AGGREGATION IN NEURAL FIELDS VIA DIFFERENTIAL CONSTRAINTS

NTUMBA ELIE NSAMPI

Dissertation zur Erlangung des Grades des

Doktors der Ingenieurwissenschaften (Dr.-Ing.)

der Fakultät für Mathematik und Informatik
der Universität des Saarlandes

Saarbrücken, 2026

Date of Colloquium	16.07.2026
Dean of the Faculty	Prof. Dr. Jan Reineke
Chair of the Committee	Prof. Dr.-Ing. Thorsten Herfet
Reviewers	Dr. Thomas Leimkühler
	Prof. Dr. Hans-Peter Seidel
	Prof. Dr. Hendrik Lensch
Academic Assistant	Dr. Corentin Salaün

ABSTRACT

Aggregation operators, that is, operators defined through integration over continuous domains such as space, direction, or light paths, are central to visual computing. In practice, aggregation on continuous functions is typically carried out either through discrete approximations or through Monte Carlo sampling. Neural fields have emerged as flexible representations of continuous signals, but despite their appealing properties, it remains unclear how aggregation operators can be realized directly on neural fields without reverting to conventional approaches.

This thesis addresses this challenge through the observation that certain aggregation operators, although global in nature, can in some cases be reformulated as local differential constraints. This makes it possible to realize aggregation on neural fields without resorting to explicit numerical integration.

Building on this observation, this thesis contributes two methods that instantiate this principle in different settings. The first contribution is a method for continuous convolution on functions encoded as neural fields, based on convolution identities and the sparsity induced by repeated differentiation of piecewise polynomial kernels. The second contribution is a novel formulation for volumetric inverse rendering under full global illumination, in which both the optical properties of the medium and the transported light field are represented as neural fields, while global illumination is enforced through a differential formulation of the radiative transfer equation together with image-based supervision.

ZUSAMMENFASSUNG

Aggregationsoperatoren, also Operatoren, die durch Integration über kontinuierliche Bereiche wie Raum, Richtung oder Lichtwege definiert werden, sind zentral für die visuelle Datenverarbeitung. In der Praxis erfolgt die Aggregation kontinuierlicher Funktionen typischerweise entweder durch diskrete Approximationen oder durch Monte-Carlo-Simulationen. Neuronale Felder haben sich als flexible Repräsentationen kontinuierlicher Signale etabliert. Trotz ihrer attraktiven Eigenschaften ist jedoch unklar, wie Aggregationsoperatoren direkt auf neuronalen Feldern realisiert werden können, ohne auf konventionelle Ansätze zurückzugreifen.

Diese Arbeit befasst sich mit dieser Herausforderung, indem sie zeigt, dass bestimmte Aggregationsoperatoren, obwohl globaler Natur, in einigen Fällen als lokale Bedingungen auf Differentiale formuliert werden können. Dies ermöglicht die Realisierung von Aggregation auf neuronalen Feldern ohne explizite numerische Integration.

Aufbauend auf dieser Beobachtung präsentiert diese Arbeit zwei Methoden, die dieses Prinzip in unterschiedlichen Kontexten anwenden. Die erste Methode ist eine Methode zur kontinuierlichen Faltung von als neuronale Felder kodierten Funktionen. Sie basiert auf Faltungsidentitäten und der durch wiederholte Differentiation stückweise polynomialer Kerne induzierten Sparsität. Der zweite Beitrag ist eine neuartige Formulierung für volumetrisches inverses Rendering unter vollständiger globaler Beleuchtung, bei der sowohl die optischen Eigenschaften des Mediums als auch das transportierte Lichtfeld als neuronale Felder dargestellt werden, während die globale Beleuchtung durch eine differentielle Formulierung der Strahlungstransportgleichung zusammen mit bildbasierter Überwachung ermöglicht wird.

ACKNOWLEDGMENTS

A PhD is often described as a piece of research, but to me it has also been a long journey of becoming: a path marked by curiosity, uncertainty, persistence, and growth. No one walks such a path alone, and this work carries with it the presence of many people whose support, guidance, and generosity made it possible.

First and foremost, I owe my deepest gratitude to Thomas Leimkühler. Working closely with him has been one of the defining privileges of this journey. His guidance shaped not only the direction of this thesis, but also the way I learned to think, question, and push ideas further. In the many conversations, iterations, and moments of doubt that accompany research, his steady presence made the path clearer.

I am equally grateful to Tobias Ritschel, whose perspective repeatedly opened new ways of looking at problems. His suggestions arrived not merely as comments, but as invitations to think more deeply and more creatively. Many ideas in this thesis are stronger because of his insight.

My sincere thanks also go to Hans-Peter Seidel, whose leadership created the kind of environment in which research can truly flourish. Institutions matter, but what matters even more is the culture they cultivate: one of freedom, rigor, and possibility. I have been fortunate to work in a place where I was given both the opportunity and the trust to explore.

I am grateful to Adarsh Djeacoumar for his help with running experiments, building the data generation pipeline, and for the many technical discussions we shared along the way. His support was an important part of the day-to-day process behind this work.

Beyond academia, I owe more than words can easily express to my wife Bibi. Throughout this journey, she was my grounding force: the one who brought perspective when the work threatened to consume everything else, who restored calm in moments of chaos, and who reminded me, again and again, that there is a life larger than deadlines, experiments, and unanswered questions. This thesis may be written in my name, but it was lived with her love and patience beside me.

I am also deeply grateful to my parents. Long before I understood what a PhD was, they gave me something far more enduring: the belief that determination can carry a person farther than they first imagine. Whatever resilience I have brought to this journey has its roots in what they taught me through their strength, their faith, and their example.

Finally, I would like to thank my previous mentors, as well as everyone at MPI, who in ways both visible and invisible became part of this journey. A kind word, a passing conversation, shared frustrations, shared excitement, moments of encouragement, or simply the presence of others walking similar paths — all of these leave traces. Even when not directly acknowledged in these pages, they are part of the story behind them.

UNIVERSITÄT
DES
SAARLANDES

Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form in einem Verfahren zur Erlangung eines akademischen Grades vorgelegt.

Declaration of original authorship

I hereby declare that this dissertation is my own original work except where otherwise indicated. All data or concepts drawn directly or indirectly from other sources have been correctly acknowledged. This dissertation has not been submitted in its present or similar form to any other academic institution either in Germany or abroad for the award of any other degree.

Saarbrücken, ***2026***

gez. / signed

Nsampi Ntumba Elie

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 MOTIVATION

In visual computing, continuous functions provide a natural abstraction for describing signals and physical processes.

While functions are defined as abstract objects in mathematics, their use in computing requires choosing a finite representation. This is because functions over continuous domains cannot be represented or manipulated exactly in practice. Consequently, the choice of representation plays a central role in determining how such functions can be processed and analyzed.

Neural fields, also known as implicit neural representations, have emerged as a powerful way to represent continuous functions and have seen widespread adoption in visual computing (Xie et al., 2022). They compactly encode functions within the weights of a neural network and are differentiable with respect to both inputs and parameters, making them particularly well suited for optimization and learning-based pipelines. Owing to their flexibility, neural fields can represent a wide range of signals, including 2D images (Song et al., 2016; Stanley, 2007), 3D geometry (Park et al., 2019), radiance fields (Mildenhall et al., 2020), and time-varying signals such as videos and three-dimensional flow fields (Liu et al., 2025).

Beyond function representation, neural fields must also support meaningful operations on the functions they encode (Dupont et al., 2022a). As they are increasingly adopted as a modality-agnostic alternative to classical, modality-specific representations such as grids and meshes, it becomes essential to understand how standard operators can be defined and evaluated directly within this implicit neural setting.

A fundamental class of operations considered in this thesis is aggregation operators. In this context, aggregation refers to operators defined through integration over a continuous domain, whose evaluation depends on accumulating contributions across that domain. Such operators play a central role in fields such as computer vision, signal processing, and computer graphics. For example, in image processing, blurring can be expressed as a convolution between an image and a kernel, where each pixel in the blurred image is obtained by integrating weighted contributions from neighboring pixels, as illustrated in Fig. 1.1 (Zhang, 2022).

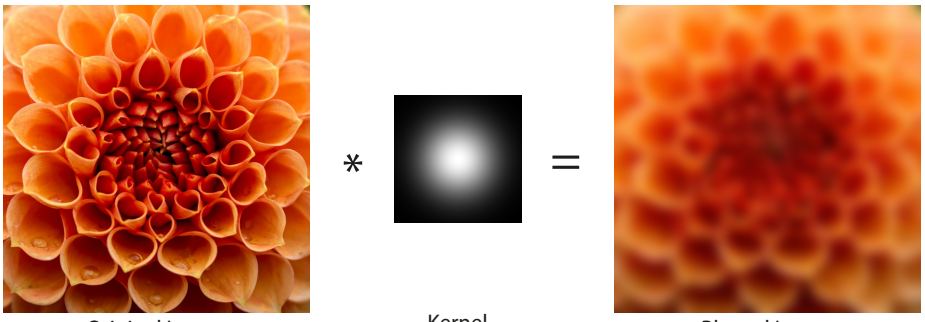

Figure 1.1: Convolution defines each output value as the aggregation of kernel-weighted contributions from a local neighborhood of the input. For a given output location, the kernel is centered at that location and combined with the corresponding input patch, yielding a weighted sum in the discrete setting or an integral in the continuous setting. Evaluating this local aggregation at all locations produces the output signal. Image by xiSerge, 2026.

In surface-based light transport, the radiance reflected toward the camera at a surface point is determined by integrating incoming illumination over the hemisphere of incident directions according to the rendering equation, as shown in Fig. 1.2a (Kajiya, 1986). In volume rendering, the value observed along a camera ray results from integrating emission, absorption, and scattering along the ray's path, as shown in Fig. 1.2b (Kajiya and Von Herzen, 1984). Depth of field and motion blur similarly arise from integration over aperture and time, respectively, as illustrated in Fig. 1.3a and Fig. 1.3b (Cook et al., 1984). In each of these cases, the observed quantity is defined through an integral operator that aggregates contributions over a continuous domain such as space, direction, time, or aperture, highlighting the central role aggregation plays in visual computing.

In practice, aggregation over continuous functions is typically carried out either by representing the function using a finite basis and applying the discrete analogue of the desired operator, or by approximating the operator using Monte Carlo sampling. While basis-based approaches can be effective in low-dimensional settings, they suffer from poor scalability due to the curse of dimensionality (Solomon, 2015). Moreover, within such finite representations, aggregation operators are typically evaluated by discretizing the domain, for example through quadrature rules or truncated basis expansions. Monte Carlo methods (Owen, 2013) alleviate some of these scalability issues but introduce high variance, often requiring a large number of samples or specialized importance sampling schemes to obtain accurate results. As a result, despite the continuous and implicit nature of neural fields, it remains unclear how aggregation

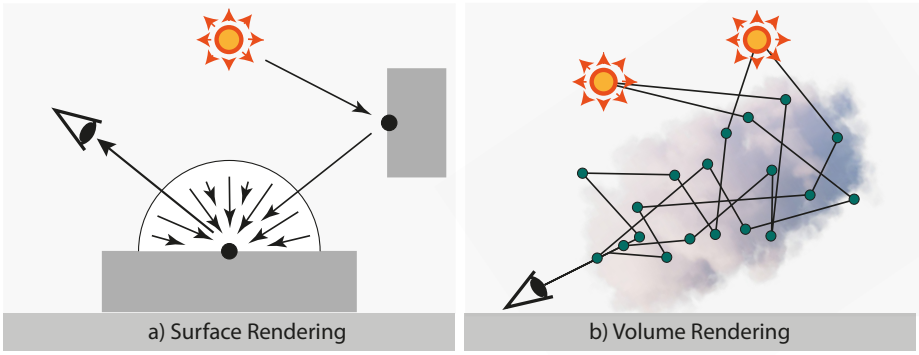

Figure 1.2: Rendering as aggregation. a) In surface rendering, outgoing radiance toward the camera is obtained by integrating incident illumination over the hemisphere of directions, weighted by the BRDF. b) In volumetric rendering, the pixel value corresponds to an integral over light transport through the medium and can be estimated by aggregating contributions from stochastic light paths.

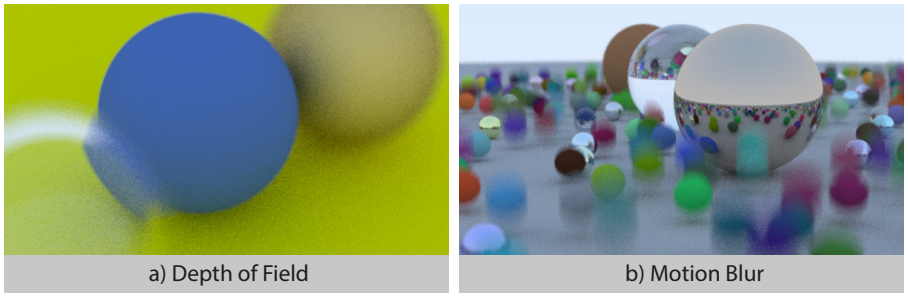

Figure 1.3: a) Depth of field integrates radiance over the camera aperture, aggregating rays from different lens positions. b) Motion blur integrates radiance over the exposure time, aggregating contributions from different temporal instants. Figure adapted from Peter Shirley, 2025.

operators can be performed directly on neural fields without reverting to discretization or stochastic sampling.

Motivated by the central role of aggregation in applications involving continuous functions, and by the challenges that arise when such operations are applied to neural field representations, this thesis proposes novel aggregation strategies for functions encoded as neural fields.

The key observation underlying the approaches proposed in this thesis is that certain aggregation operators, although global in nature, can be characterized by local differential relationships. In classical calculus, for example, the fundamental theorem of calculus shows that an accumulated quantity defined as $F(x) = \int_0^x f(x)\,dx$ is fully determined by the local relationship $\frac{dF(x)}{dx} = f(x)$ together with an appropriate boundary value. In this sense, an aggregation operator can be characterized through

a local differential constraint. More generally, in the settings considered in this thesis, aggregation operators defined through integrals can be reformulated as differential equations whose solutions encode the corresponding global behavior. This observation suggests that aggregation need not always be carried out through explicit numerical integration, but may instead be enforced through local differential relationships.

We show how the above principle can be realized in two distinct settings. First, we consider convolution, a fundamental aggregation operator in signal processing, and extend it to neural field representations, thereby enabling integral transforms to be performed directly on continuous functions. Second, we consider volumetric inverse rendering, where optical properties are recovered from observations, and use neural fields to represent both the medium and the transported light while enforcing physically grounded global light transport through a differential formulation without explicit global light transport simulation.

1.2 CONTRIBUTIONS

In Chapter 4, we introduce a method for performing continuous convolutions directly on neural fields and other continuous signals. By exploiting convolution identities and the structure of piecewise polynomial kernels under repeated differentiation, we train a repeated integral field that enables efficient large-scale convolutions across diverse data modalities. We contribute:

- A principled and versatile framework for performing convolutions in neural fields.
- Two novel enabling ingredients: an efficient method to train a repeated integral field, and the optimization of continuous kernels that are sparse after repeated differentiation.
- The evaluation of our framework on a range of modalities and kernels.

In Chapter 5, we introduce a method to recover the optical properties of participating media from multiview images. We propose a formulation that represents both the medium and the transported light as neural fields and enforces global light transport through a differential form of the radiative transfer equation. This enables the reconstruction of spatially varying optical properties from multi-view images and supports generative modeling of participating media. We contribute:

- A novel formulation for volumetric inverse rendering that accounts for full global illumination with only minimal reliance on transport-specific simulation.
- A joint estimation framework for a scene's optical properties and full light field, both represented as neural fields and constrained by first principles of global light transport.
- Experimental demonstrations of optical property reconstruction and the application of the formulation to generative modeling of physically meaningful participating media.

1.3 STRUCTURE

The remainder of this thesis is structured as follows:

- Chapter 2 introduces the required background knowledge, providing the technical foundation for the following chapters.
- Chapter 3 reviews previous work related to neural field representations, aggregation, and inverse rendering.
- Chapter 4 introduces Neural Field Convolutions, a method for performing general continuous convolutions on signals represented as neural fields.
- Chapter 5 proposes a neural inverse rendering method based on a neural formulation that reconciles physically complete light transport with general-purpose neural optimization.
- Chapter 6 concludes this thesis, summarizes our contributions, and discusses possible directions for future work.

1.4 PUBLICATIONS

The methods presented in this thesis are also publicly available in the following self-contained works:

- Nsampi, Ntumba Elie, Adarsh Djeacoumar, Hans-Peter Seidel, Tobias Ritschel, and Thomas Leimkühler (Dec. 2023). "Neural Field Convolutions by Repeated Differentiation." In: *ACM Transactions on Graphics (TOG)*. 42.6. ISSN: 0730-0301
- Nsampi, Ntumba Elie, Adarsh Djeacoumar, Hans-Peter Seidel, Tobias Ritschel, and Thomas Leimkühler (2026). "Volumetric Inverse

Rendering via Neural Radiative Transfer." In: *Computer Graphics Forum* 45. ISSN: 1467-8659. DOI: 10.1111/cgf.70541

Further contributions were made to the following works, which are not part of this thesis:

- Mujkanovic, Felix, Ntumba Elie Nsampi, Christian Theobalt, Hans-Peter Seidel, and Thomas Leimkühler (July 2024). "Neural Gaussian Scale-Space Fields." In: *ACM Transactions on Graphics (TOG).* 43.4. ISSN: 0730-0301
- Rubab, Fizza, Ntumba Elie Nsampi, Martin Balint, Felix Mujkanovic, Hans-Peter Seidel, Tobias Ritschel, and Thomas Leimkühler (2025). "Learning Neural Antiderivatives." In: *Vision, Modeling, and Visualization*. Ed. by Bernhard Egger and Tobias Günther. The Eurographics Association. ISBN: 978-3-03868-294-3

2 BACKGROUND

In this chapter, we introduce the background necessary for understanding the chapters that follow. We begin by introducing the concept of convolution and its practical computation. Next, we introduce the basics of volumetric light transport. Finally, we introduce neural fields. For a more in-depth treatment of these topics, we refer the reader to Gonzalez, 2009 for convolution, Chandrasekhar, 1960; Pharr et al., 2016 for volumetric light transport, and Essakine et al., 2025; Xie et al., 2022 for neural fields.

2.1 CONVOLUTION

We consider the convolution of arbitrary continuous signals $f \in R^{d_{\text{in}}} \to R^{d_{\text{out}}}$ with arbitrary continuous kernels $g \in R^d \to R$. Both inputs and outputs of f are low- to medium-dimensional. The signal can be any continuous function, including but not limited to a neural network. We assume the kernel has compact but potentially large support. The kernel does not necessarily extend across *all* input dimensions of f, i. e. $d \leq d_{\text{in}}$. To simplify our exposition, without loss of generality, we assume that the first d dimensions of f correspond to the filter dimensions of g. Further, we allow g to vary for different locations in the input space. An example of this setup is a space-time signal f encoding an RGB ($d_{\text{out}} = 3$) video with two spatial and a temporal dimension ($d_{\text{in}} = 3$), to be convolved with a kernel g that applies a foveated blur to each time slice of the video ($d = 2$). In the following derivations, we assume a spatially invariant kernel for ease of notation. Sec. 4.2.1.3 explains how our method can be easily extended to the spatially varying case.

Formally, we seek to carry out the continuous convolutions

$$(f * g)(\mathbf{x}) = \int_{R^d} f(\mathbf{x} - \boldsymbol{\tau}) g(\boldsymbol{\tau}) \mathrm{d}\boldsymbol{\tau}. \tag{2.1}$$

This integral operation does not have a closed-form solution for all but the most constrained sets of signals and/or kernels. In particular, it is unclear how this continuous operation can be applied to generic neural fields, which naturally only support point samples. The typical solution for these integrals is numerical approximation: For low-dimensional integration domains, quadrature rules are feasible, while the scalable gold standard in higher dimensions is Monte Carlo integration. The latter

proceeds by sampling the integration domain and approximating the integral by a weighted sum of integrand evaluations:

$$(f * g)(\mathbf{x}) = \mathbb{E}_{\boldsymbol{\tau}} \left[f(\mathbf{x} - \boldsymbol{\tau}) g(\boldsymbol{\tau}) \right] \approx \frac{1}{N} \sum_{\boldsymbol{\tau} \sim p} \frac{f(\mathbf{x} - \boldsymbol{\tau}) g(\boldsymbol{\tau})}{p(\boldsymbol{\tau})}, \tag{2.2}$$

where $\mathbb{E}$ is the expectation and $\boldsymbol{\tau}$ are now random samples drawn from the probability density function p. Unfortunately, a high number N of samples is required for large kernels g and/or high-frequency signals f, rendering this approach inefficient.

In Chapter 4, we develop a method that performs continuous convolutions in the form of Eq. 2.1, while only requiring a very low number of network evaluations, independent of the kernel size.

