric models. *Computer Physics Communications* (2024), 108911. https://doi.org/10.1016/j.cpc.2023.108911

Matt Pharr, Wenzel Jakob, and Greg Humphreys. 2023. *Physically Based Rendering, fourth edition: From Theory to Implementation.* The MIT Press, Cambridge London.

Thomas Pierre, Jean-Claude Krapez, Helcio R. B. Orlande, Christophe Rodiet, Dylan Le Maux, MickaÃ«l Courtois, Philippe Le Masson, and Bernard Lamien. 2022. Simultaneous estimation of temperature and emissivity of metals around their melting points by deterministic and Bayesian techniques. *International Journal of Heat and Mass Transfer* (2022), 122077. https://doi.org/10.1016/j.ijheatmasstransfer.2021.122077

Max Planck. 1906. *The theory of heat radiation.* Henry Holt and Company.

Kripasindhu Sarkar, Marcel C. BÃ¼hler, Gengyan Li, Daoye Wang, Delio Vicini, JÃ©rÃ©my Riviere, Yinda Zhang, Sergio Orts-Escolano, Paulo Gotardo, Thabo Beeler, and Abhimitra Meka. 2023. LitNeRF: Intrinsic Radiance Decomposition for High-Quality View Synthesis and Relighting of Faces. In *SIGGRAPH Asia 2023 Conference Papers.* ACM, 1–11. https://doi.org/10.1145/3610548.3618210

Rohan Sawhney, Bailey Miller, Ioannis Gkioulekas, and Keenan Crane. 2023. Walk on Stars: A Grid-Free Monte Carlo Method for PDEs with Neumann Boundary Conditions. *ACM Transactions on Graphics* (2023), 80:1–80:20. https://doi.org/10.1145/3592398

Rohan Sawhney, Dario Seyb, Wojciech Jarosz, and Keenan Crane. 2022. Grid-free Monte Carlo for PDEs with spatially varying coefficients. *ACM Transactions on Graphics* (2022), 53:1–53:17. https://doi.org/10.1145/3528223.3530134

Christophe Schlick. 1994. An Inexpensive BRDF Model for Physically-based Rendering. *Computer Graphics Forum* (1994), 233–246. https://doi.org/10.1111/1467-8659.1330233

Johannes L. Schonberger and Jan-Michael Frahm. 2016. Structure-from-Motion Revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 4104–4113. https://doi.org/10.1109/CVPR.2016.445

FranÃ§is X. Sillion, James R. Arvo, Stephen H. Westin, and Donald P. Greenberg. 1991. A global illumination solution for general reflectance distributions. In *Proceedings of the 18th annual conference on Computer graphics and interactive techniques.* ACM, 187–196. https://doi.org/10.1145/122718.122739

Pratul P. Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T. Barron. 2021. NeRV: Neural Reflectance and Visibility Fields for Relighting and View Synthesis. IEEE Computer Society, 7491–7500. https://doi.org/10.1109/CVPR46437.2021.00741

Delio Vicini, SÃ©bastien Speierer, and Wenzel Jakob. 2022. Differentiable signed distance function rendering. *ACM Transactions on Graphics* (2022), 125:1–125:18. https://doi.org/10.1145/3528223.3530139

Donald V. Walters and Richard O. Buckius. 1994. Monte Carlo methods for radiative heat transfer in scattering media. *Annual Review of Heat Transfer* (1994). https://doi.org/10.1615/AnnualRevHeatTransfer.v5.50

Dongqing Wang, Tong Zhang, and Sabine SÃ¼sstrunk. 2023. NEMTO: Neural Environment Matting for Novel View and Relighting Synthesis of Transparent Objects. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV).* IEEE, 317–327. https://doi.org/10.1109/ICCV51070.2023.00036

Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. 2021. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. In *Advances in Neural Information Processing Systems.* Curran Associates, Inc., 27171–27183. https://proceedings.neurips.cc//paper/2021/hash/e41e164f7485ec4a28741a2d0ea41c74-Abstract.html

Qi Wu, Janick Martinez Esturo, Ashkan Mirzaei, Nicolas Moenne-Loccoz, and Zan Gojcic. 2025. 3DGUT: Enabling Distorted Cameras and Secondary Rays in Gaussian Splatting. *Conference on Computer Vision and Pattern Recognition (CVPR)* (2025).