2.2 VOLUMETRIC LIGHT TRANSPORT

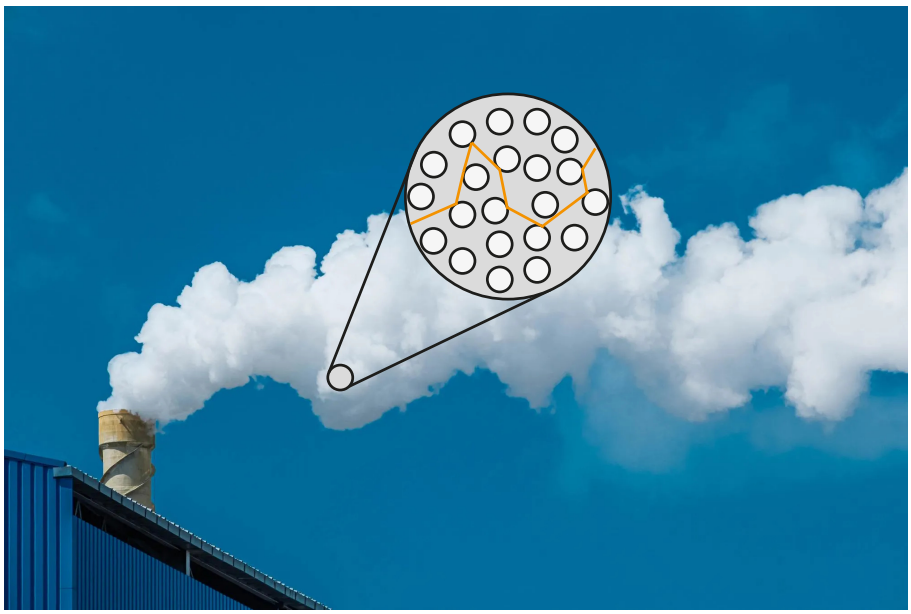

Figure 2.1: Particle–light interactions in a participating medium. A participating medium is modeled as a volume containing suspended particles distributed throughout space. As light propagates through the medium, it can undergo discrete interactions with these particles; the magnified inset illustrates such events, where an incident beam is scattered and redirected into new directions. Through the accumulation of many interactions, the medium attenuates and redistributes radiance along the light path, producing characteristic volumetric appearance effects. Image by Tanso Technologies Technologies, 2026.

A participating medium (also called a volume) is a region of space filled with particles that interact with light as it travels through the

region. Unlike a vacuum, where light propagates along straight paths without attenuation, participating media modify the energy transported through interactions with photons. While surface interactions are defined at material boundaries, where surfaces emit and reflect light at an interface, a participating medium can additionally absorb and scatter light throughout the volume, as illustrated in Fig. 2.1.

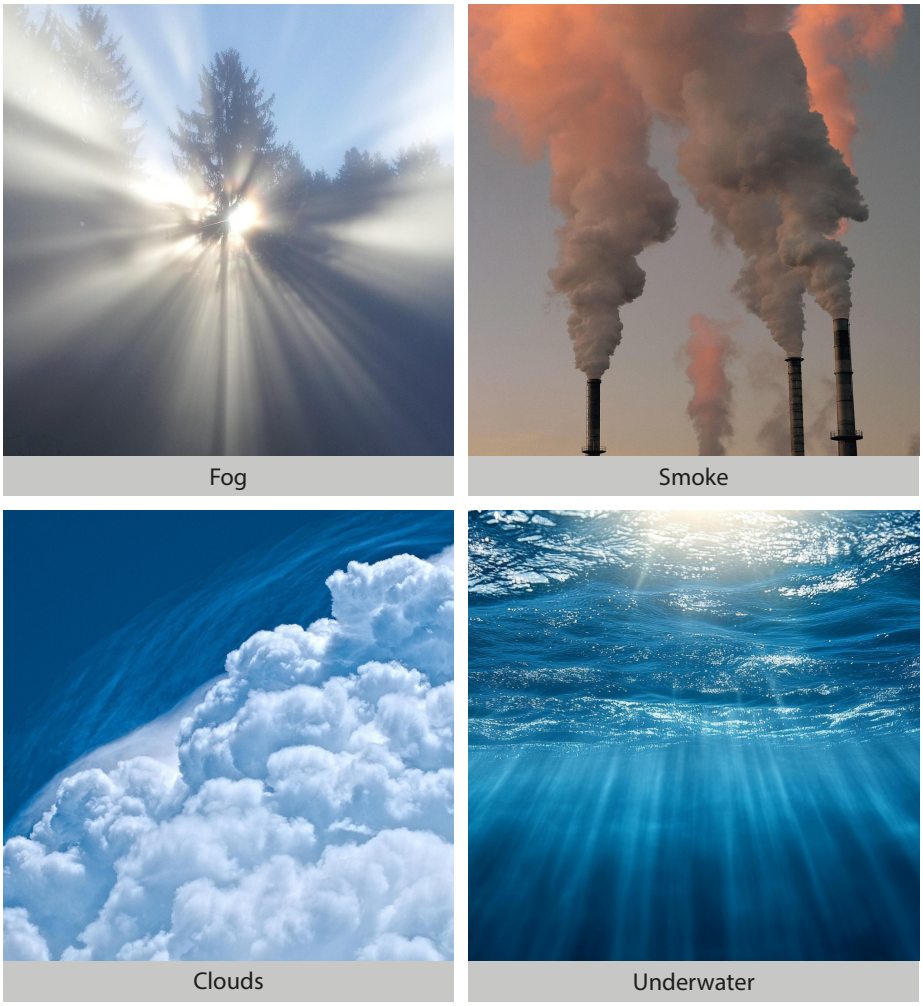

Figure 2.2: Examples of participating media: fog, smoke, clouds, and turbid underwater. The figure illustrates volumes in which light is attenuated and redistributed by interactions with suspended particles. Images by: hamist, 2026, shogun, 2026, dexmac, 2026, TungArt7, 2026.

Common examples of participating media include fog, smoke, clouds, and murky (turbid) water, as shown in Fig. 2.2, all of which can reduce visibility and produce characteristic volumetric effects such as softened contrast and visible light shafts.

The effect of a participating medium on light transport is described by its optical properties, most commonly the extinction, the albedo, and the phase function. In the following, we introduce these optical properties.

2.2.1 *Volumetric Optical Properties*

Extinction. We have previously stated that a participating medium is composed of particles interacting with light. In practice, however, one cannot explicitly model every single particle making up the medium. Instead, the medium is represented probabilistically via the extinction coefficient $\sigma_t(\mathbf{x})$ at $\mathbf{x} \in \mathbb{R}^3$. Intuitively, $\sigma_t(\mathbf{x})$ tells us how dense the medium is at $\mathbf{x}$, giving the local likelihood that a photon traveling through the medium will undergo an interaction at $\mathbf{x}$. These interactions are either absorption or scattering, described by the absorption coefficient $\sigma_a(\mathbf{x})$ and the scattering coefficient $\sigma_s(\mathbf{x})$, respectively, giving us the relative likelihood of the light being absorbed or scattered in a different direction, with

$$\sigma_t(\mathbf{x}) = \sigma_a(\mathbf{x}) + \sigma_s(\mathbf{x}).$$

Albedo. The appearance of a participating medium is determined by the amount of light it scatters, and the amount of light it emits. Assuming a non-emissive medium, the scattered light acts as the sole factor determining the final appearance. We define the single-scattering albedo as

$$\alpha(\mathbf{x}) = \frac{\sigma_s(\mathbf{x})}{\sigma_t(\mathbf{x})} = \frac{\sigma_s(\mathbf{x})}{\sigma_s(\mathbf{x}) + \sigma_a(\mathbf{x})}.$$

While the extinction dictates the likelihood of light interacting with the medium, the albedo dictates which type of interaction occurs. Since $\sigma_s(\mathbf{x}) \geq 0$ and $\sigma_a(\mathbf{x}) \geq 0$, it follows that $\alpha(\mathbf{x}) \in [0,1]$. Therefore, values near $\alpha(\mathbf{x}) = 1$ correspond to a purely scattering medium (all interactions are scattering, with negligible absorption), whereas $\alpha(\mathbf{x}) = 0$ corresponds to a purely absorbing medium (no scattering). Intermediate values indicate that interactions are split between scattering and absorption, with $\alpha(\mathbf{x})$ giving the fraction of extinction events that result in scattering. The visual effect of varying the albedo, ranging from predominantly scattering to predominantly absorbing, is illustrated in Fig. 2.3.

Phase function. When a scattering event occurs within a participating medium, the phase function $f_p(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}')$ describes the angular distribution of light after the interaction at position $\mathbf{x} \in \mathbb{R}^3$, where $\boldsymbol{\omega}' \in \mathbb{S}^2$ denotes the incident direction and $\boldsymbol{\omega} \in \mathbb{S}^2$ the scattered outgoing direction. In other words, it specifies the probability that light arriving from

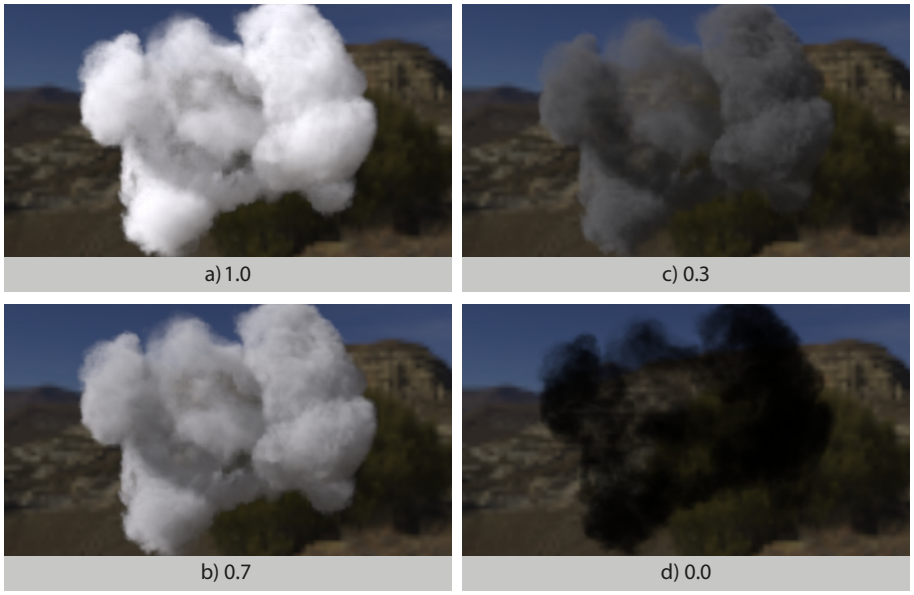

Figure 2.3: Effects of the single-scattering albedo on the appearance of a participating medium. As the albedo decreases from 1.0 to 0.0, the medium transitions from predominantly scattering to predominantly absorbing. For albedo values close to 1, nearly all interactions with the medium result in scattering, causing radiance to be redistributed throughout the volume and giving rise to a bright and diffuse volumetric appearance. As the albedo approaches 0, absorption becomes dominant, such that radiance is increasingly removed rather than redirected, leading to a darker appearance with minimal volumetric scattering. VDB asset by JangaFX, 2026. Environment Map by Zaal, 2026.

direction ω' at position $\mathbf{x}$ is scattered into direction ω. For a phase function to be physically valid, it must satisfy two fundamental properties. First, it must be non-negative, since scattering within a participating medium cannot produce negative radiance. Second, it must be normalized over the sphere, ensuring that the scattering process neither creates nor removes energy.

Depending on their scattering behavior, participating media can be classified as either isotropic or anisotropic. In an isotropic medium, scattering is independent of direction, meaning that light is equally likely to be scattered into any outgoing direction. The corresponding phase function is therefore constant and given by

$$f_{\mathrm{p}}(\mathbf{x}) = \frac{1}{4\pi}.$$

In an anisotropic medium, by contrast, scattering depends on the relative orientation of the incident direction ω' and the outgoing direction ω. Consequently, some scattering directions are favored over others.

Many participating media exhibit predominantly forward scattering, in which light is more likely to continue in a direction close to its incident direction, while others exhibit backward scattering. A commonly used analytic model for describing such anisotropic scattering is the Henyey–Greenstein phase function (Henyey and Greenstein, 1941):

$$f_{\mathrm{p}}(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}') = \frac{1}{4\pi} \frac{1 - g^2}{(1 + g^2 - 2g\cos\theta)^{3/2}}$$

where θ denotes the angle between the incident and outgoing directions, and $g \in [-1,1]$ is a user-specified parameter that controls the asymmetry of the phase function. Depending on the value of g, the Henyey–Greenstein phase function can model backward scattering ($g < 0$), forward scattering ($g > 0$), and isotropic scattering ($g = 0$). The shape of the phase function for different values of g is illustrated in Fig. 2.4.

2.2.2 *The Radiative Transfer Equation*

Having introduced the optical properties of participating media, we can now describe how they jointly govern the propagation of light through the volume. The full transport of radiance in scenes with participating media is described by the Radiative Transfer Equation (RTE) (Chandrasekhar, 1960; Pharr et al., 2023), an integro-differential equation that models the evolution of radiance under extinction, emission, and scattering.

We denote by $L(\mathbf{x},\boldsymbol{\omega}) \in \mathbb{R}^3$ the radiance at spatial position $\mathbf{x} \in \mathbb{R}^3$ traveling in direction $\boldsymbol{\omega} \in \mathbb{S}^2$. This quantity characterizes the spatio-angular distribution of light throughout the medium. The spectral dependence of this light field[1] and of all other quantities is modeled using three RGB channels.

Under the assumption of steady state, the RTE takes the form

$$\overbrace{(\boldsymbol{\omega} \cdot \nabla)\, L(\mathbf{x}, \boldsymbol{\omega})}^{\text{Transport}} = -\overbrace{\sigma_{\mathrm{t}}(\mathbf{x})\, L(\mathbf{x}, \boldsymbol{\omega})}^{\text{Extinction}} + \overbrace{\sigma_{\mathrm{a}}(\mathbf{x})\, L_{\mathrm{e}}(\mathbf{x}, \boldsymbol{\omega})}^{\text{Emission}} + \underbrace{\sigma_{\mathrm{s}}(\mathbf{x}) \int_{\mathbb{S}^2} f_{\mathrm{p}}(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\omega}')\, L(\mathbf{x}, \boldsymbol{\omega}')\, \mathrm{d}\boldsymbol{\omega}'.}_{\text{In-scattering}} \tag{2.3}$$

The left-hand side of Eq. 2.3 describes the directional transport of radiance through the medium, while the right-hand side accounts for local

1 We use the term *light field* to denote the spatio-angular distribution of radiance throughout the volume, and avoid the term *radiance field* to prevent confusion with emission–absorption-based scene representations (Mildenhall et al., 2020), in which "radiance" denotes a local appearance term that is conceptually closer to L_{e} than to the steady-state radiance L satisfying the RTE.

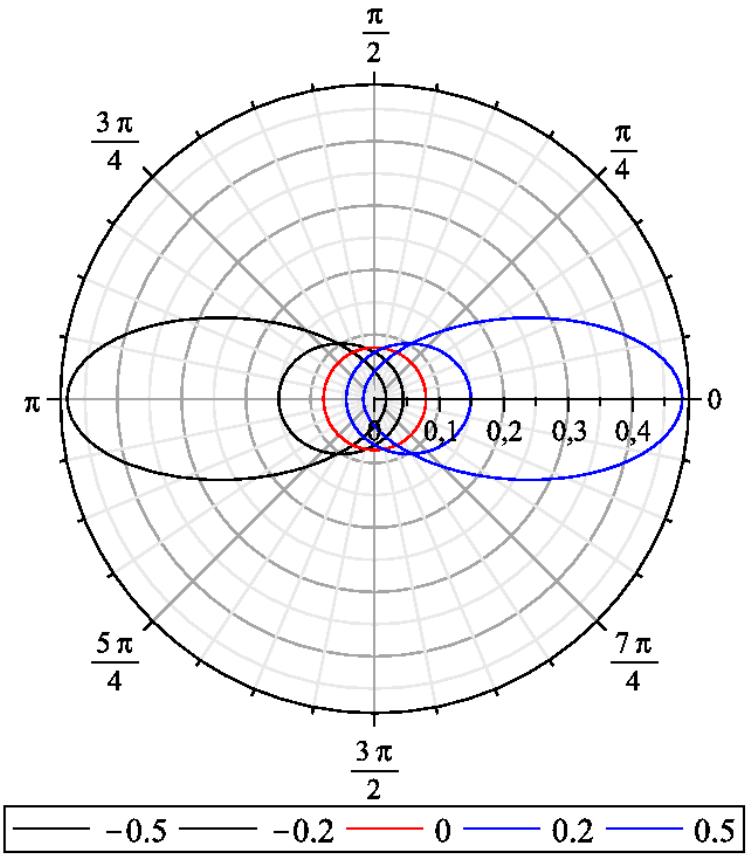

Figure 2.4: Shape of the Henyey–Greenstein phase function for different values of the asymmetry parameter g. Negative values of g correspond to predominantly backward scattering, positive values to predominantly forward scattering, and $g = 0$ to isotropic scattering. Image by JojoV, 2018.

loss due to extinction and local gain due to emission and in-scattering. The extinction term represents the removal of radiance through absorption, as illustrated in (Fig. 2.5a), and out-scattering, as illustrated in (Fig. 2.5b); the emission term models radiance generated locally within the medium, as shown in (Fig. 2.5c); and the in-scattering term redistributes incident radiance from all directions according to the phase function, as shown in (Fig. 2.5d). Although the scattering term integrates radiance over directions, all interactions in the RTE are local in space. Through this angular coupling induced by scattering, radiance at any location and direction arises from a dense superposition of indirect light transport contributions spanning the entire volume, corresponding to full global illumination.

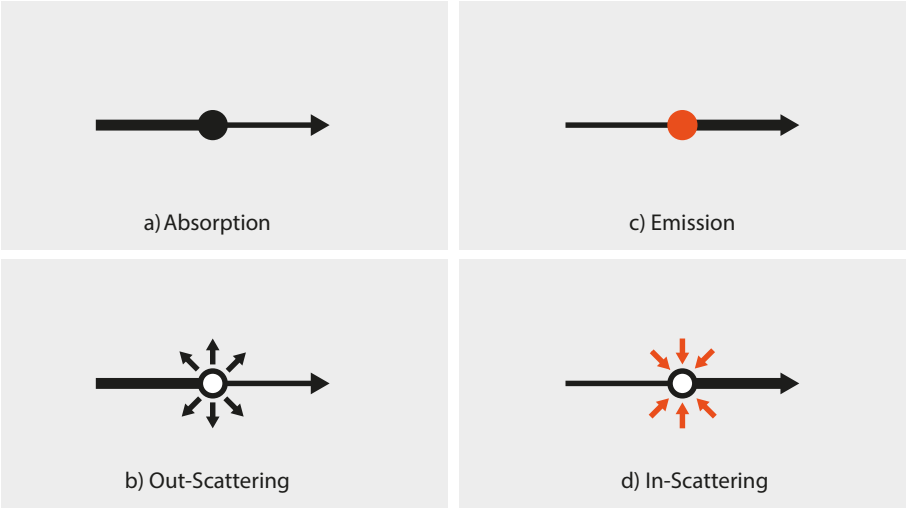

Figure 2.5: Illustration of light–matter interactions in a participating medium. As radiance propagates through the medium, it may undergo losses due to a) absorption, which removes energy from the beam, and b) out-scattering, which redirects radiance away from the direction of propagation. Conversely, radiance may increase through c) emission, whereby the medium itself contributes radiance, and d) in-scattering, in which radiance from other directions is redirected into the direction of propagation.

An explicit equation for $L(\mathbf{x},\boldsymbol{\omega})$ can be obtained by integrating both sides of Eq. 2.3. This integral form has a long history in radiative transfer physics Chandrasekhar, 1960, and is known in computer graphics as the Volume Rendering Equation (VRE) Kajiya and Von Herzen, 1984; Pharr et al., 2023:

$$L(\mathbf{x},\boldsymbol{\omega}) = \int_0^{t_b} \overbrace{\exp\left(-\int_0^t \sigma_\mathrm{t}(\mathbf{x}_s)\,\mathrm{d}s\right)}^{\text{Transmittance}} \Bigg[\overbrace{\sigma_\mathrm{a}(\mathbf{x}_t)\,L_\mathrm{e}(\mathbf{x}_t,\boldsymbol{\omega})}^{\text{Emission}} + \underbrace{\sigma_\mathrm{s}(\mathbf{x}_t)\int_{\mathrm{S}^2} f_\mathrm{p}(\mathbf{x}_t,\boldsymbol{\omega},\boldsymbol{\omega}')\,L(\mathbf{x}_t,\boldsymbol{\omega}')\,\mathrm{d}\boldsymbol{\omega}'}_{\text{In-scattering}} \Bigg] \mathrm{d}t. \tag{2.4}$$

Here, points along the ray $\mathbf{x}+\tau\,\boldsymbol{\omega}$ are denoted by $\mathbf{x}_\tau$ for $\tau \geq 0$, and t_b is the distance to the volume boundary. The volume rendering equation expresses radiance along a ray as an integral over local source terms, with the local extinction coefficient accumulated into a transmittance factor that encodes cumulative attenuation.

Both the differential RTE and the integral VRE are inherently recursive due to scattering-induced self-coupling of radiance; however, this recursion is expressed implicitly in the RTE through spatially local gain–loss

constraints, whereas in the VRE it appears explicitly through nested, spatially nonlocal ray integrals.

2.3 NEURAL FIELDS

A neural field, also referred to as an implicit neural representation, models a signal as a continuous function parameterized by a neural network. In many applications, this function is represented by a multilayer perceptron (MLP), although alternative architectures may also be employed (Bemana et al., 2020). Given spatial coordinates, optionally together with additional conditioning variables, the network predicts a target quantity of interest. Formally, this can be written as

$$\hat{\mathbf{y}} = f_\theta(\mathbf{x}, \mathbf{c}),$$

where $\mathbf{x} \in \mathbb{R}^{d_{in}}$ denotes the input coordinates, $\mathbf{c}$ optional conditioning variables, $\hat{\mathbf{y}} \in \mathbb{R}^{d_{out}}$ the predicted output, and θ the learnable network parameters. These parameters are learned by minimizing an objective function, typically using gradient-based optimization:

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathbf{x}\in\mathbb{R}^{d_{in}}} \left[\ell(f_\theta(\mathbf{x},\mathbf{c}), \mathbf{y})\right],$$

where $\mathbf{y}$ denotes the corresponding ground-truth target value and $\ell(\hat{\mathbf{y}}, \mathbf{y}$ is a suitable loss function. As optimization proceeds, the neural field gradually becomes a more accurate approximation of the underlying signal. The general training procedure of a 2D neural field is illustrated in Fig. 2.6.

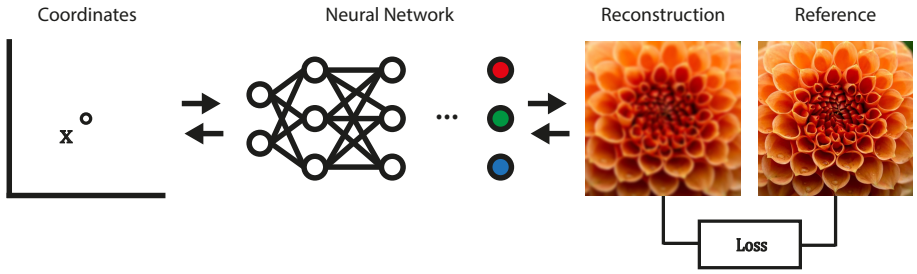

Figure 2.6: Overview of neural field training in 2D. A spatial coordinate is provided as input to an MLP, which predicts the corresponding signal value. The prediction is compared with reference data through a loss function, and the resulting gradients are backpropagated to optimize the network parameters. Right and left pointing arrows indicate the forward and backward passes, respectively.

In practice, neural fields implemented as standard MLPs are affected by the spectral bias, that is, they tend to fit low-frequency components of

a signal more readily than high-frequency ones (Rahaman et al., 2019). Consequently, when raw coordinates are used directly as input, such models often struggle to represent fine details, sharp transitions, and other high-frequency structure present in the target signal.

Two common strategies are used to address this limitation. The first is to enrich the input representation before it is processed by the network. The second is to modify the network itself so that it can better represent high-frequency variation.

The first strategy is commonly realized through positional encoding. The main idea is to map the input coordinates to a higher-dimensional space before passing them to the network, thereby enriching the representation with higher-frequency components that facilitate the learning of detailed structure. A widely used example is Fourier feature encoding (Rahimi and Recht, 2007; Tancik et al., 2020), in which the coordinates are transformed using sine and cosine functions at multiple frequencies:

$$\gamma(\mathbf{x}) = \left[\sin(2^0\pi\mathbf{x}), \cos(2^0\pi\mathbf{x}), \ldots, \sin(2^{L-1}\pi\mathbf{x}), \cos(2^{L-1}\pi\mathbf{x})\right].$$

Here, $\gamma(\mathbf{x})$ denotes the encoded representation of the input coordinate $\mathbf{x}$, and L is the number of frequency bands used in the encoding. By augmenting the input with sinusoidal functions at multiple frequencies, this encoding makes high-frequency structure more accessible to the MLP.

The second strategy is to improve the frequency representation capabilities of the network by using activation functions that are specifically designed to mitigate spectral bias. Such activation functions enable the network to capture a broader range of frequencies when learning the target signal. Examples activations include sinusoidal activations (Liu et al., 2024; Sitzmann et al., 2020), Gaussian activations (Ramasinghe and Lucey, 2022), wavelet activations (Saragadam et al., 2023), and sinc activations (Saratchandran et al., 2024). MLPs equipped with these activation functions can represent high-frequency content much more effectively even when operating directly on coordinates, thereby providing an alternative to positional encoding.

Neural fields possess several properties that make them attractive compared to explicit signal representations such as grids. **Continuity** follows from the fact that they define signals over continuous coordinates. **Discretization independence** allows them to be queried at arbitrary continuous coordinates, independently of any fixed discretization of the training data. **Differentiability** with respect to both network parameters and input coordinates makes them particularly suitable for optimization-based inverse problems. **Compactness** arises from storing the signal implicitly in the model parameters rather than explicitly at every sample

location. **Scalability** allows neural fields to handle complex and high-dimensional signals, since their memory and computational requirements are governed more by the complexity of the represented signal than by the dimensionality of its domain.

3 RELATED WORK

In this chapter, we review the related work relevant to the topics addressed in this thesis. We first discuss prior work on convolution and its applications, and then survey existing work on radiative transfer and light transport, providing context for the methods developed in the subsequent chapters.

3.1 CONVOLUTION

The convolution of a signal, eventually in some higher dimension, with a kernel is a central operation in modern signal processing (Zhang, 2022). In this thesis, we consider a slightly generalized form of convolution, where a kernel *varies spatially* across the signal.

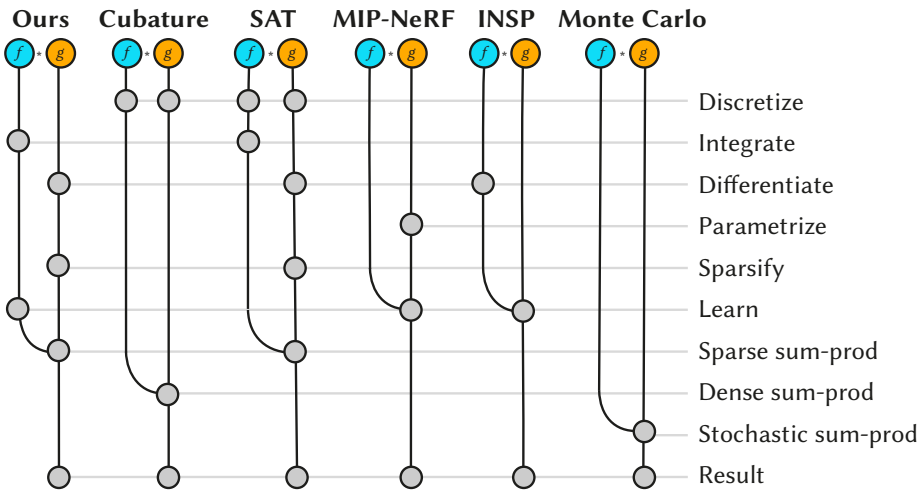

Figure 3.1: The landscape of convolution methods as combinations of different operations applied to a signal and kernel *(top)*, leading to a result *(bottom)*.

3.1.1 *Discrete Case*

For discrete signals and kernels (Cubature and SAT in Fig. 3.1), most convolutions are based on cubature, i. e., a dense sum-product operation across all dimensions. This, unfortunately, does not scale to larger kernels or higher dimensions but allows spatially-varying kernels. A common

acceleration is the *Fourier transform* (Brigham, 1988), which also requires time and space to perform and store the transformed signal. Most of all, it requires the kernel to be spatially invariant. As a remedy, certain spatially-varying convolutions can be realized using spatially-varying combinations or transformations of stationary filters (Fournier and Fiume, 1988; Freeman, Adelson, et al., 1991; Mitchel et al., 2020). For a large class of filters, *pyramidal* schemes (Farbman et al., 2011; Williams, 1983) can be a solution, but require additional memory. The key idea is that intermediate pyramid values store a partial aggregate of the signal. Techniques that store integrals without reducing the resolution are called *summed-area tables* (SAT) or *integral images* (Crow, 1984; Viola and Jones, 2001). Notably, SATs and their variants allow efficient spatially-varying convolution by considering differentiated kernels (Heckbert, 1986; Leimkühler et al., 2018; Simard et al., 1998). This efficiency comes from the fact that the differentiated kernel is sparse ("Sparsify" for SAT in Fig. 3.1) and the SAT only needs to be evaluated at very few locations. Our approach in chapter 4 will take this idea to the continuous neural domain.

3.1.2 *Continuous Case*

Convolution becomes more challenging, if the signal, the kernel, or both are continuous. *Monte Carlo* methods, that straightforwardly sample signal and kernel randomly and sum the result (Monte Carlo in Fig. 3.1) can handle this case. These scale very well to high dimensions, but at the expense of noise that only vanishes with many samples, even when specialized blue noise (Singh et al., 2019) or low-discrepancy (Niederreiter, 1992; Sobol, 1967) samplers are used. Similar to our approach in chapter 4, the use of derivatives has been shown to be beneficial (Kettunen et al., 2015). Practical unstructured convolution (Hermosilla et al., 2018; Shocher et al., 2020; Vasconcelos et al., 2023; Wang et al., 2018) does away with cubature and evaluates the product of kernel and signal only at specific sparse positions such as the points in a point cloud. Our approach in chapter 4 does not rely on random sampling but works directly on a continuous signal.

3.1.3 *Neural Case*

It has recently been proposed to replace discrete representations with continuous neural networks, so-called *neural fields* (Tewari et al., 2022; Xie et al., 2022). These have applications in geometry representation (Park et al., 2019), novel-view synthesis (Mildenhall et al., 2020; Sitzmann et al., 2019), dynamic scene reconstruction (Park et al., 2021; Tretschk et al., 2021;

Yuan et al., 2022), etc. Replacing a discrete grid with complex continuous functions requires developing the same operations available to grids (Dupont et al., 2022a), including convolutions, as we set out to do in chapter 4. Early work has been conducted to explore the manifold of all natural neural fields (Du et al., 2021) and to build a generative model of neural fields (Dupont et al., 2022b; Erkoç et al., 2023). Specialized network architectures allow the decomposition of signals into a discrete set of frequency bands (Fathony et al., 2020; Lindell et al., 2022; Yang et al., 2022). Further, limited forms of geometry processing have been considered in this representation (Yang et al., 2021). However, none of them is looking into general, efficient, large-scale, and/or spatially-varying convolutions.

A very specific form of convolution occurs as anti-aliasing or depth-of-field in image-based rendering. To account for these effects, neural fields can be learned that are conditioned on the parameters of the convolution kernel, such as its bandwidth (Barron et al., 2021, 2022; Isaac-Medina et al., 2023; Wang et al., 2022b). The network can then be evaluated to directly produce the filtered result, i. e., signal representation and convolution operation are intertwined. This Mip-NeRF-style convolution is in principle applicable to other filters, as long as they can be parametrized to become conditions to input into the network (MIP-NeRF in Fig. 3.1). Unfortunately, this limits the kernels that can be applied to a parametric family that needs to be known in advance. Further, it significantly increases training time, since kernel parameters act as additional input dimensions to the network. We use inductive knowledge of integration and differentiation to arrive at a more efficient formulation that generalizes across kernels.

The key to efficient convolutions is a combination of sparsity, differentiation, and integration. Fortunately, tools to perform integration and differentiation on neural fields are available. AutoInt (Lindell et al., 2021) proposes to learn a neural network that, when automatically differentiated, fits a signal. By evaluating the original network without differentiation, the antiderivative can be evaluated conveniently. Unfortunately, this approach does not scale well to the higher-order antiderivatives needed for efficient convolutions, as the size of the derivative graphs grows quickly. In contrast, our approach leverages higher-order finite differences to train a repeated integral field, which scales with the number of integrations required.

Recently, (Xu et al., 2022) have proposed a method with the same aim as ours (INSP in Fig. 3.1). Given a trained neural field, they learn to combine higher-order derivatives of the field to approximate a convolution. Similar to a Taylor expansion, this requires high-order derivatives to reason about

larger neighborhoods and, unfortunately, hence only allows for very small, spatially invariant kernels.

Finally, aggregation in neural fields in the form of range queries has been studied by (Sharp and Jacobson, 2022). Their approach allows to retrieve a conservative estimate of the field's extrema within a query volume, which unfortunately does not provide enough information to perform accurate continuous convolutions.

3.2 INVERSE RENDERING AND DIFFERENTIABLE LIGHT TRANSPORT

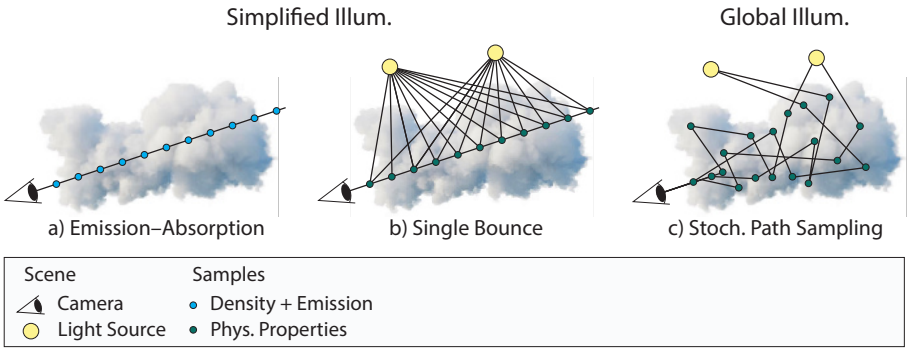

Figure 3.2: Different approaches to volumetric inverse rendering. *(a)* The emission–absorption model bakes illumination into the volume, preventing recovery of physically meaningful optical properties. *(b)* Restricting light transport to a single bounce enables limited relighting but fails to capture the complexity of global illumination. *(c)* Differentiable stochastic path sampling accounts for global illumination during reconstruction, but requires substantial methodological and engineering effort.

3.2.1 *Discretization-Based Inverse Radiative Transfer*

Inverse radiative transfer problems have traditionally been solved by numerically discretizing the radiative transfer equation Adams and Larsen, 2002; McCormick, 1992. Traditional methods typically represent the light field and the optical properties using linear basis expansions defined on a computational mesh, resulting in a finite-dimensional approximation of the governing equation Bangerth and Joshi, 2008. The optical properties are then inferred by iteratively solving the resulting system. These approaches have been successfully applied in fields such as atmospheric remote sensing Levis et al., 2015, 2017, medical imaging Abdoulaev et al.,

2005 and constitute the foundation of many established radiative transfer frameworks Stamnes et al., 2000. While traditional methods rely on discretization, our proposed method in chapter 5 is based on a physics-informed learning framework, where we represent both the light field and the optical properties as neural networks. The governing physics and observational constraints are then enforced through the optimization objective.

3.2.2 *Differentiable Monte Carlo Light Transport*

Inverse volume rendering is traditionally addressed by differentiating stochastic light transport simulations, typically based on path tracing, to optimize the optical properties of participating media (Fig. 3.2c). Gradients of image measurements with respect to scene parameters are estimated by differentiating Monte Carlo estimators of the rendering equation, enabling optimization under physically complete light transport Nimier-David et al., 2019; Zhang et al., 2019, 2021a. This framework has been applied to volumetric and translucent materials, allowing inverse estimation of optical properties in participating media Deng et al., 2022, 2024; Gkioulekas et al., 2016, 2013; Weier et al., 2025. While conceptually general, differentiable Monte Carlo methods are computationally expensive and often limited by high gradient variance, motivating extensive work on improving efficiency and stability Nicolet et al., 2023; Nimier-David et al., 2020; Vicini et al., 2021. Moreover, multiple scattering requires differentiating an expectation over a stochastic branching process. Obtaining gradients which are useful in optimization requires substantial methodological and engineering effort and remains an active area of research Nimier-David et al., 2022; Worchel et al., 2025; Zeltner et al., 2021. Our approach in chapter 5 sidesteps these challenges by avoiding path sampling altogether, while still accounting for physically complete light transport.

3.2.3 *Inverse Rendering with Learned Representations*

Inverse rendering approaches based on learned representations commonly adopt simplified light transport models. A prominent example is the emission–absorption formulation (Fig. 3.2a), which represents volumes as purely emissive and neglects explicit illumination and scattering (Kajiya and Von Herzen, 1984; Kerbl et al., 2023; Mildenhall et al., 2020). Single-scattering approaches constitute the next level of modeling complexity by accounting for direct illumination and a single light bounce (Fig. 3.2b) A common paradigm in this setting is to infer surface-

based material properties, such as albedo and normals (Bi et al., 2020; Dihlmann et al., 2024; Gao et al., 2024; Li et al., 2024; Liang et al., 2024; Zhang et al., 2021b; Zhang et al., 2021c). Accuracy can be further improved by explicitly modeling incident illumination using specialized components (Boss et al., 2021; Fan et al., 2025; Srinivasan et al., 2021; Yao et al., 2022) and by introducing physics-inspired regularization terms that encourage consistency with simplified reflectance models (Wu et al., 2025). Recent work has also explored learning-based solutions for volumetric participating media with anisotropic scattering (Zhang et al., 2023), but remains limited to restricted transport models. Multi-bounce illumination in inverse rendering has been considered more rarely. In this setting, surface-based representations can leverage precomputed radiance transfer (Lyu et al., 2022), caching techniques (Shi et al., 2025), or explicit global-illumination decomposition strategies (Chen et al., 2025; Jin et al., 2023). For volumetric scenes, however, existing approaches typically rely on strong assumptions, such as homogeneous media (Che et al., 2020), or on simplified treatments of higher-order light transport (Zheng et al., 2021). In contrast, our approach in chapter 5 does not rely on such restrictive assumptions and directly targets physically complete volumetric light transport.

3.3 PHYSICS-INFORMED LEARNING FOR LIGHT TRANSPORT AND SCENE RECONSTRUCTION

Physics-informed neural networks (PINNs) integrate physical laws, typically expressed as (integro-)differential equations, into neural network training by penalizing violations of these equations in the loss function (Raissi et al., 2019). Incorporating the governing equations of volumetric light transport into such frameworks has been explored in both forward and inverse settings (Mishra and Molinaro, 2021; Riganti and Negro, 2023; Zucker et al., 2025). These works focus on ultra low-frequency optical properties and light fields, and often are further restricted to low-dimensional (1D or 2D) settings, whereas our approach in chapter 5 operates in full 3D and aims to reconstruct spatially complex, high-frequency volumetric structure, as required in inverse graphics. This is achieved using an explicit integration scheme along primary rays, which helps alleviate the low-frequency bias of neural networks in general Rahaman et al., 2019 and of PINN-based formulations in particular Wang et al., 2022a.

Physics-informed learning has also been explored for light transport in surface-based scenes, in both forward (Hadadan et al., 2021) and inverse (Hadadan et al., 2023) settings. In these surface-based formulations,

non-local transport through free space is naturally handled via stochastic ray sampling, whereas our volumetric formulation in chapter 5 enforces light transport locally throughout the volume by imposing the RTE as a pointwise constraint.

Optimizing hard geometry, i. e.surfaces with sharp discontinuities in PINN-based inverse rendering remains challenging and has not yet been demonstrated, whereas our method naturally supports the optimization of soft geometry (e.g., continuous volumetric media such as clouds or fog).

Another line of work addresses the neural reconstruction of dynamic volumes by incorporating physical soft constraints derived from fluid dynamics Chu et al., 2022; Yu et al., 2024. While these approaches model the transport of matter in participating media, our work instead focuses on the transport of light.

3.4 GENERATIVE MODELS OF OPTICAL PROPERTIES

Recent work has explored generative modeling of scene appearance and optical properties, typically under simplified image formation models. A first line of work focuses on surface-based representations and learns generative models of material appearance under simple shading assumptions, often limited to direct illumination (Chen et al., 2023b; Jiang et al., 2023; Pan et al., 2021; Violante et al., 2024; Zhang et al., 2024). These methods do not model volumetric light transport or participating media. A second line of work considers volumetric scene representations and learns generative models under emission–absorption rendering (Chan et al., 2022; Chen et al., 2023a; Diolatzis et al., 2023; Henzler et al., 2019; Müller et al., 2023). While some approaches target volumetric phenomena such as participating media, the representations remain appearance-driven and do not explicitly model physical optical parameters or enforce global illumination. To our knowledge, no prior work enables generative modeling of physical volumetric optical properties, such as absorption, scattering, and phase functions, under full global illumination.

4

NEURAL FIELD CONVOLUTIONS BY REPEATED DIFFERENTIATION

Neural fields are evolving towards a general-purpose continuous representation for visual computing. Yet, despite their numerous appealing properties, they are hardly amenable to signal processing. As a remedy, we present a method to perform general continuous convolutions with general continuous signals such as neural fields. Observing that piecewise polynomial kernels reduce to a sparse set of Dirac deltas after repeated differentiation, we leverage convolution identities and train a repeated integral field to efficiently execute large-scale convolutions. We demonstrate our approach on a variety of data modalities and spatially-varying kernels.

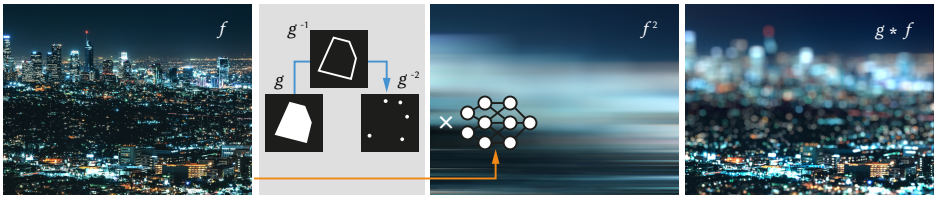

Figure 4.1: We introduce an algorithm to perform efficient continuous convolution of neural fields f by piecewise polynomial kernels g. The key idea is to convolve the sparse repeated derivative of the kernel (g^{-n}) with the repeated antiderivative of the signal (f^n).