Jiacong Xu, Mingqian Liao, Ram Prabhakar Kathirvel, and Vishal M. Patel. 2025. Leveraging Thermal Modality toÂ Enhance Reconstruction inÂ Low-Light Conditions. In *Computer Vision â€“ ECCV 2024.* Springer Nature Switzerland, 321–339. https://doi.org/10.1007/978-3-031-72913-3_18

Yingyan Xu, Gaspard Zoss, Prashanth Chandran, Markus Gross, Derek Bradley, and Paulo Gotardo. 2023. ReNeRF: Relightable Neural Radiance Fields with Nearfield Lighting. (2023). https://doi.org/10.1109/ICCV51070.2023.02064

Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. 2021. Volume rendering of neural implicit surfaces. In *Proceedings of the 35th International Conference on Neural Information Processing Systems.* Curran Associates Inc., Red Hook, NY, USA, 4805–4815.

Vickie Ye, Ruilong Li, Justin Kerr, Matias Turkulainen, Brent Yi, Zhuoyang Pan, Otto Seiskari, Jianbo Ye, Jeffrey Hu, Matthew Tancik, and Angjoo Kanazawa. 2025. gsplat: An Open-Source Library for Gaussian Splatting. *Journal of Machine Learning Research* (2025), 1–17. http://jmlr.org/papers/v26/24-1476.html

Ekrem Fatih Yilmazer, Delio Vicini, and Wenzel Jakob. 2024. Solving Inverse PDE Problems using Grid-Free Monte Carlo Estimators. *ACM Transactions on Graphics* (2024), 1–18. https://doi.org/10.1145/3687990

Zehao Yu, Torsten Sattler, and Andreas Geiger. 2024a. Gaussian Opacity Fields: Efficient Adaptive Surface Reconstruction in Unbounded Scenes. *ACM Transactions on Graphics* (2024), 271:1–271:13. https://doi.org/10.1145/3687937

Zihan Yu, Lifan Wu, Zhiqian Zhou, and Shuang Zhao. 2024b. A Differential Monte Carlo Solver For the Poisson Equation. In *ACM SIGGRAPH 2024 Conference Papers.* ACM, 1–10. https://doi.org/10.1145/3641519.3657460

Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. 2021a. PhySG: Inverse Rendering with Spherical Gaussians for Physics-based Material Editing and Relighting. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 5449–5458. https://doi.org/10.1109/CVPR46437.2021.00541

Xiuming Zhang, Pratul P. Srinivasan, Boyang Deng, Paul Debevec, William T. Freeman, and Jonathan T. Barron. 2021b. NeRFactor: Neural Factorization of Shape and Reflectance Under an Unknown Illumination. *ACM Transactions on Graphics* (2021), 1–18. https://doi.org/10.1145/3478513.3480496

Youjia Zhang, Teng Xu, Junqing Yu, Yuteng Ye, Yanqing Jing, Junle Wang, Jingyi Yu, and Wei Yang. 2023. NeMF: Inverse Volume Rendering with Neural Microflake Field. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV).* IEEE, 22862–22872. https://doi.org/10.1109/ICCV51070.2023.02095

Liu Zhenyuan, Yu Guo, Xinyuan Li, Bernd Bickel, and Ran Zhang. 2025. BiGS: Bidirectional Primitives for Relightable 3D Gaussian Splatting. IEEE Computer Society, 1022–1031. https://doi.org/10.1109/3DV66043.2025.00099