4.1 INTRODUCTION

Neural fields have recently emerged as a powerful way of representing signals and have witnessed widespread adoption in particular for visual data (Tewari et al., 2022; Xie et al., 2022). Also referred to as implicit or coordinate-based neural representations, neural fields typically use a multi-layer perceptron (MLP) to encode a mapping from coordinates to values. This representation is universal and allows to capture a multitude of modalities, such as mapping from 2D location to color for images (Stanley, 2007), from 3D location to the signed distance to a surface for geometry (Park et al., 2019), from 5D light field coordinates to emitted radiance of an entire scene (Mildenhall et al., 2020), and many more.

The appealing properties of neural fields are three-fold: First, they represent signals in a continuous way, which is a good fit for the mostly continuous visual structure of our world. Second, they are compact, since they encode complex signals into a relatively small number of MLP weights (Dupont et al., 2021), while adapting well to local signal complexities. Third, they are easy to optimize by construction. Taking all of these properties together, it comes at no surprise that neural fields are rapidly evolving towards a general-purpose data representation (Dupont et al., 2022a). However, to be a true alternative to established specialized representations such as pixel arrays, meshes, point clouds, etc., neural fields are still lacking in a fundamental aspect: They are hardly amenable to *signal processing* (Xu et al., 2022; Yang et al., 2021). As a remedy, in this work, we propose a general framework to apply a core signal processing technique to neural fields: convolutions.

The versatility and expressivity of neural representations have evolved significantly over the last couple of years, mostly due to advances in architectures and training methodologies (Hertz et al., 2021; Müller et al., 2022; Sitzmann et al., 2020; Tancik et al., 2020). However, at their core, neural fields only support *point samples*. This is sufficient for point operations, such as the remapping of input coordinates, e. g. for the purpose of deformations (Kopanas et al., 2022; Park et al., 2021; Tretschk et al., 2021; Yuan et al., 2022), or the remapping of output values (Vicini et al., 2022). In contrast, a convolution requires the *continuous integration* of values over coordinates weighted by a continuous kernel.

Aggregation in neural fields can be approximated using either discretization followed by cubature, or Monte Carlo sampling, resulting in excessive memory requirements and noise, respectively. Another solution is to consider a narrow parametric family of kernels and train the field using supervision on filtered versions of the signal (Barron et al., 2021). Representation and learned convolution operation can be explicitly disentangled using higher-order derivatives (Xu et al., 2022), but this comes at the cost of only supporting small spatially invariant kernels. AutoInt (Lindell et al., 2021) performs analytic integration using automatic differentiation, but only considers unweighted integrals. We advance the state of the art by presenting a method to *efficiently* perform *general, large-scale, spatially-varying* convolutions *natively* in neural fields.

In our approach, we consider neural fields to be convolved with piecewise polynomial kernels, which reduce to a sparse set of Dirac deltas after repeated differentiation (Heckbert, 1986). Combining this insight with convolution identities on differentiation and integration, our approach requires only a small number of point samples from a neural integral field to perform an exact continuous convolution, independent

of kernel size. This integral field needs to be trained in a specific way, supervised via continuous higher-order finite differences, corresponding to a minimal kernel of a certain polynomial degree. Once trained, our neural fields are ready to be convolved with *any* piecewise polynomial kernel of that degree.

We showcase the generality and versatility of our approach using different modalities, such as images, videos, geometry, character animations, and audio, all natively processed in a neural field representation. Further, we demonstrate (spatially-varying) convolutions with a variety of kernels, such as smoothing and edge detection of different shapes and sizes. In summary, our contributions are:

- A principled and versatile framework for performing convolutions in neural fields.
- Two novel enabling ingredients: An efficient method to train a repeated integral field, and the optimization of continuous kernels that are sparse after repeated differentiation.
- The evaluation of our framework on a range of modalities and kernels.

4.2 METHOD

We efficiently convolve a continuous signal f with a continuous kernel g by approximating g with a piecewise polynomial function, which becomes sparse after repeated differentiation. Our approach requires the evaluation of the repeated integral of f at a sparse set of sample positions dictated by the differentiated kernel (Sec. 4.2.1), leading to a substantial speed-up of the convolution operation. This general approach has first been studied by (Heckbert, 1986) in the context of discrete representations. Our method lifts the idea to the continuous neural setting by optimizing for a sparse differential representation of g (Sec. 4.2.2), and obtaining the repeated continuous integral of f by supervising on a minimal kernel using higher-order finite differences (Sec. 4.2.3). Once trained, any sparsity-optimized convolution kernel can be applied to f efficiently, without requiring additional input parameters. Our method supports spatially-varying convolutions in the form of continuously transformed kernels, leveraging the continuous nature of the representation. Fig. 5.3 gives an overview of our approach.

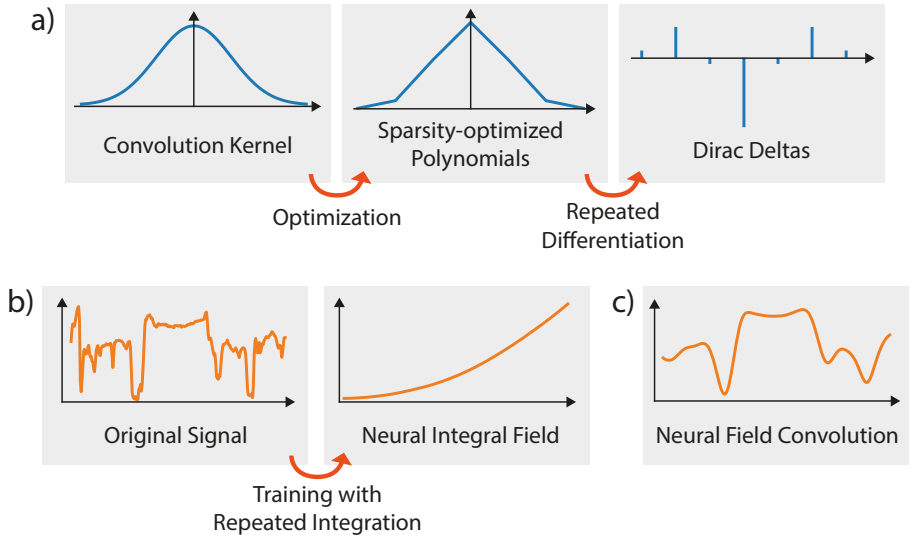

Figure 4.2: Overview of our approach. *a*) Given an arbitrary convolution kernel, we optimize for its piecewise polynomial approximation, which under repeated differentiation yields a sparse set of Dirac deltas. *b*) Given an original signal, we train a neural field that captures the repeated integral of the signal. *c*) The continuous convolution of the original signal and the convolution kernel is obtained by a discrete convolution of the sparse Dirac deltas from *a*) and corresponding sparse samples of the neural integral field from *b*).

4.2.1 *Convolution by Repeated Differentiation*

Our approach requires two conceptual ingredients: First, the convolution operation in Eq. 2.1 reduces to a discrete sum if g consists of only Dirac deltas (Sec. 4.2.1.1). Second, the right-hand side of Eq. 2.1 can be transformed using repeated differentiation and integration (Sec. 4.2.1.2). Putting both ingredients together leads to an efficient discrete formulation of the continuous convolution operation (Sec. 4.2.1.3) involving a repeated integral field and a sparse differential kernel consisting of Dirac deltas (Heckbert, 1986).

4.2.1.1 *Dirac Kernels*

As our first ingredient, consider a kernel g that is non-zero only at a small set of m locations in R^d, i. e., g is

$$g(\mathbf{x}) = \sum_{i=1}^{m} \delta(\mathbf{x} - \mathbf{x}^{(i)}) w^{(i)}, \tag{4.1}$$

a sum of Dirac deltas δ, where $\mathbf{x}^{(i)} \in R^d$ denotes the location and $w^{(i)} \in R$ the magnitude[1] of the i'th impulse. Then, by the *sifting* property of Dirac deltas, Eq. 2.1 simplifies to

$$(f * g)(\mathbf{x}) = \sum_{i=1}^{m} f(\mathbf{x} - \mathbf{x}^{(i)}) w^{(i)}, \tag{4.2}$$

i. e., we have reduced the computation from a continuous integral to a discrete sum – an efficient operation if m is small.

4.2.1.2 *Convolutions with Differentiation and Integration*

Our second ingredient is the following identity:

$$f * g = \left(\int f \mathrm{d}\mathbf{x}_i \right) * \left(\frac{\partial}{\partial \mathbf{x}_i} g \right),$$

i. e., in order to convolve f with g we might as well convolve the antiderivative of f with the corresponding derivative of g. Applying this principle repeatedly yields (Heckbert, 1986; Perlin, 1984)

$$f * g = \underbrace{\left(\int^n \dots \int^n f \mathrm{d}\mathbf{x}_1^n \dots \mathrm{d}\mathbf{x}_d^n \right)}_{f^n} * \underbrace{\left(\frac{\partial^{dn}}{\partial \mathbf{x}_1^n \dots \partial \mathbf{x}_d^n} g \right)}_{g^{-n}}. \tag{4.3}$$

Here, we sequentially differentiate g n times with respect to *each* of its dimensions. We denote this multidimensional repeated derivative as g^{-n}. For equality in Eq. 4.3 to hold, this pattern is mirrored for f, replacing differentiations with antiderivatives, where superscripts n denote repeated integrations along the individual dimensions. We denote the repeated multidimensional antiderivative of f as f^n. We refer to (Heckbert, 1986) for a proof of Eq. 4.3. Notice that input and output dimensions of f and g do not change after integration and differentiation.

4.2.1.3 *Efficient Neural Field Convolutions*

The central idea of our approach is to combine both ingredients presented above for the case of piecewise polynomial kernels $\hat{g}$. Concretely, we observe that piecewise polynomial functions turn into a sparse set of Dirac deltas after repeated differentiation (Heckbert, 1986) (Fig. 4.3), i. e., $\hat{g}^{-n}$ reduces to the form of Eq. 4.1. This implies that Eq. 4.2 can be used

1 Technically, $\delta(\mathbf{0}) = \infty$. But since a Dirac delta integrates to one, in the context of continuous convolutions, we refer to $w^{(i)}$ as "magnitudes" nevertheless.

Figure 4.3: Repeated differentiation of a bilinear patch ($d = n = 2$). After the first differentiation, the patch exhibits linear variation only along the vertical dimension. After subsequent differentiations, we obtain a constant patch, two vertical lines, and, finally, four Dirac deltas.

to perform a convolution with this kernel. Combining Eq. 4.2 and Eq. 4.3, our final convolution operation reads

$$\boxed{(f * \hat{g})(\mathbf{x}) = \sum_{i=1}^{m} f^n(\mathbf{x} - \mathbf{x}^{(i)}) w^{(i)}.} \tag{4.4}$$

Notice that this formulation requires only m evaluations of the repeated integral of f at locations dictated by the Dirac deltas of the differentiated kernel to yield the same result as the equivalent continuous convolution in Eq. 2.1.

The number of integrations and differentiations n directly depends on the desired order of the kernel polynomials, as detailed in Sec. 4.2.2. A disk-shaped kernel simulating thin-lens depth of field in an image can be approximated well using a piecewise *constant* function (corresponds to $n = 1$), while a Gaussian might require a piecewise *quadratic* approximation (corresponds to $n = 3$) to yield high-quality results with a low number of Dirac deltas.

Notice that our approach allows us to realize *spatially-varying* convolutions as well: The evaluation of Eq. 4.4 is independent for different evaluation locations $\mathbf{x}$. Therefore, we can make the choice of the convolution kernel $\hat{g}$ a function of $\mathbf{x}$ itself. In Sec. 4.2.2.2 we give details on how to obtain continuous parametric kernel families.

In summary, our method requires two components: (*i*) A piecewise polynomial kernel that results in a sparse set of Dirac deltas after repeated differentiation, and (*ii*) an efficient way to obtain and evaluate the repeated multidimensional integral of a continuous signal. These are detailed in Sec. 4.2.2 and Sec. 4.2.3, respectively.

4.2.2 *Sparse Differential Kernels*

Our approach requires a kernel g which, after repeated differentiation, results in a sparse set of Dirac deltas with positions $\mathbf{x}^{(i)}$ and magnitudes $w^{(i)}$:

$$g^{-n}(\mathbf{x}) = \sum_{i=1}^{m} \delta(\mathbf{x} - \mathbf{x}^{(i)}) w^{(i)}. \tag{4.5}$$

This property is satisfied for piecewise polynomial kernels of degree $n-1$, which reduce to Dirac deltas positioned at the junctions between the segments after n differentiations per dimension (Heckbert, 1986) (Fig. 4.3). Thus, given a kernel g, we seek to find its optimal piecewise polynomial approximation $\hat{g}$ adhering to a user-specified budget of m Dirac deltas (Fig. 4.4).

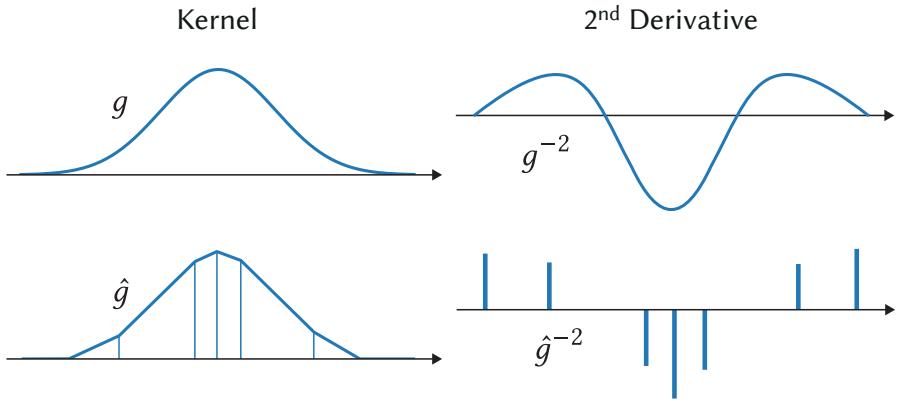

Figure 4.4: Kernel representation in 1D for the case $n = 2$, i. e., a piecewise linear function. *Top row*: The original continuous kernel g has a continuous second derivative. *Bottom row*: We approximate g with a piecewise linear function $\hat{g}$, which reduces to a sparse set of Dirac deltas in its second derivative.

To parameterize $\hat{g}$, we utilize the linear structure of Eq. 4.5 and the linearity of differentiation: We consider the d-dimensional n-fold repeated antiderivative of the Dirac delta function

$$\delta^n(\mathbf{x}) = \begin{cases} \frac{\prod_{i=1}^{d} \mathbf{x}_i^{n-1}}{(n-1)^d!} & \min_i \mathbf{x}_i \geq 0 \\ 0 & \text{else} \end{cases}$$

which is referred to as the n'th-order *ramp* (Fig. 4.5). We now write our polynomial kernel $\hat{g}$ as a linear combination of shifted ramps:

$$\hat{g}(\mathbf{x}) = \sum_{i=1}^{m} \delta^n(\mathbf{x} - \mathbf{x}^{(i)}) w^{(i)}. \tag{4.6}$$

Please note that Equation (4.5) is the n'th derivative of Equation (4.6) by construction. Thus, we have established a parameterization of a C^{n-2}-continuous piecewise polynomial kernel $\hat{g}$, from which we can directly read off Dirac delta positions and magnitudes.

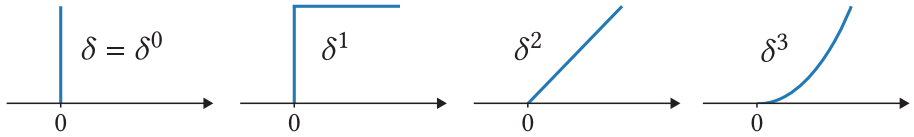

Figure 4.5: 1D ramps of different orders (redrawn from (Heckbert, 1986)). Each ramp is the antiderivative of its predecessor.

We now optimize the following objective:

$$\min_{\mathbf{x}^{(i)}, w^{(i)}} \left[\mathbb{E}_{\mathbf{x} \in R^d} \left[\| g(\mathbf{x}) - \hat{g}(\mathbf{x}) \|_2^2 \right] + \lambda \left| \sum_{i=1}^{m} w^{(i)} \right| \right]. \tag{4.7}$$

The first term encourages the solution to be close to the reference kernel on the entire domain. The second term steers the optimization to prefer solutions where the Dirac magnitudes sum to zero, which effectively enforces the kernel to be compact.

4.2.2.1 *Optimization*

We initialize the ramp positions $\mathbf{x}^{(i)}$ on a regular grid and their magnitudes $w^{(i)}$ to zero. We use the Adam (Kingma and Ba, 2015) optimizer with standard parameters and set $\lambda = 0.1$ in all our experiments. In each iteration, we uniformly sample the continuous kernel domain. We observe that when the provided ramp budget m is too high, the magnitudes of individual ramps will approach zero, further increasing sparsity. We capitalize on this fact by monitoring ramp magnitudes in regular intervals. If an absolute magnitude falls below a small threshold, we remove the ramp from the mixture and continue optimizing. Separable kernels can be obtained by optimizing the respective 1D filters and combining them with an outer product. We provide timings of the optimization for different kernels in Tab. 4.1. Here, we consider a 1D Gaussian kernel, represented by different numbers m of Diracs, using different orders of differentiation n. We give the reconstruction error in terms of the mean squared error (MSE) and the time (in seconds) our unoptimized implementation takes to obtain a converged result.

We note the strong connection to spline-based approximations of functions (Ahlberg et al., 2016). In the low-order regime under the continuity requirements we operate on, we find that our practical stochastic gradient

Table 4.1: Accuracy of kernels and time to optimize them.

m	3		7		13		24	
n	1	2	1	2	1	2	1	2
MSE	1.6e-1	1.5e-2	1.5e-2	7.9e-4	4.0e-3	7.6e-5	1.1e-3	2.3e-5
Time (s)	3	25	4	60	6	75	9	132

descent-based approach yields high-quality results without the need for more elaborate techniques.

4.2.2.2 *Kernel Transformations*

Many applications of convolutions require kernels of different sizes and shapes, in particular in the case of spatially-varying convolutions. For example, foveated imagery or the simulation of depth-of-field require continuous and fine-grained control over the size of a blur kernel. Our approach supports on-the-fly kernel transformations without the need to sample and optimize entire parametric families of kernels by leveraging the continuous nature of the kernel and the signal.

Concretely, to continuously shift and (anisotropically) scale an optimized kernel $\hat{g}$ using a matrix T, we simply apply T to the Dirac delta positions, i. e., $\mathbf{x}_\mathsf{T}^{(i)} = \mathsf{T}\mathbf{x}^{(i)}$. The updated Dirac delta magnitudes are given by $w_\mathsf{T}^{(i)} = \frac{w^{(i)}}{\det(\mathsf{T})^n}$. Thus, we need to run the optimization for a kernel type only once in a canonical position and size, and obtain continuously transformed kernel instances at virtually no computational cost. Notice that, as a useful consequence, our approach enables continuous scale-space analysis (Lindeberg, 2013; Witkin, 1987), as illustrated in Figure 4.9.

4.2.3 *Neural Repeated Integral Field*

To compute Equation (4.4), we need to evaluate f^n, the n-th antiderivative of f as the second ingredient. We choose to implement f^n as a neural field $\hat{f}^n$. Ideally, it would hold that $\hat{f}^n = f^n$. This might be difficult to achieve without knowing an analytic form of the antiderivative. We could try Monte Carlo-estimating the antiderivative from the signal, leading to a loss like

$$\mathbb{E}_{\mathbf{x}\in\mathbb{R}^{d_\text{in}}}\left[\left\|\hat{f}^n(\mathbf{x}) - \mathbb{E}_{\boldsymbol{\tau}\geq 0}\left[f(\mathbf{x}-\boldsymbol{\tau})\right]\right\|\right]. \tag{4.8}$$

The inner expectation would sum over the entire half-domain $\boldsymbol{\tau} \geq 0$, leading to a high variance and a low-quality $\hat{f}^n$. An example of this is

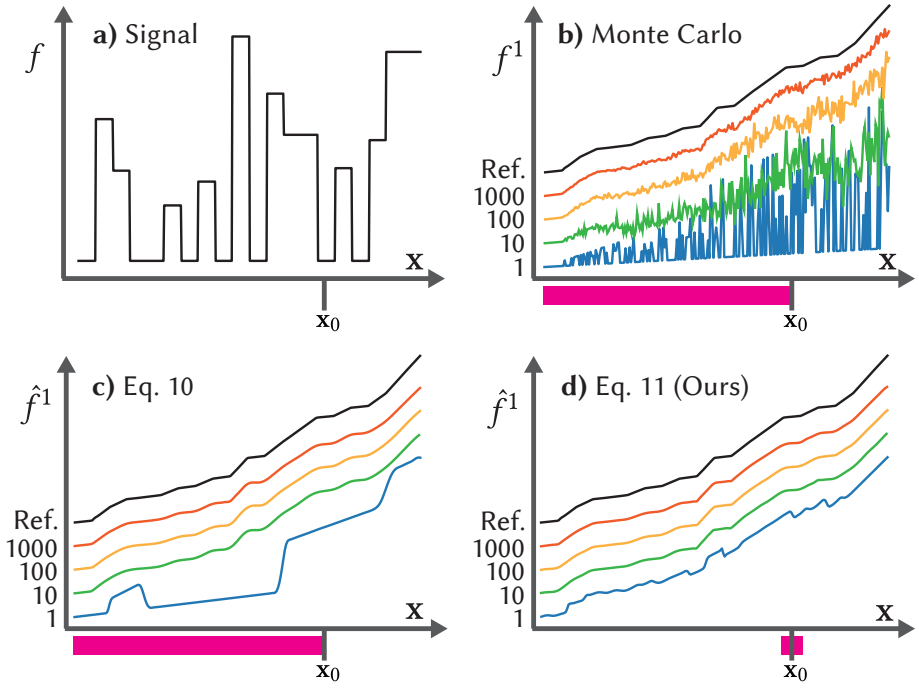

Figure 4.6: (*a*): Given a signal f, we are interested in finding its antiderivative f^1. (*b*): A scalable way to obtain the antiderivative is Monte Carlo estimation $\mathbb{E}_{\boldsymbol{\tau}\geq 0}\left[f(\mathbf{x}-\boldsymbol{\tau})\right]$. Increasing the number of samples (different colors) gets us closer to the true solution (black). (*c*): Using the estimates from *b*) to supervise the training of a network $\hat{f}^1$ as per Eq. 4.8 requires many samples, while the regressed antiderivative remains blurry. (*d*): In contrast, our approach only requires a convolution with a small kernel (Eq. 4.9) to yield high-quality results, including sharp features, with only a low number of Monte Carlo samples. For each method, a pink bar marks the region that needs to be considered for estimating/training the antiderivative value at location $\mathbf{x}_0$. The different graphs in *b*)-*d*) use different constants of integration for improved visualization.

shown in Figure 4.6, where the input is a 1D HDR signal (Figure 4.6a). To estimate the antiderivative at $\mathbf{x}_0$, we have to sample the entire pink area in Figure 4.6b, resulting in significant variance, even if the sample count increases. When using these estimates to train $\hat{f}^n$, this variance leads to a low-quality, blurry regression (Figure 4.6c), as no loss function is known that properly captures the statistical properties of Monte Carlo noise (Lehtinen et al., 2018).

For our purpose, convolution, what really needs to hold, is that $\hat{f}^n * h_n^{-n} = f * h_n$, for any piecewise polynomial kernel h_n of degree $n-1$. The resulting loss to achieve this is

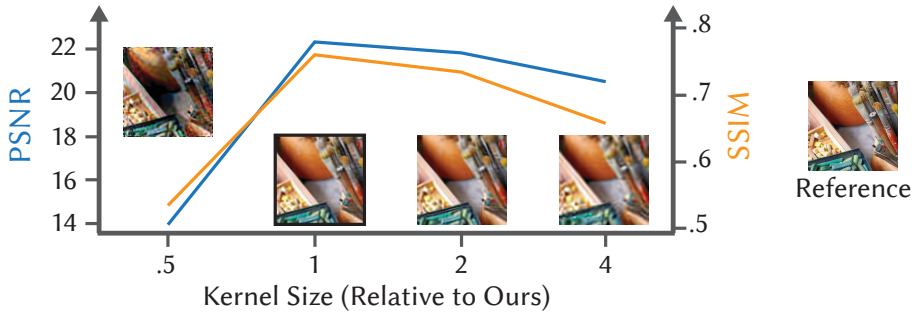

Figure 4.7: Antiderivative quality as a function of kernel size used in Eq. 4.9. We measure and display quality by repeated automatic differentiation of the learned antiderivatives. We see that our *minimal kernel* is optimal in terms of quality. Smaller kernels lead to instabilities and larger ones to blur.

$$\mathbb{E}_{\mathbf{x}\in\mathbb{R}^{d_{\text{in}}}}\left[\left\|\sum_{i=1}^{m}\hat{f}^n(\mathbf{x}-\mathbf{x}^{(i)})w^{(i)}-\mathbb{E}_{\boldsymbol{\tau}\in\text{supp}(h_n)}\left[f(\mathbf{x}-\boldsymbol{\tau})h_n(\boldsymbol{\tau})\right]\right\|\right], \tag{4.9}$$

where $\mathbf{x}^{(i)}$ and $w^{(i)}$ are the Dirac delta positions and magnitudes of h_n^{-n}. The first term is due to Equation (4.4) and the second term is a Monte Carlo estimate of the convolved signal.

For efficient training and a high-quality antiderivative, it is crucial for h_n to be very compact: First, the right part of Equation (4.9) becomes a tame interval, significantly reducing variance and thus enabling training with a low number of Monte Carlo samples (Fig. 4.6d). Second, it prevents $\hat{f}^n$ from "cheating" by not learning the antiderivative of the signal but the antiderivative of a convolved signal. However, we cannot reduce the support of h_n arbitrarily, as the left part of Equation (4.9) tends to result in instabilities, as the distances between the $\mathbf{x}^{(i)}$ shrink. Fig. 4.7 illustrates the inherent trade-off of this situation: There exists a sweet spot for the kernel size, producing the highest-quality antiderivatives. Smaller kernels lead to training instabilities, larger kernels to blur. We refer to the optimal solution as the *minimal kernel*.

While we could use any filter shape as minimal kernel h_n, we choose the n-fold convolution of a box with itself (Figure 4.8). This has the advantage that the left part of the loss Equation (4.9) becomes a sum over only $m = (n+1)^d$ elements that is efficient to compute, corresponding to higher-order finite differences.

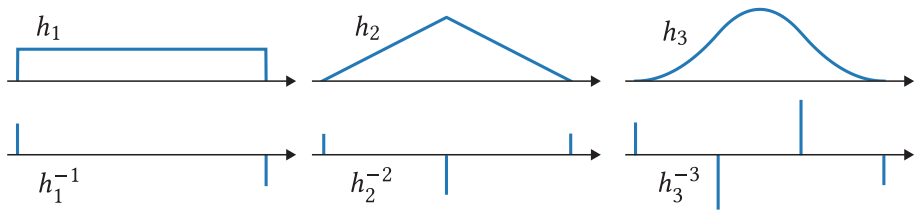

Figure 4.8: Minimal piecewise polynomial kernels of different degrees (*top row*) and their corresponding Dirac deltas (*bottom row*).

4.2.4 *Implementation Details*

We have implemented our prototype within the PyTorch (Paszke et al., 2017) environment. Our integral fields are realized using MLPs, where exact architectures vary slightly depending on the modality to be represented. In Tab. 4.2 we give details of network architectures for the repeated integral fields we use per application. In all cases we use a multi-layer perceptron (MLP). Similar to Lindell et al. (2021), we observed best results with Swish Ramachandran et al., 2017 activation functions. We report the number of hidden layers, the number of features per layer, and the resulting number of trainable parameters.

Table 4.2: Architecture details of our integral fields.

Application	#Layers	#Features	#Trainable Params.
Images (256x256)	5	256	270,851
Images (3000x3000)	5	512	1,065,987
Videos	9	256	534,019
Geometry	5	256	270,851
Audio	5	256	270,851
Animation	5	256	270,851

For training our repeated integral field, we again use the Adam (Kingma and Ba, 2015) optimizer with standard parameters. We handle boundaries by mirror-padding the training signal. Considering a unit domain, we train with a size of 0.025 for the minimal kernel h_n until convergence, followed by a fine-tuning on a kernel of size 0.0125. Empirically, we found this size to produce the highest-quality antiderivatives (Fig. 4.7). As the kernel size cannot be further decreased, our repeated integral field $\hat{f}^n$ is a slightly low-pass filtered version of f^n, resulting in a lower limit of filter sizes our convolutions can faithfully compute.

Table 4.3: Settings for all applications.

	Input			Kernel				
Modality	d_{in}	d_{out}	Format	Shape	SVa	d	Order	m b
Images	2	3	Grid	Gauss	✗	2	Linear	169
	2	3	Grid	DoG	✗	1	Linear	13
	2	3	Grid	Circle	✓	2	Const.	141
Bilat. Images	3	3	Grid	Gauss	✓	3	Const.	343
Video	3	3	Grid	Tent	✗	1	Linear	3
Geometry	3	1	SDF	Box	✓	3	Const.	8
Animation	1	69	Paths	Gauss	✗	1	Linear	13
Audio	1	1	Wave	Box	✗	1	Const.	2

Experiments with spatially-varying kernels.

The number of Diracs automatically adapts to the kernel as described in Sec. 4.2.2.1.

Fortunately, these small filters are highly amenable to efficient Monte Carlo estimation. Thus, at test time, whenever a kernel size falls below the threshold, we Monte-Carlo-estimate the convolution.

Convolution per Eq. 4.4 can be efficiently implemented as an augmented neural field based on $\hat{f}^n$, by prepending positional offsets $\mathbf{x}^{(i)}$ and appending a linear layer containing $w^{(i)}$.

4.3 APPLICATIONS

To demonstrate the generality and efficiency of our approach, we consider five signal modalities: images, videos, geometry, character animations, and audio. An overview of settings for all modalities is given in Tab. 4.3. References are computed using Monte Carlo estimation as per Eq. 2.2 until convergence. Further, we consider the following baselines:

INSP (XU ET AL., 2022) Similar to our approach, INSP relies on higher-order derivatives but uses them in a point-wise fashion to reason about local neighborhoods, reminiscent of a Taylor expansion. This method has to be trained for each convolution kernel separately, while our integral fields can be used with any kernel of a fixed polynomial degree. We follow their original implementation and provide all second-order partial derivatives. We found that providing more derivatives did not markedly improve their results.

BACON (LINDELL ET AL., 2022) AND PNF (YANG ET AL., 2022) While our method supports arbitrary filters at test time, both BACON and PNF are limited to a discrete cascade of k specific and fixed intermediate network outputs with different frequency contents, i. e., a set of pre-filtered versions of the signal. To compare ours to BACON and PNF, we approximate the convolution with an arbitrary filter as a linear combination of their intermediate outputs: Given the reference result, we optimize for a set of weights that when multiplied with corresponding intermediate outputs is closest to the target. Notice that this procedure requires a reference, while ours does not. For PNF, we observed that finer-scale intermediate outputs are not zero-mean. We compensate for this by subtracting the mean from all intermediate outputs and adding the sum of these means to the coarsest output. This way, finer levels only add higher-frequency details to the solution, but no global color shifts.

Figure 4.9: Gaussian 2D image blur of the input signal, with increasing bandwidth. It can be seen how our approach works also for large kernels.

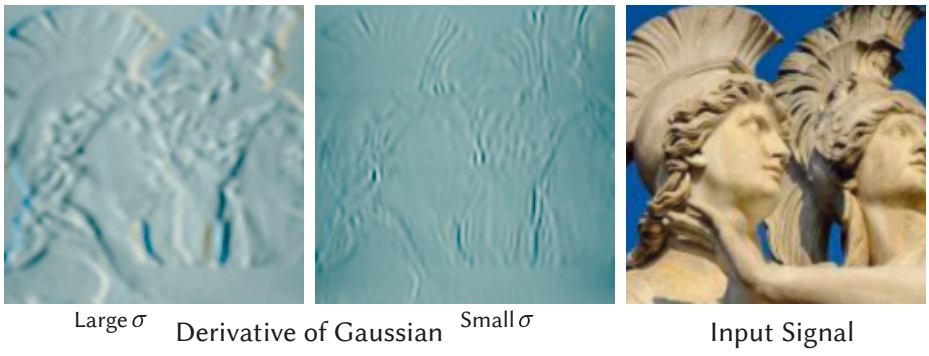

Figure 4.10: Derivative-of-Gaussian filtering 2D result. Note, that our approach supports such non-convex filters, producing signed results.

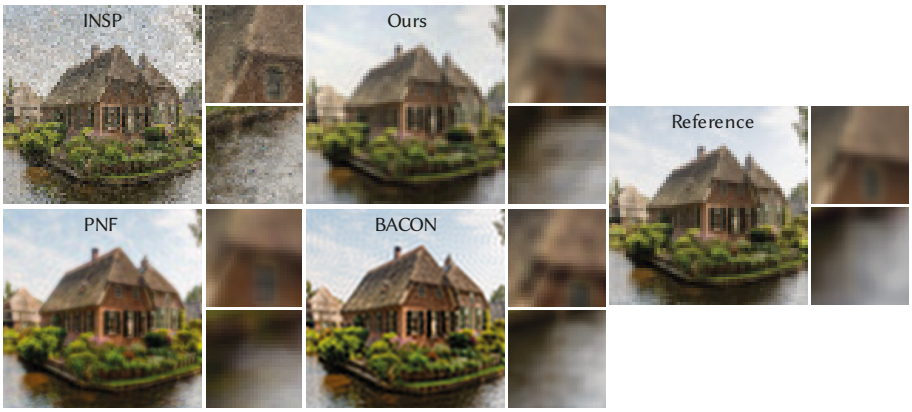

Figure 4.11: A qualitative comparison between INSP (Xu et al., 2022), BACON (Lindell et al., 2022), PNF (Yang et al., 2022), and our approach using a Gaussian kernel. INSP approach does hardly perform any filtering, when the kernel is larger, such as our approach supports, but suffers from noise. BACON fairs better, but shows ringing at high-contrast edges such as the house against the sky, while PNF also struggles to produce low-pass filtering results.

4.3.1 *Images*

In this application, we consider (high dynamic range) RGB images $f \in R^2 \to R^3$ as signals.

LINEAR FILTERING We show results for a Gaussian image filter in Figure 4.9 and a derivative-of-Gaussian filter in Figure 4.10. More results can be found in supplemental. Table 4.4 and Figure 4.11 evaluate image quality on a set of 50 images. Since no other method can produce competitive results for large kernels, to facilitate an insightful analysis, we limit our numerical evaluation to the small-kernel regime. We see that for very small kernels, BACON and PNF tend to produce higher-quality results, while we significantly outperform all methods as the kernel size increases.

NON-LINEAR FILTERING Our approach can also be used for non-linear filtering, such as bilateral filtering (Tomasi and Manduchi, 1998). We implement this by optimizing for the repeated integral field of a bilateral grid (Chen et al., 2007), which augments a 2D image with an additional signal dimension, reducing filtering to a linear operation in this extended space. We see in Fig. 4.12 that our approach is able to faithfully produce edge-aware filtering results.

Table 4.4: Image quality comparison for Gaussian kernels.

	$\sigma = 0.04$			$\sigma = 0.05$			$\sigma = 0.07$		
	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM	PSNR	LPIPS	SSIM
INSP	27.2	0.305	0.774	25.8	0.398	0.713	23.9	0.535	0.622
BACON	**35.2**	0.079	**0.946**	33.6	0.091	0.935	30.6	0.139	0.904
PNF	34.5	**0.072**	0.938	**33.9**	0.086	0.935	32.0	0.132	0.922
Ours	28.6	0.076	0.934	31.0	**0.042**	**0.963**	**34.1**	**0.032**	**0.98**

SPATIALLY-VARYING FILTERING is demonstrated in Fig. 4.1 and Fig. 4.13. In these cases, the strength of a circular blur filter is determined by a spatially-varying auxiliary signal in the form of a depth buffer.

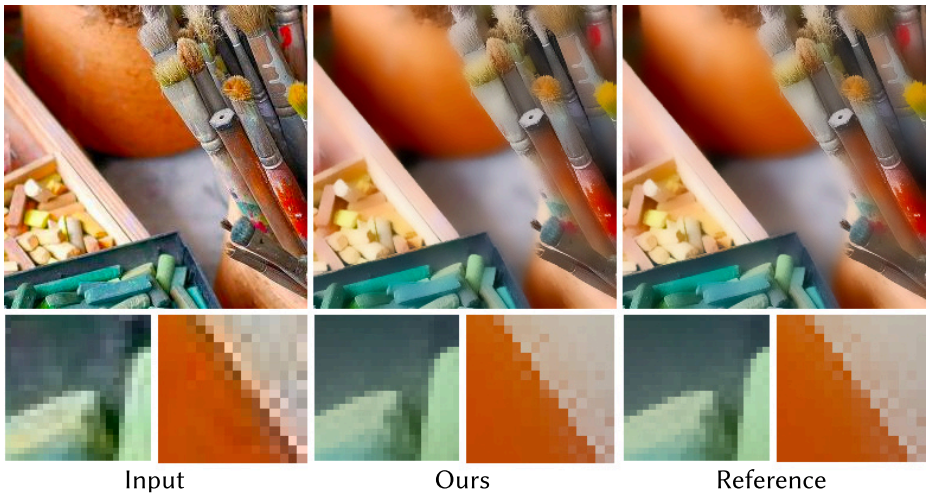

Figure 4.12: Non-linear (bilateral) filtering of an input image (left) by our method (middle) and the reference (right).

4.3.2 *Videos*

Videos are fields $f \in R^3 \to R^3$, mapping 2D location and time to RGB color. In Fig. 4.14, we apply smoothing with a tent filter along the time dimension to create appealing non-linear motion blur. We refer to our supplemental material for a more extensive evaluation.

Figure 4.13: Depth-of-field applied to a synthetic image with a depth map.

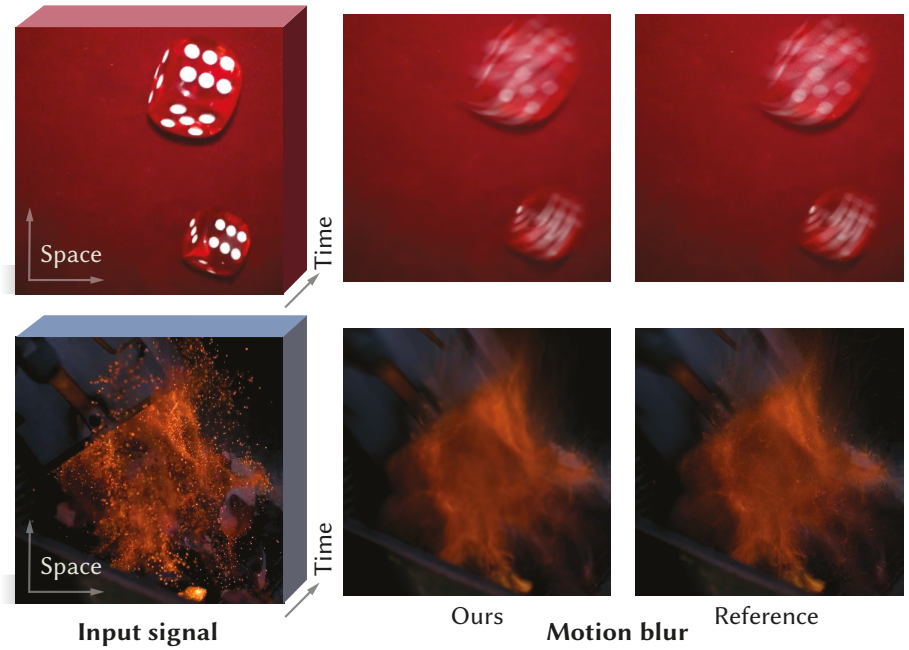

Figure 4.14: We apply our approach to a 3D space-time HDR field (video, seen left) to filter along the time axis with a tent kernel. The resulting motion blur (middle) compares favorably to the reference.

4.3.3 *3D Geometry*

Surfaces can be modeled using signed distance functions (SDFs) $f \in R^3 \to R$. We apply 3D box filters with different side lengths σ to an SDF, resulting in a progressively smoothed surface. We again compare our approach to INSP and BACON (no code is available to apply PNF to the 3D case) in Fig. 4.15 and Table 4.5, where numerical evaluations are averaged across the three 3D objects studied in INSP. As metrics, we

Table 4.5: Quality comparisons of filtered SDFs using 3D box kernels.

	$\sigma = 0.05$			$\sigma = 0.15$		
	MSE	Cham.	IoU	MSE	Cham.	IoU
INSP	292.50000	171.77	0.90	281.30000	388.26	0.73
BACON	0.40145	360.90	0.82	1.85791	932.84	0.61
Ours	**0.00109**	**20.97**	**0.99**	**0.00026**	**13.24**	**0.99**

consider the mean squared error (MSE) of the SDF, the intersection over union (IoU), as well as the Chamfer distance of the reconstructed surface. We observe that we outperform INSP and BACON across all kernel sizes and metrics. More qualitative results can be found in supplemental. Additionally, Fig. 4.16 demonstrates a spatially-varying SDF filtering result.

Figure 4.15: Two geometric shapes represented by an SDF (left) are filtered with a box kernel ($\sigma = 0.05$). While INSP, in the second column, suffers from noise, and BACON, in the third column, cannot reproduce larger-scale filtering, our result is close to the MC reference.

Figure 4.16: Spatially-varying blur from two views computed using our method.

4.3.4 *Animation*

We consider the task of filtering a neural field representation of a motion-capture sequence, which contains a significant amount of noise. Our test sequence consists of 23 3D joint position paths over time, resulting in a field $f \in R \to R^{69}$. In Fig. 4.17 and the supplemental, we show the result of applying a Gaussian filter to the noisy animation data, resulting in smooth motion trajectories.

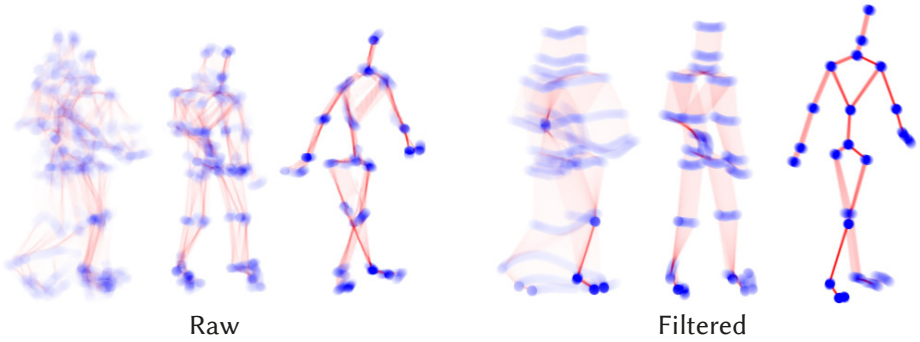

Figure 4.17: A noisy motion-capture sequence (left) is filtered using our approach to yield smooth motion trajectories (right).

4.3.5 *Audio*

Finally, we apply our framework to the task of filtering an audio signal $f \in R \to R$.

4.4 ANALYSIS

Here, we analyze further individual aspects of our approach.

REPEATED INTEGRAL FIELDS We seek to gain more insights into our learned integral fields. To this end, we consider the 2D image case (Sec. 4.3.1) both for single and double integrals per dimension, as required for convolutions with piecewise constant and linear kernels, respectively. In Tab. 4.6 we compare our antiderivatives against AutoInt (Lindell et al., 2021). As the original implementation does not support integration with respect to more than one variable, we re-implemented this baseline using the `functorch` library for repeated differentiation. We compute the mean squared error (MSE) between the original signal and the obtained integral fields after repeated automatic differentiation, averaged over three images. We do not compare integrals directly, as their absolute

values are dominated by higher-order constants of integration. Further, we measure the time required to train the fields, as well as the size of the network during training in terms of the number of nodes. As there is no straightforward procedure to count the number of nodes of the repeated-derivative graphs, we use the official AutoInt implementation for this particular calculation and differentiate two and four times with respect to *one* input variable, to get a good approximation of the two cases studied. Corresponding graphs are visualized in Fig. 4.18.

We see that our approach produces higher-quality antiderivatives than AutoInt while taking significantly less time to train, in particular for higher-order integrals. Further, our network size during training is independent of the integration order, while the computational graphs of AutoInt grow quickly due to the symbolic differentiation required.

We are further interested in how the quality of the learned antiderivative affects convolution quality. For this analysis, we consider antiderivative MSE as above (lower is better), and convolution quality in terms of PSNR (higher is better). We find the two measures highly correlated: Pearson's R = -0.98.

In Tab. 4.1 we study the accuracy of our optimized kernels.

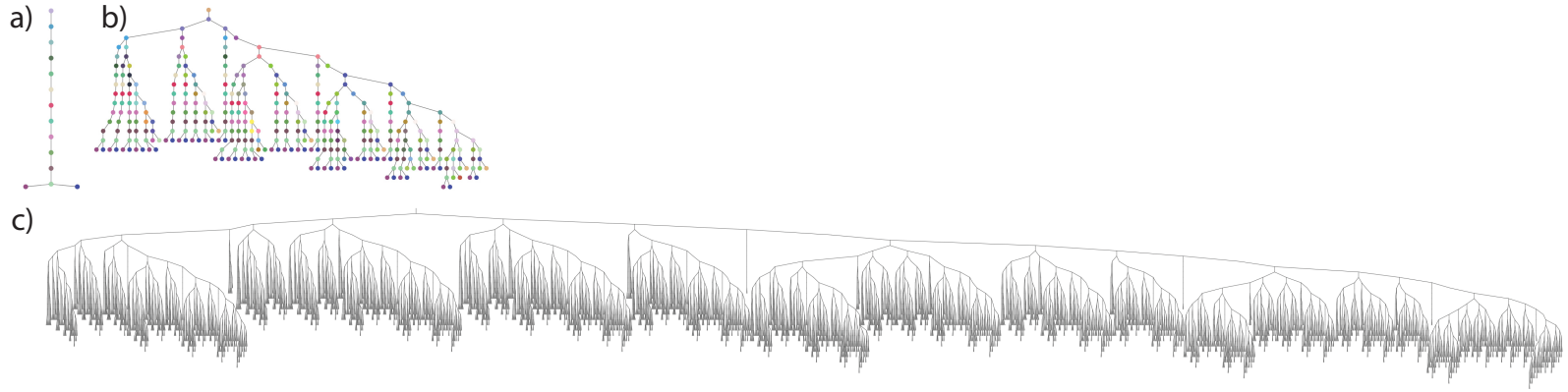

Figure 4.18: Comparison of different graph sizes. *a)* Our method retains a small graph independent of integration order. *b)* The AutoInt graph after two differentiations (first Int. Op. in Tab. 4.6). *c)* The AutoInt graph after four differentiations (second Int. Op. in Tab. 4.6). Same colors represent same operations.

Table 4.6: Integral field evaluation.

Int. Op.	Method	MSE ($\times 10^{-3}$)	Time	Graph Size
$\int\int$	AutoInt	6.83	1.3h	339
	Ours	**6.48**	**1.1h**	**11**
$\int^2\int^2$	AutoInt	7.71	14.7h	15,407
	Ours	**7.28**	**1.2h**	**11**

COMPARISON TO EQUAL-EFFORT MC While we use *converged* Monte Carlo estimates of convolutions (Eq. 2.2) as references in our experiments, we are also interested in the quality of such an estimate when reducing the number of samples to the number of Dirac deltas we use in our method, providing an equal-effort comparison. In Fig. 4.19 we show an illustration of this analysis. We observe that MC suffers from extensive noise, while our solution is smooth.

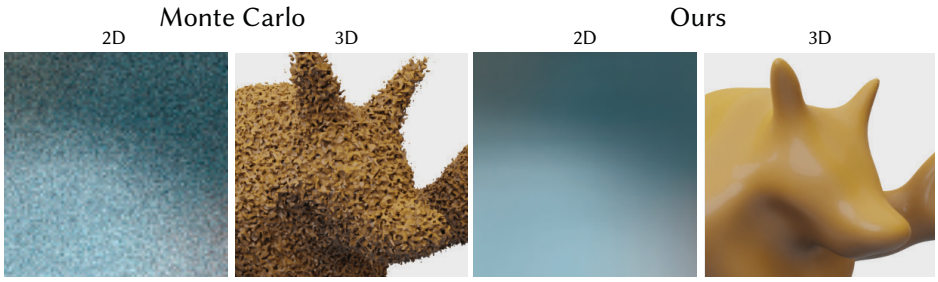

Figure 4.19: Ours vs. an equal-effort Monte Carlo estimate of a convolution.

4.5 DISCUSSION AND CONCLUSION

We have presented a novel approach to perform general, spatially-varying convolutions in continuous signals. Capitalizing on the fact that piecewise polynomial kernels become sparse after repeated differentiation, we only require a small number of integral-network evaluations to perform large-scale continuous convolutions.

Since our work is one of the first steps in this direction, there is ample opportunity for future work. Currently, one of our biggest limitations is that we need access to the entire signal to train our repeated integral field. This prevents the treatment of signals that are only partially observed through a differentiable forward map, e.g., as prominently is the case for neural radiance fields (Mildenhall et al., 2020).

Our integral fields are trained using generalized finite differences, which we found to become unstable for small kernels (Sec. 4.2.4). This naturally imposes an upper frequency limit on the learned antiderivatives. Fortunately, this becomes noticeable only for small kernels, in which case we resort to Monte Carlo sampling of the convolution, which is efficient in this condition.

Our assumption is that kernels to be used for spatially-varying filtering are (anisotropically) scaled versions of a reference kernel, which, arguably, covers a broad range of applications. We do not have a scalable solution for situations that violate this assumption – except for special problems such as bilateral filtering. One solution could be blending between the Dirac deltas of a set of pre-computed reference kernels. Further, kernel transformations are limited to axis-aligned operations. We envision that re-parameterizations leveraging the continuous nature of the representation might be able to lift this restriction.

In this work, we do not claim superiority over operating on a grid, if enough space is available to represent the signal that way. But if one needs to use a neural field, e.g. for extreme compression, continuous queries, or other advantages (Sec. 4.1), an approach such as ours is needed. We hope to inspire future work on signal processing in continuous neural representations to help them reach their full potential.

5

VOLUMETRIC INVERSE RENDERING VIA NEURAL RADIATIVE TRANSFER

Volumetric inverse rendering seeks to recover the optical properties of participating media from images. Existing approaches either rely on differentiable stochastic light transport simulation, which require substantial algorithmic effort, or use simplified models that fail to capture global illumination. We propose a formulation that reconciles physically complete light transport with general-purpose neural optimization. The optical properties of the medium and the full light field are represented as neural fields and estimated through a joint optimization process. Global illumination is enforced via a residual objective derived from the Radiative Transfer Equation in local differential form, complemented by a volume rendering term along primary viewing rays to mitigate spectral bias. We demonstrate reconstruction of spatially varying, color-resolved scattering, absorption, and phase function parameters from multi-view images. Beyond reconstruction, the same framework supports learning generative models of participating media with physical optical properties under global illumination.

5.1 INTRODUCTION

Light transport in volumetric scenes is complex. As light propagates through participating media, it is continuously scattered and absorbed, often undergoing multiple scattering events. These interactions generate long, intertwined transport paths that span the volume and tightly entangle illumination with the spatially varying optical properties of the medium. The radiance observed at any point is therefore not the result of a small number of local interactions, but the cumulative outcome of a dense superposition of indirect light contributions. Computational models of this process typically rely on radiative transfer simulations that use stochastic sampling of light paths (Pharr et al., 2023).

Here, we study the corresponding inverse problem: given image observations of a scene containing participating media, we aim to reconstruct its spatially varying optical properties. This challenging task is commonly formulated as an optimization problem, in which volumetric properties

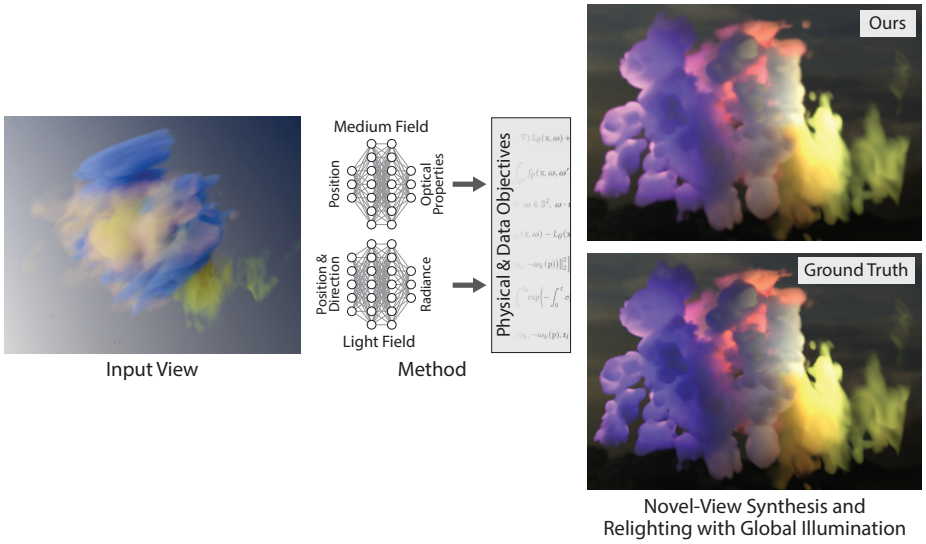

Figure 5.1: We propose a formulation for volumetric inverse rendering under global illumination without explicit global-illumination rendering. By framing the problem as constrained optimization over neural fields, the approach enables physically consistent reconstruction, novel-view synthesis, and relighting. Here, relighting of the reconstructed volume is demonstrated using a combination of environment illumination and three colored local light sources.

are estimated by minimizing a reconstruction objective using gradient-based methods. Existing approaches can be broadly divided into two categories. First, stochastic light path sampling is made differentiable (Li et al., 2018; Nimier-David et al., 2019), enabling the optimization to account for full radiative transfer (Fig. 5.2c). However, these methods demand substantial methodological and engineering effort to achieve acceptable efficiency (Nicolet et al., 2023; Nimier-David et al., 2020; Vicini et al., 2021) and to control bias in the estimated gradients (Nimier-David et al., 2022). In a second line of work, simplified models of light transport are employed, often in combination with neural scene representations (Mildenhall et al., 2020; Srinivasan et al., 2021; Zhang et al., 2023; Zheng et al., 2021) (Fig. 5.2a, b). While these approaches benefit from general-purpose, easy-to-use machine-learning frameworks, they typically fail to capture accurate optical properties under global illumination. To date, these two lines of work have remained largely disjoint.

In this work, we explore how to *reconcile the generality of physically complete light transport formulations with machine learning*. Our approach shifts complexity from transport-specific algorithm design to learned representations and optimization, thereby decoupling physical modeling from specialized numerical treatment.

Specifically, we represent both the optical properties of the participating media and the full light field of the scene using neural fields (Xie et al., 2022), and formulate their joint estimation as a continuous optimization problem (Fig. 5.1). Crucially, this optimization involves only minimal explicit simulation of light transport (Fig. 5.2d). Global illumination is incorporated through a residual objective derived from the Radiative Transfer Equation (RTE), following a physics-informed formulation based on local differential operators (Hadadan et al., 2021, 2023; Raissi et al., 2019). Although the supervision signals induced by the RTE are local, their propagation through optimization implicitly resolves the dense superposition of light paths, leading the joint estimate of the light field and the medium's optical properties to converge to a globally consistent light transport equilibrium without explicit transport modeling. The input observations and illumination conditions are incorporated through corresponding data constraints. In addition, we include a data term based on the Volume Rendering Equation (VRE), which integrates radiance along primary viewing rays only. This term mitigates the well-known spectral bias of physics-informed neural networks (Wang et al., 2022a) by coupling the reconstructed light field to view-ray integrals (we avoid the term "spectral bias", commonly used in the machine learning literature, to prevent confusion with the wavelength dependence of light). Importantly, explicit higher-order light transport simulation is avoided, since the RTE-based formulation accounts for global illumination. This substantially decouples inverse rendering research from renderer-specific algorithm design.

Using our formulation, we demonstrate the feasibility of reconstructing spatially varying, color-resolved scattering, absorption, and phase function parameters from multi-view observations under known illumination conditions, with competitive performance. Moreover, because the formulation relies on general-purpose neural optimization rather than transport-specific algorithmic machinery, it naturally extends beyond reconstruction to generative modeling of participating media with physically meaningful optical properties under full global illumination, learned directly from image observations. In summary, our contributions are:

- A novel formulation for volumetric inverse rendering that accounts for full global illumination with only minimal reliance on transport-specific simulation.
- A joint estimation framework for a scene's optical properties and full light field, both represented as neural fields and constrained by first principles of global light transport.

- Experimental demonstrations of optical property reconstruction and the application of the formulation to generative modeling of physically meaningful participating media.

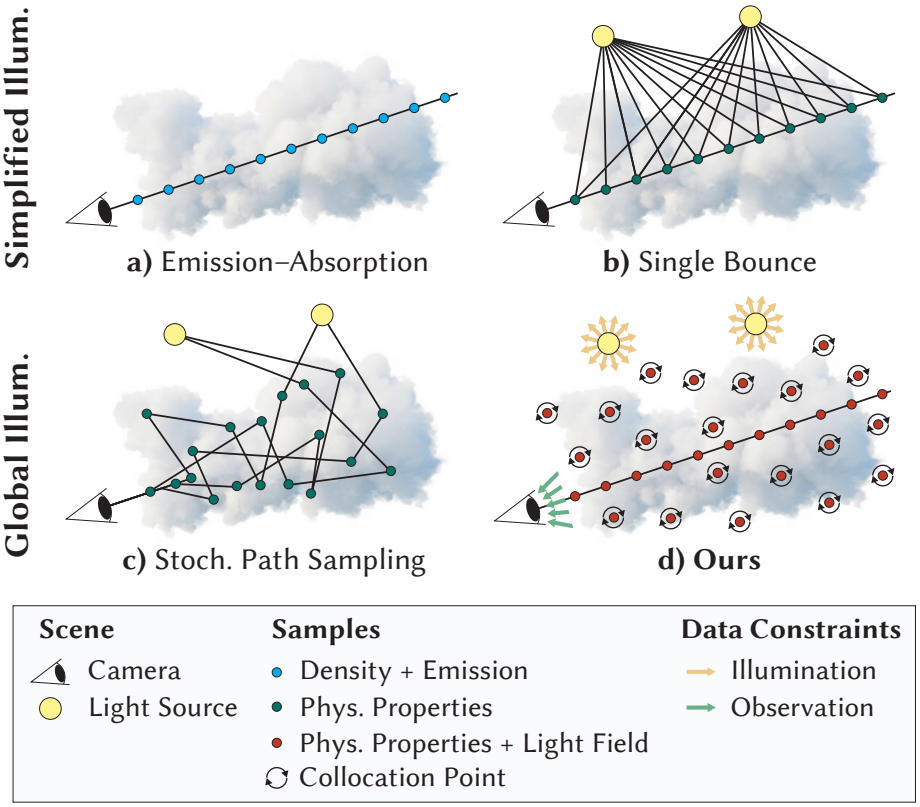

Figure 5.2: (Extension of (Fig. 3.2). Different approaches to volumetric inverse rendering. *(a)* The emission–absorption model bakes illumination into the volume, preventing recovery of physically meaningful optical properties. *(b)* Restricting light transport to a single bounce enables limited relighting but fails to capture the complexity of global illumination. *(c)* Differentiable stochastic path sampling accounts for global illumination during reconstruction, but requires substantial methodological and engineering effort. *(d)* Our approach enables global-illumination-aware reconstruction with only minimal explicit light transport modeling by jointly optimizing the medium's physical properties and the scene's light field in a neural representation. Illumination and observations are incorporated via data constraints on the light field within a physics-informed formulation, while local light transport constraints are enforced at continuously sampled collocation points. To capture high-frequency detail, an integral volume rendering formulation is applied along primary viewing rays only.

5.2 METHOD

Input to our method is a set of posed RGB images $\{I_k\}$ providing multi-view observations of a scene containing heterogeneous participating media. We assume a known and fixed environment illumination. Our goal is to reconstruct the optical properties of the participating media, namely spatially varying, color-resolved absorption and scattering coefficients $\sigma_a(\mathbf{x})$ and $\sigma_s(\mathbf{x})$, as well as spatially varying phase functions $f_p(\mathbf{x},\omega,\omega')$, under physically complete volumetric light transport.

While our formulation focuses on reconstructing the optical properties of a single scene, it naturally generalizes to a generative setting when observations from multiple scenes are available. Such a generative model can be used to populate virtual scenes with a diverse set of assets, such as clouds. It could also be employed in an inverse setting, where a latent code is optimized to explain an observation (a photo of a participating medium), thereby providing a strong prior on the distribution of plausible 3D volumes. In both settings – reconstruction and generation – the recovered physical scene representations enable physically meaningful analysis, novel-view synthesis, relighting under novel illumination, and other operations that rely on interpretable optical properties.

Physically accurate light transport in participating media involves complex, highly coupled interactions (Sec. 2.2) that are typically addressed using specialized simulation techniques. Here, we instead explore how scene properties can be estimated under these interactions through general-purpose neural optimization, with only minimal reliance on explicit transport simulation.

We first introduce our general framework for volumetric inverse rendering (Sec. 5.2.1). We then describe how this approach can be extended to the generative setting (Sec. 5.2.2), before providing implementation details (Sec. 5.2.3).

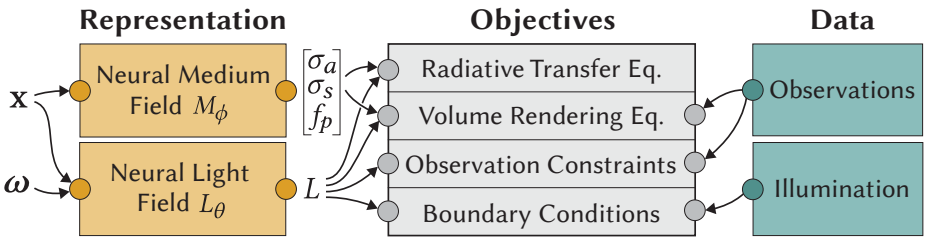

Figure 5.3: Overview of our approach. Two neural fields encode the optical properties of the participating medium and the scene's light field (orange). Disentanglement is guided by multiple optimization objectives (grey) that incorporate the available data (green).

5.2.1 *Neural Representation and Optimization*

Central to our approach is the representation of the light field and the optical properties of the participating medium as neural fields (Fig. 5.3, left). Specifically, we employ a neural field $L_\theta : (\mathbf{x},\boldsymbol{\omega}) \to \mathbb{R}^3$, which encodes the spatio-angular light field of the entire scene (Sitzmann et al., 2021), and a neural field $M_\phi : \mathbf{x} \to (\sigma_\mathrm{a}, \sigma_\mathrm{s}, g)$, which encodes the participating medium through spatially varying absorption and scattering coefficients and a scalar parameter g that defines a spatially varying Henyey–Greenstein (Henyey and Greenstein, 1941) phase function f_p. The subscripts θ and ϕ denote the optimizable parameters of the respective neural fields. To keep the notation concise, we omit explicit parameter subscripts for the medium properties in the remainder of this section and implicitly treat σ_a, σ_s, and f_p as outputs of M_ϕ.

Disentanglement of the light field and medium properties, as well as the incorporation of data, is achieved through physics-informed optimization objectives combined with data constraints (Fig. 5.3, right), rather than through specialized architectural mechanisms for separating illumination and material properties. Accordingly, the remainder of this section details the individual constraints that form the core of our approach. While our ultimate objective is the recovery of the output of M_ϕ, most optimization objectives are imposed on L_θ to facilitate disentanglement of illumination and medium.

The central objective enforcing physically consistent global light transport is the formulation of Eq. 2.3 as a residual:

$$\begin{aligned}\mathcal{L}_\mathrm{RTE}(\theta,\phi) =& \mathbb{E}_{\mathbf{x},\boldsymbol{\omega}}\Bigg[\Big\|(\boldsymbol{\omega}\cdot\nabla)\,L_\theta(\mathbf{x},\boldsymbol{\omega}) + \sigma_\mathrm{t}(\mathbf{x})\,L_\theta(\mathbf{x},\boldsymbol{\omega}) \\ &- \sigma_\mathrm{s}(\mathbf{x})\int_{\mathrm{S}^2} f_\mathrm{p}(\mathbf{x},\boldsymbol{\omega},\boldsymbol{\omega}')\,L_\theta(\mathbf{x},\boldsymbol{\omega}')\,\mathrm{d}\boldsymbol{\omega}'\Big\|_2^2\Bigg].\end{aligned} \tag{5.1}$$

Here, we omit the emission term, since we consider only environment illumination, and $\sigma_\mathrm{a}(\mathbf{x})$, $\sigma_\mathrm{s}(\mathbf{x})$, and $f_\mathrm{p}(\mathbf{x},\boldsymbol{\omega},\boldsymbol{\omega}')$ are obtained from the neural medium field M_ϕ. Note that this formulation enforces the radiative transfer constraints at continuously sampled collocation points $(\mathbf{x},\boldsymbol{\omega})$ within the spatio-angular domain (Fig. 5.2d) and is purely local in space. The first term in the expectation is evaluated using automatic differentiation. We approximate the in-scattering integral using Monte Carlo sampling over the angular domain, evaluated locally at each collocation point's spatial location $\mathbf{x}$, without tracing rays or explicitly tracking light through the volume.

To ensure a unique solution of the radiative transfer equation, the light field is additionally constrained by inflow boundary conditions at the volume boundary. Let $B \subset \mathbb{R}^3$ denote the spatial domain of the scene, with boundary ∂B and outward unit normal $\mathbf{n}_B(\mathbf{x})$. The inflow boundary Γ consists of all boundary position–direction pairs for which radiance enters the domain:

$$\Gamma = \left\{ (\mathbf{x}, \boldsymbol{\omega}) \;\middle|\; \mathbf{x} \in \partial B,\, \boldsymbol{\omega} \in \mathbb{S}^2,\, \boldsymbol{\omega} \cdot \mathbf{n}_B(\mathbf{x}) < 0 \right\}. \tag{5.2}$$

Boundary conditions are enforced through

$$\mathcal{L}_{\mathrm{BC}}(\theta) = \mathbb{E}_{(\mathbf{x},\boldsymbol{\omega})\in\Gamma}\left[\left\| L_{\mathrm{in}}(\mathbf{x},\boldsymbol{\omega}) - L_\theta(\mathbf{x},\boldsymbol{\omega}) \right\|_2^2 \right], \tag{5.3}$$

where L_{in} represents the incoming radiance at the boundary originating from environment illumination.

So far, we have only enforced data-agnostic physical and boundary constraints within the optimization. To incorporate the available image data, we introduce additional constraints on the light field derived from the observations. For an image I_k, each pixel $\mathbf{q} \in \mathbb{R}^2$ corresponds to a viewing direction $\boldsymbol{\omega}_k(\mathbf{q}) \in \mathbb{S}^2$ originating at the camera center $\mathbf{o}_k \in \mathbb{R}^3$. Under known camera calibration, the observed pixel value $I_k(\mathbf{q}) \in \mathbb{R}^3$ specifies the radiance arriving at the camera location from the corresponding viewing direction, i. e.traveling along direction $-\boldsymbol{\omega}_k(\mathbf{q})$ (green arrows in Fig. 5.2d). We therefore impose this as a pointwise constraint through

$$\mathcal{L}_{\mathrm{obs}}(\theta) = \mathbb{E}_{k,\mathbf{q}}\left[\left\| I_k(\mathbf{q}) - L_\theta\left(\mathbf{o}_k, -\boldsymbol{\omega}_k(\mathbf{q})\right) \right\|_2^2 \right], \tag{5.4}$$

where the expectation is taken over all observed views and pixels. This term constrains the light field only at the camera locations and for the set of observed incoming directions.

While the three objectives introduced so far are theoretically sufficient to guide the joint optimization of the light field L_θ and the material field M_ϕ, in practice we observe that the resulting solutions are overly smooth and lack high-frequency detail. This aligns with previous observations that neural network–based representations exhibit a spectral bias toward low-frequency solutions even under direct supervision (Rahaman et al., 2019), and that related optimization pathologies have been reported for physics-informed formulations relying on differential constraints (Wang et al., 2022a). We alleviate this effect by introducing an additional objective based on the VRE (Eq. 2.4), which explicitly integrates radiance along primary viewing rays (ray in Fig. 5.2d) and thereby introduces nonlocal

constraints that counteract the oversmoothing induced by purely local supervision:

$$\mathcal{L}_{\mathrm{VRE}}(\theta,\phi) = \mathbb{E}_{k,\mathbf{q}}\Bigg[\Bigg\| I_k(\mathbf{q}) - \int_0^{t_b} \exp\Bigg(-\int_0^t \sigma_{\mathrm{t}}(\mathbf{x}_s)\,\mathrm{d}s\Bigg) \Bigg(+\sigma_{\mathrm{s}}(\mathbf{x}_t)\int_{\mathbb{S}^2} f_{\mathrm{p}}(\mathbf{x}_t,-\boldsymbol{\omega}_k(\mathbf{q}),\boldsymbol{\omega}')\, L_\theta(\mathbf{x}_t,\boldsymbol{\omega}')\,\mathrm{d}\boldsymbol{\omega}'\Bigg)\mathrm{d}t\Bigg\|_2^2\Bigg], \tag{5.5}$$

where the locations $\mathbf{x}_\tau$ lie on the primary viewing ray $\mathbf{o}_k + \tau\,\boldsymbol{\omega}_k(\mathbf{q})$, while radiance is evaluated in the incoming direction $-\boldsymbol{\omega}_k(\mathbf{q})$.

Crucially, the in-scattering radiance in Eq. 5.5 is obtained by querying the neural light field L_θ, which is jointly optimized under all objectives, with global light transport consistency enforced by the RTE residual Eq. 5.1. As a result, although Eq. 5.5 aggregates radiance only along primary viewing rays, the radiance values integrated during volume rendering are not restricted to single-scattering or direct illumination, but instead evolve to encode the cumulative effect of arbitrarily many scattering interactions throughout the volume. Intuitively, the RTE governs the accumulation of indirect illumination throughout the volume, while the VRE term provides complementary non-local data coupling that anchors the directly observed component along each viewing direction.

In practice, the integrals in Eq. 5.5 are approximated using stochastic sampling. The line integral along each viewing ray is evaluated using jittered stratified sampling, following standard practice (Mildenhall et al., 2020). The directional in-scattering integral appearing in Eq. 5.5 is handled identically to the corresponding term in the RTE residual Eq. 5.1, using Monte Carlo sampling over the angular domain and evaluated locally at each spatial location.

Our complete optimization objective is given by

$$\begin{aligned}\mathcal{L}(\theta,\phi) = \mathcal{L}_{\mathrm{RTE}}(\theta,\phi) + \lambda_{\mathrm{BC}}\mathcal{L}_{\mathrm{BC}}(\theta) \\ + \lambda_{\mathrm{obs}}\mathcal{L}_{\mathrm{obs}}(\theta) + \lambda_{\mathrm{VRE}}\mathcal{L}_{\mathrm{VRE}}(\theta,\phi),\end{aligned} \tag{5.6}$$

where the scalar weights λ_{BC}, λ_{obs}, and λ_{VRE} balance the relative contributions of the individual objectives.

5.2.2 *Extension to Generative Modeling*

Beyond inverse rendering of individual scenes, our approach naturally extends to learning a generative model of volumetric scene properties from a distribution of observations across *multiple* scenes. Specifically, we assume access to an extended set of posed observations $\{I_{l,k}\}$, where l

indexes scenes and k indexes images within each scene, as before. In our proof-of-concept application, we assume the same illumination conditions across scenes.

We employ an auto-decoder framework (Park et al., 2019), where each scene is associated with an optimizable latent code $\mathbf{z}_l \in \mathbb{R}^{d_\mathbf{z}}$. This latent code is normalized to the unit hypersphere and provided as an additional input to both L_θ and M_ϕ, conditioning them on the corresponding scene. Training proceeds as in the scene-specific case described in Sec. 5.2.1, with only two changes. First, the scene-specific latent codes $\{\mathbf{z}_l\}$ are jointly optimized with the neural field parameters θ and ϕ, and are additionally regularized with an L_2 penalty Park et al., 2019. Second, the observation-related objectives $\mathcal{L}_{\text{obs}}$ (Eq. 5.4) and $\mathcal{L}_{\text{VRE}}$ (Eq. 5.5) are extended to account for the distribution over scenes. Eq. 5.4 now reads

$$\mathcal{L}_{\text{obs}}(\theta, \{\mathbf{z}_l\}) = \mathbb{E}_{l,k,\mathbf{q}}\Big[\big\| I_{l,k}(\mathbf{q}) - L_\theta\big(\mathbf{o}_k, -\boldsymbol{\omega}_k(\mathbf{q}), \mathbf{z}_l\big)\big\|_2^2\Big], \tag{5.7}$$

and Eq. 5.5 is modified analogously. The resulting model defines a generative distribution over volumetric scenes, jointly capturing light fields and material property fields, which can be sampled by randomly drawing a latent code $\mathbf{z}$ and evaluating the corresponding neural fields to obtain a complete scene.

It is worth noting that alternative inverse rendering formulations, including differentiable stochastic light transport, could in principle also be extended to learn distributions over medium properties, provided that the underlying scene representation supports such variability. However, doing so would require an independent light transport simulation for each sampled scene instance and for each training iteration, incurring the full cost of global illumination evaluation repeatedly during optimization. In contrast, our formulation jointly models a distribution over both medium properties and light fields, allowing the optimization to exploit shared structure across scenes and to amortize the resolution of global light transport through the learned light field representation. While the light field remains an auxiliary quantity, this joint modeling enables efficient learning of scene-level variability under physically complete volumetric illumination.

5.2.3 *Implementation Details*

Our implementation relies exclusively on general-purpose machine-learning components provided by PyTorch (Paszke et al., 2017), without requiring any custom operations or rendering-specific kernels. We minimize Eq. 5.6 using the Adam (Kingma and Ba, 2015) optimizer with default settings, and $\lambda_{\text{BC}} = \lambda_{\text{obs}} = 1$, and $\lambda_{\text{VRE}} = 70$.

As observed in prior work on both differentiable sampling-based and learning-based inverse rendering Nimier-David et al., 2022; Zhang et al., 2021c, scene initialization plays a crucial role in obtaining high-quality solutions. Accordingly, following common practice, we adopt a two-stage optimization procedure. In the first stage, we train an emission–absorption-based reconstruction model F_ξ Mildenhall et al., 2020 (Fig. 5.2a) using a neural hash grid architecture Müller et al., 2022. The color output of F_ξ is discarded, as it represents a non-physical emission term. In the second stage, we jointly optimize the neural medium field M_ϕ and the light field L_θ, both parameterized by SIREN networks Sitzmann et al., 2020. The medium field M_ϕ multiplicatively modulates the density output of F_ξ, which provides a strong high-frequency initialization for the volumetric structure. We enforce positivity of σ_s and σ_a via an exponential mapping, and bound g using a hyperbolic tangent. Experiments using an albedo–extinction parameterization yielded slightly inferior results. In the generative setting, latent codes $\mathbf{z}$ are incorporated differently, with F_ξ using multiplicative modulation of its hash grid features via an MLP-based encoding of $\mathbf{z}$ prior to decoding, and M_ϕ and L_θ concatenating the latent code with the positional and directional inputs. The described architectural design choices yielded the best results in a reasonably comprehensive pilot search.

In all experiments, we use 10k collocation points to evaluate Eq. 5.1, 32 directional samples to approximate the in-scattering integrals in Eq. 5.1 and Eq. 5.5, 10k boundary samples for Eq. 5.3, and 100 samples per viewing ray for evaluating Eq. 5.5. All samples are drawn randomly from the corresponding continuous domains at each optimization iteration. To avoid optimization gradient bias arising from stochastic sampling of in-scattering directions, we draw independent samples for the forward and backward passes Azinovic et al., 2019. The runtime of our method is almost entirely attributable to evaluating and optimizing the four objective terms $\mathcal{L}_{\text{RTE}}$ (38%), $\mathcal{L}_{\text{BC}}$ (7%), $\mathcal{L}_{\text{obs}}$ (2%), and $\mathcal{L}_{\text{VRE}}$ (53%).

Table 5.1: Quantitative evaluation of different methods (rows) across multiple evaluation protocols (columns) for scenes with *isotropic* phase functions. We report the accuracy of the recovered optical properties, the quality of novel-view synthesis under novel illumination, optimization time when using recommended hyperparameters on a single A100 GPU, and the generative quality under training and novel illumination. Dashes denote an evaluation that is not supported by a method.

Mode	Reconstruction							Generation	
Evaluation	Medium Properties			Novel-view Synthesis & Relighting			Time	Train. Illum.	Novel Illum.
	σ_a	σ_s	σ_t						
Metric	MSE$_{\times 10^2}\downarrow$	MSE$_{\times 10^2}\downarrow$	MSE$_{\times 10^2}\downarrow$	PSNR↑	SSIM↑	LPIPS↓	hours	FID↓	FID↓
Diff. Ratio Tracking	0.87	4.06	6.88	43.3	.993	.035	3.0	—	—
TensorIR	—	—	—	11.4	.594	.595	6.3	—	—
Ours	1.33	1.66	0.46	30.5	.989	.054	5.5	42.3	40.8

Table 5.2: Evaluation of our method for volume reconstruction on scenes with *anisotropic* phase functions using the MSE ($\times 10^2$).

	σ_a	σ_s	σ_t	g
Ours	1.99	2.37	1.33	1.21

5.3 EVALUATION

We first describe our evaluation setup, followed by an evaluation of our approach on volumetric scene reconstruction (Sec. 5.3.1) and generation (Sec. 5.3.2). We conclude with additional analysis (Sec. 5.3.3).

DATA Our work explores a novel formulation for modeling global illumination in volumetric inverse rendering, rather than aiming to develop a robust end-to-end pipeline for imperfect real-world data. To our knowledge, real multi-view datasets of volumetric scenes with known illumination do not yet exist. Accordingly, we evaluate our approach on synthetic scenes with known camera parameters and illumination conditions, with access to ground-truth volumetric properties for evaluation. To cover a broad range of volumetric configurations, we construct a custom dataset of 50 scenes. Each scene is assembled from a random set of volumetric elements drawn from the Disney Clouds dataset (Kallweit et al., 2017), which spans a continuum of structures from diffuse, low-frequency regions to sharply defined boundaries. For each selected element, we define RGB absorption and scattering coefficients, σ_a and σ_s, by scaling its base volumetric density with independently sampled random colors. This yields a high dynamic range of σ_a and σ_s (coefficients take values up to 5, corresponding to an expected 20 light–volume interactions per path) and produces diverse combinations of absorption and scattering, resulting in a wide range of optical behaviors spanning weakly absorbing, strongly scattering, and highly opaque elements. We additionally assign each element a Henyey-Greenstein phase function parameter g sampled uniformly from $[-0.5, 0.5]$. A scene is formed by randomly selecting between one and four augmented elements, applying independent random spatial shifts, rotations, and isotropic scalings, and combining them into a single volume. Illumination is provided by ten environment maps from polyhaven. For each scene, we randomly choose two different environment maps and render the scene under both conditions to enable evaluation of relighting. Observations are obtained by

volumetric path tracing from 500 camera viewpoints placed on a sphere around the volume and oriented toward its center; we reserve 50 views for testing. In addition to this multi-illumination dataset, we construct a single-illumination variant in which the same environment map is shared across all scenes for training and evaluating the generative setting. Finally, to enable direct comparison with baseline implementations that do not support anisotropic scattering, we also provide a variant with isotropic scattering in which all phase functions are fixed to $g = 0$. All datasets will be released upon publication.

BASELINES We consider two baseline approaches. First, we compare against Differential Ratio Tracking (Nimier-David et al., 2022), which accounts for global illumination by differentiating through stochastic light path sampling. enabling the optimization of a 3D grid of optical parameters. Like our approach, the method relies on an emission–absorption initialization. To ensure a meaningful comparison, we initialize both methods using our emission–absorption solution, which we found yields improved performance for Differential Ratio Tracking compared to its default setup. For quality evaluation, we perform an equal-time comparison by running DRT beyond its recommended iteration count until its total runtime matches that of our method. The resulting slightly improved DRT results are those reported. For runtime comparisons, however, we use DRT's recommended hyperparameters. DRT's publicly available implementation supports only isotropic phase functions for inverse rendering.

Our second baseline is an approach based on a learned representation. Since we are not aware of a learning-based method that publicly provides a complete implementation for volumetric inverse rendering under global illumination, we compare against TensorIR Jin et al., 2023 under fixed illumination as a representative baseline. TensorIR approximates second-order light transport using ray marching through an emissive volume and is not designed to recover physical volumetric optical properties.

5.3.1 *Reconstruction*

We evaluate all methods on the large-scale dataset with isotropic scattering using two evaluation protocols. First, we compute the mean squared error (MSE) of the reconstructed absorption, scattering, and extinction coefficients, as well as the phase function parameters, by densely sampling the corresponding quantities throughout the volume; the error of g is weighted by the local scattering coefficient σ_s prior to aggregation. Second, we evaluate novel-view synthesis on test views under a novel illumination condition. Images for Differential Ratio Tracking and our

approach are synthesized by re-rendering the reconstructed volumes using volumetric path tracing. We evaluate image quality using PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018) on logarithmically tonemapped images to account for high dynamic range.

Quantitative results are listed in Tab. 5.1, left, while qualitative comparisons are shown in Fig. 5.4. We observe that our method achieves significantly more accurate recovery of the scattering and extinction coefficients than differentiable stochastic path sampling. At the same time, image-level novel-view synthesis quality is highest for the stochastic path-sampling approach, reflecting the benefit of explicit Monte Carlo simulation for optimizing pixel-level appearance. The learning-based baseline did not produce competitive results on our challenging dataset, despite extensive efforts to ensure a fair evaluation. Specifically, we tested configurations using LDR input observations, fixed the environment illumination to the ground truth, and explored a range of training hyperparameters. As we were able to reproduce the authors' reported results on the datasets used in their evaluation, we hypothesize that the method may not be well suited to the intricate volumetric structures present in our dataset.

A quantitative evaluation of reconstruction quality for our method with anisotropic scattering is provided in Tab. 5.2. We observe that incorporating anisotropic scattering as an additional degree of freedom leads to only a moderate reduction in reconstruction accuracy compared to the isotropic case, while none of the baseline implementations support recovery of anisotropic scattering coefficients. Fig. 5.1 further demonstrates the accuracy of our recovered scene properties using complex relighting including strong local light sources.

5.3.2 *Generation*

To evaluate the generative capabilities of our approach, we train a generative model using our large-scale dataset with a single shared illumination condition and a latent dimensionality of $d_{\mathbf{z}} = 4$. We then synthesize 30 scenes by sampling latent codes $\mathbf{z}$ and render the corresponding scenes under the training illumination. To assess the faithfulness of the learned distribution over optical properties, we additionally render the same generated scenes under randomly sampled novel illuminations. In both cases, the distribution of the resulting renderings is compared to that of the corresponding real images – either from the training set or from the multi-illumination dataset – using the FID score (Heusel et al., 2017). For both evaluations, we extract random crops from each image to improve the robustness of the distributional comparison. The corresponding scores

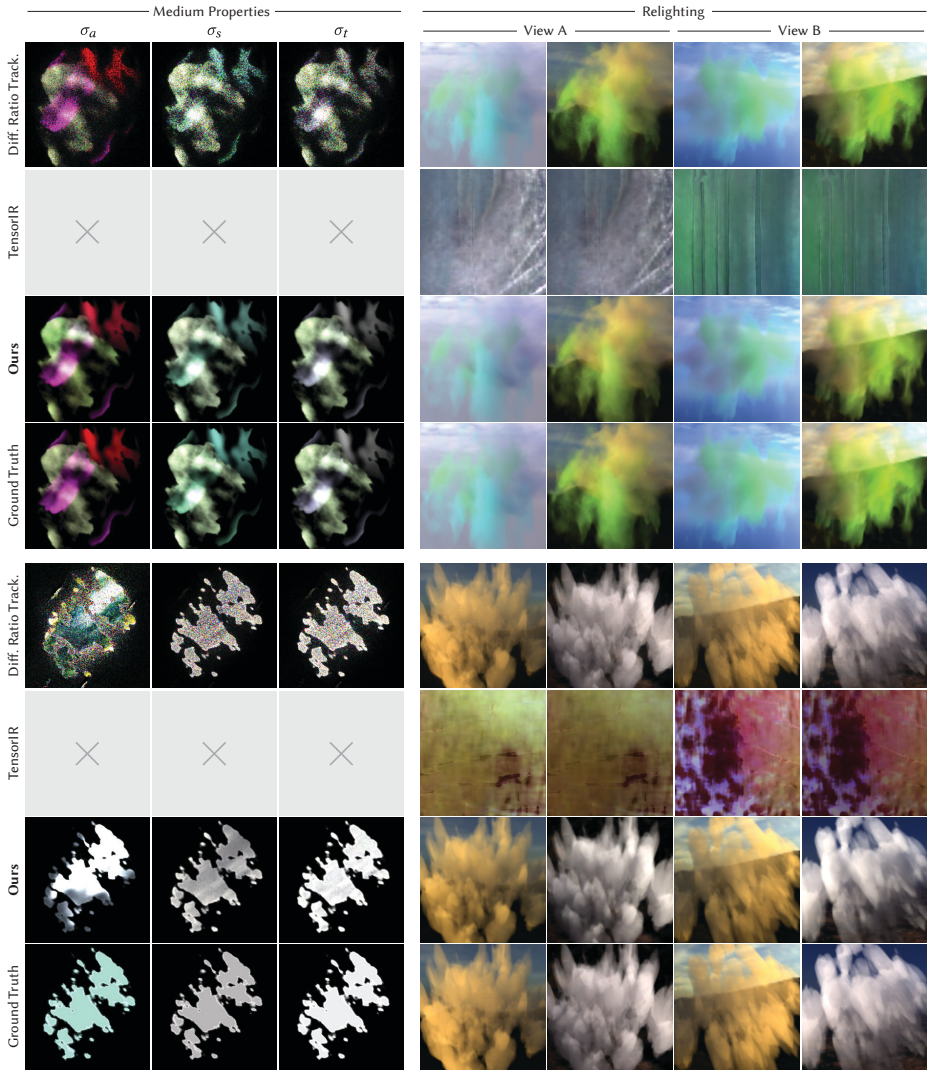

Figure 5.4: Reconstruction results on two scenes (row blocks), comparing different methods (rows). The first three columns show linearly tonemapped slices through the reconstructed medium properties, namely absorption (σ_a), scattering (σ_s), and extinction (σ_t). The remaining columns demonstrate novel-view synthesis under novel illumination. TensorIR does not recover volumetric medium properties and is therefore shown only for the image-based comparisons.

are reported in Tab. 5.1, right, while qualitative samples are shown in Fig. 5.5. Our results show that the model learns a generative distribution over physical volumetric optical properties – rather than appearance – yielding scene samples that support consistent global illumination and relighting, a capability that has not been demonstrated by prior volumetric generative approaches.

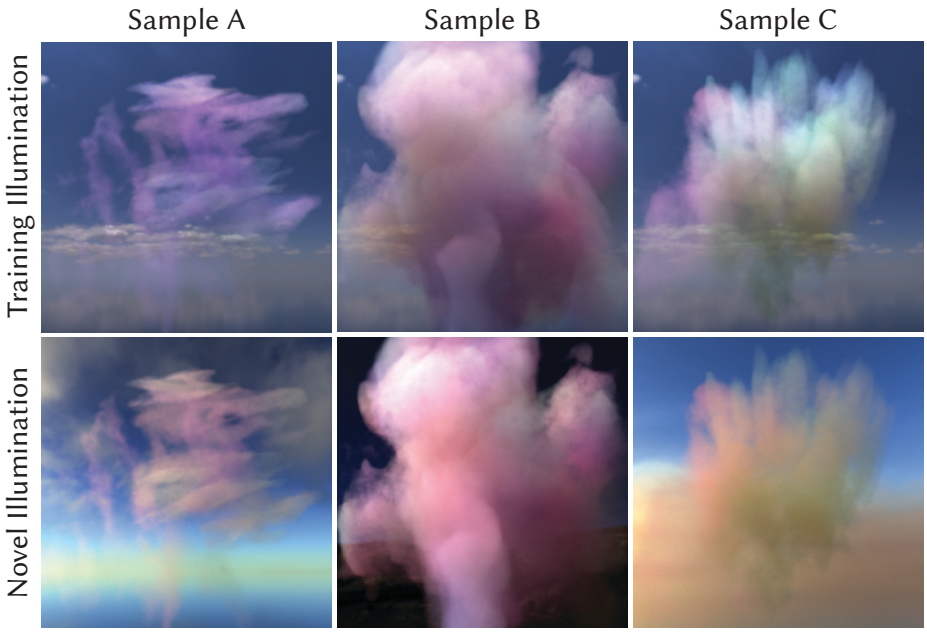

Figure 5.5: Three scenes (columns) sampled from our generative model, showing diverse volumetric structures with physically meaningful optical properties, rendered under training (top row) and novel (bottom row) illumination. All images are rendered from to the same view.

5.3.3 *Ablation*

Our formulation is intentionally minimal, relying on a small set of physically motivated objectives (Sec. 5.3.3.1) and two generic neural fields for scene representation (Sec. 5.3.3.2), with solutions obtained via optimization (Sec. 5.3.3.3).

5.3.3.1 *Objectives*

Most of our optimization objectives are essential: removing any of the RTE residual, boundary conditions, or observation constraints renders the problem underconstrained and prevents convergence to a meaningful solution. Specifically, this leads to physically inconsistent solutions, neglect of illumination, and disregard of observations, respectively. We therefore focus our ablation study on the only component of our formulation that appears optional, namely the VRE-based objective (Eq. 5.5). As shown in Fig. 5.6, omitting the VRE term leads to overly smooth reconstructions that lack high-frequency detail. This behavior is consistent with the known low-frequency bias of physics-informed neural networks relying solely on local differential constraints. Importantly, this effect is largely

orthogonal to architectural choices such as positional encoding Mildenhall et al., 2020; Tancik et al., 2020, which only provide the *capacity* to represent high-frequency functions but still require sufficiently direct supervision for this capacity to be utilized. Incorporating the nonlocal VRE objective effectively provides such supervision and yields substantially sharper and more faithful reconstructions.

5.3.3.2 *Representation*

We use neural fields to represent both the light field and the volumetric properties. For reconstruction, the former is inherently a 5D function and thus naturally represented using a neural field, whereas the latter is 3D and therefore, in principle, amenable to straightforward discretization. However, as shown in Tab. 5.3, a grid-based representation yields inferior results compared to our neural representation in this setting. Moreover, grid-based representations are fundamentally incompatible with our generative modeling formulation.

5.3.3.3 *Optimization*

Our optimization proceeds in two stages. First, an emission–absorption model is obtained, whose density is subsequently modulated by a function optimized in the second stage to yield the final medium properties. As shown in Tab. 5.3, a direct, single-stage optimization yields inferior results compared to our two-stage approach.

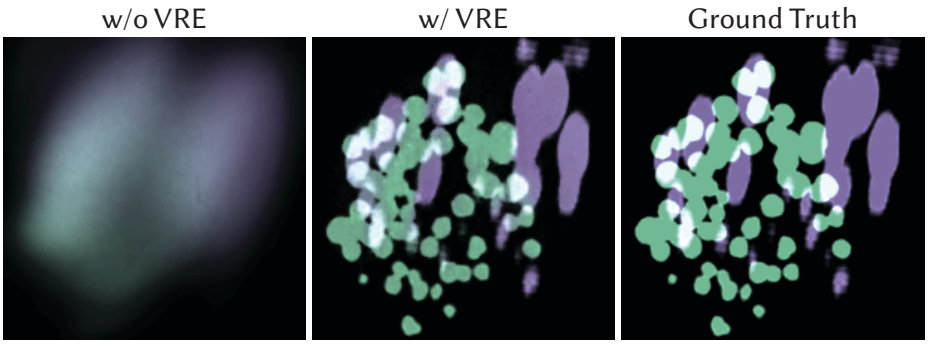

Figure 5.6: Ablation. Reconstruction of scattering parameters σ_s (shown here as representative of all recovered volumetric parameters) on a slice through the volume. Omitting the VRE objective (Eq. 5.5) leads to overly smooth reconstructions, whereas incorporating it yields results closer to the ground truth.

Table 5.3: Quantitative results of ablation studies, measured as MSE ($\times 10^2$) of reconstructed medium properties.

	σ_a	σ_s	σ_t
Grid Representation	1.98	5.20	3.43
Single-stage Optimization	1.68	2.94	2.60
Ours	1.33	1.66	0.46

5.4 CONCLUSION

We have demonstrated that, for global-illumination-aware inverse rendering, a global-illumination renderer is not essential. By representing both the optical properties of participating media and the full light field as neural fields, and by constraining them through predominantly local formulations of radiative transfer, our optimization-based approach enables the reconstruction of high-quality, physically meaningful volumetric parameters. Beyond inverse rendering, the proposed formulation naturally supports generative modeling of participating media with physical optical properties, learned directly from image observations.

Because both the scene representation and the optimization objective are expressed in terms of principled, physics-based constraints, the framework is agnostic to the dimensionality of the underlying signal and compositional with respect to constraints. This opens promising directions for extending the approach to spectral and transient light transport, as well as dynamic scenes, while preserving physical consistency. Additionally, while our experiments focus on environment illumination, the proposed framework can be extended to other lighting representations, such as local light sources, by incorporating the emission term into the radiative transfer equation. Further, exploring the application of Eq. 5.5 to estimate incident illumination within the volume could lead to hybrid approaches that combine our neural formulation with explicit simulation. We believe this perspective provides a flexible foundation for future work on volumetric scene understanding and synthesis.

6 CONCLUSION

6.1 SUMMARY AND DISCUSSION

This thesis has shown how aggregation operators can be realized directly on neural fields. The key observation is that such operators, although global in nature, can in some cases be reformulated as local differential constraints. This observation underlies the two main contributions of the thesis.

Our first contribution, presented in Chapter 4, is a method for performing continuous convolutions directly on neural fields. The method builds on the convolution identity that a convolution between a signal and a kernel can equivalently be expressed as the convolution between a repeated antiderivative of the signal and a repeated derivative of the kernel, and vice versa. Combined with the observation that piecewise polynomial kernels become sparse under repeated differentiation, this allows the original convolution to be reformulated in terms of a sparse differential kernel and a repeated integral representation of the signal. To make this practical in the neural setting, we introduced a repeated integral field that, together with the sparse kernel representation, enables the convolution to be evaluated using only a small number of neural field queries, independent of kernel size.

At the same time, this formulation has clear limitations. Most importantly, training the repeated integral field requires access to the full signal, which prevents its immediate application to settings in which the signal is observed only indirectly through a differentiable forward model, such as in neural radiance fields. In addition, the generalized finite-difference training becomes unstable for very small kernels, imposing an upper frequency limit on the learned antiderivatives. Below this regime, the method reverts to Monte Carlo sampling. Spatially varying filtering is also currently restricted to kernels that arise as anisotropically scaled versions of a reference kernel, leaving a fully general treatment open.

Our second contribution, presented in Chapter 5, addresses volumetric inverse rendering under full global illumination. The proposed method represents both the optical properties of the participating medium and the transported light field as neural fields, and enforces global illumination through a differential formulation of the radiative transfer equation together with image-based supervision. This makes it possible to recover

spatially varying, physically meaningful volumetric parameters from multi-view images without explicit global illumination simulation during optimization. The same formulation also supports the generative modeling of participating media, extending its scope from inverse estimation to the synthesis of volumetric scenes with physically plausible optical properties.

This formulation likewise has important limitations. The ablation study shows that local differential constraints alone are not sufficient to recover high-frequency structure reliably. Omitting the VRE-based objective leads to reconstructions that are overly smooth and lack detail. This indicates that, while local transport constraints are essential, they must in practice be complemented by suitably chosen nonlocal objectives when the target solution contains substantial high-frequency content.

Although the two contributions address different technical problems, they share the same underlying idea: the use of differential constraints to handle operations that are global in nature. In both cases, the starting point is an operator defined through integration over a continuous domain, which is treated not in its original form but through local differential constraints. In Chapter 4, such a differential constraint is used to train the repeated integral field required for continuous convolution. In Chapter 5, it is used to enforce full global illumination through a differential form of the radiative transfer equation.

The focus on differential constraints also leads to a broader observation about neural fields themselves. In this thesis, neural fields are used not only as representations of continuous signals, but also as computational representations on which operators and physical constraints can be imposed directly. This points to a broader perspective on neural fields: they should be understood not only as a means of storing continuous signals, but also as a substrate for structured continuous computation. More broadly, this thesis suggests that progress in neural fields should be assessed not only in terms of representational quality, but also in terms of the extent to which such representations support structured, efficient, and physically grounded computation.

Recent developments further reinforce this perspective. Neural Gaussian Scale-Space Fields (Mujkanovic et al., 2024) shows that fully continuous anisotropic Gaussian scale spaces can be learned directly from signals using appropriately constrained neural network architectures. In a closely related direction, Spectral Prefiltering of Neural Fields (Yaldiz et al., 2025) demonstrates that filtering can also be performed analytically and efficiently in the spectral domain, enabling prefiltered neural fields corresponding to a broader variety of kernels to be obtained in a single forward pass. Together, these works support the idea that filtering

and multiscale analysis can be carried out natively on continuous representations. More broadly, Neural Fields as Distributions (Rebain et al., 2024) places neural fields within a broader signal-processing framework beyond Euclidean domains, suggesting that the perspective developed in Chapter 4 may extend beyond standard spatial settings. Although these approaches are methodologically distinct from the repeated-integration-based formulation proposed in Chapter 4, they are aligned with the same broader objective: to make filtering and related signal-processing operations native to neural fields.

In terms of scene representation, the field has been shifting toward more explicit volumetric representations. In particular, 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) established anisotropic Gaussian primitives as an efficient scene representation capable of real-time rendering and has emerged as a major alternative to purely neural representations. Subsequent work suggests that such representations may also be well suited to inverse rendering, including the recovery of optical properties. (Condor et al., 2025) show that such primitive-based representations can be extended to scattering and emissive media in both forward and inverse settings. Similarly, Gabor Fields (Condor et al., 2026) shows that explicit volumetric representations can incorporate continuous frequency filtering and level-of-detail control directly into the representation, with corresponding benefits for rendering efficiency. These developments raise broader questions about the advantages and trade-offs of different representations and their suitability for the task at hand. In this respect, they reflect a broader trend: volumetric representations are increasingly being designed not only to store scene information, but also to support rendering more directly through the representation itself.

6.2 FUTURE WORK

Several directions for future work follow naturally from the results of this thesis. For neural field convolutions (Chapter 4), an important next step is to relax the requirement that the full signal be available during training of the repeated integral field. A formulation that operates under indirect or partial supervision would substantially broaden the applicability of the method, particularly to inverse problems in which the signal is observed only through a rendering or forward operator. One possible direction in such settings is to first reconstruct the signal itself using an existing method and then use the trained representation as supervision for learning the repeated integral field within our formulation. In the case of neural radiance fields, for example, this would amount to first training the NeRF and then using the resulting representation to super-

vise the corresponding neural integral field. Such a strategy could extend the method to signals that are not directly available, although it would introduce the additional requirement that the underlying signal representation be trained beforehand. A second direction concerns the treatment of spatially varying kernels. The current framework already supports a practically relevant class of transformed kernels, but a more general treatment of continuously varying kernel families would strengthen the claim that convolution can be performed natively and flexibly on neural field representations.

For volumetric inverse rendering (Chapter 5), the proposed formulation likewise opens natural directions for extension. Because both the scene representation and the optimization objectives are expressed through principled, physics-based constraints, the framework is not restricted to the precise transport regime considered in this thesis. Spectral and transient light transport, as well as dynamic participating media, constitute especially promising directions. Another natural direction is to relax the transport model itself. The current formulation assumes classical exponential attenuation. Recent work has shown that volumetric rendering can be extended to non-exponential transmittance regimes, albeit thus far only in the emission–absorption setting (Speierer and Jarabo, 2026). Extending the present formulation in this direction would enable the reconstruction of a broader range of materials exhibiting non-exponential attenuation. A further limitation is the lack of support for anisotropic material structure. The current model does not account for oriented scattering structures such as fur, hair, or woven fabrics, for which the standard radiative transfer setting is no longer sufficient (Jakob et al., 2010). Incorporating radiative transfer models for anisotropic media would therefore broaden the class of materials that can be reconstructed. In addition, the present formulation assumes straight light paths and does not account for continuous refraction within the medium. Extending it toward the refractive radiative transfer equation (Ament et al., 2014), in which rays are no longer assumed to propagate along straight lines, would substantially enlarge the range of optical effects that can be modeled. Finally, the generative extension introduced in Chapter 5 suggests that combining neural fields with physical constraints may provide a useful basis not only for reconstruction, but also for physically meaningful scene generation under richer transport settings.

Beyond these extensions, the thesis points toward a broader methodological direction. The use of local differential formulations in Chapters 4 and 5 suggests that related strategies may be applicable to other continuous operators and inverse problems as well. This includes not only

additional transforms, but also potentially broader classes of physically constrained simulation and generative modeling.

BIBLIOGRAPHY

Abdoulaev, Gassan S, Kui Ren, and Andreas H Hielscher (2005). "Optical tomography as a PDE-constrained optimization problem." In: *Inverse Problems*.

Adams, Marvin L and Edward W Larsen (2002). "Fast iterative methods for discrete-ordinates particle transport calculations." In: *Progress in nuclear energy*.

Ahlberg, J Harold, Edwin Norman Nilson, and Joseph Leonard Walsh (2016). *The Theory of Splines and Their Applications*. Elsevier.

Ament, Marco, Christoph Bergmann, and Daniel Weiskopf (2014). "Refractive radiative transfer equation." In: *ACM Transactions on Graphics (TOG)*.

Azinovic, Dejan, Tzu-Mao Li, Anton Kaplanyan, and Matthias Nießner (2019). "Inverse path tracing for joint material and lighting estimation." In: *CVPR*.

Bangerth, Wolfgang and Amit Joshi (2008). "Adaptive finite element methods for the solution of inverse problems in optical tomography." In: *Inverse Problems*.

Barron, Jonathan T., Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan (2021). "Mip-NeRF: A Multiscale Representation for Anti-Aliasing Neural Radiance Fields." In: *ICCV*.

Barron, Jonathan T., Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman (2022). "Mip-NeRF 360: Unbounded Anti-Aliased Neural Radiance Fields." In: *CVPR*.

Bemana, Mojtaba, Karol Myszkowski, Hans-Peter Seidel, and Tobias Ritschel (Nov. 2020). "X-Fields: implicit neural view-, light- and time-image interpolation." In: *ACM Transactions on Graphics (TOG)*.

Bi, Sai, Zexiang Xu, Pratul Srinivasan, Ben Mildenhall, Kalyan Sunkavalli, Miloš Hašan, Yannick Hold-Geoffroy, David Kriegman, and Ravi Ramamoorthi (2020). "Neural reflectance fields for appearance acquisition." In: *arXiv preprint arXiv:2008.03824*.

Boss, Mark, Varun Jampani, Raphael Braun, Ce Liu, Jonathan Barron, and Hendrik Lensch (2021). "Neural-pil: Neural pre-integrated lighting for reflectance decomposition." In: *NeurIPS*.

Brigham, E Oran (1988). *The fast Fourier transform and its applications*. Prentice-Hall, Inc.

Chan, Eric R, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. (2022). "Efficient geometry-aware 3d generative adversarial networks." In: *CVPR*.

Chandrasekhar, Subrahmanyan (1960). *Radiative Transfer*. Courier Corporation.

Che, Chengqian, Fujun Luan, Shuang Zhao, Kavita Bala, and Ioannis Gkioulekas (2020). "Towards learning-based inverse subsurface scattering." In: *ICCP*. IEEE.

Chen, Hansheng, Jiatao Gu, Anpei Chen, Wei Tian, Zhuowen Tu, Lingjie Liu, and Hao Su (2023a). "Single-stage diffusion nerf: A unified approach to 3d generation and reconstruction." In: *ICCV*.

Chen, Hongze, Zehong Lin, and Jun Zhang (2025). "GI-GS: Global Illumination Decomposition on Gaussian Splatting for Inverse Rendering." In: *ICLR*.

Chen, Jiawen, Sylvain Paris, and Frédo Durand (2007). "Real-time edge-aware image processing with the bilateral grid." In: *ACM Transactions on Graphics (TOG)*.

Chen, Rui, Yongwei Chen, Ningxin Jiao, and Kui Jia (2023b). "Fantasia3D: Disentangling Geometry and Appearance for High-quality Text-to-3D Content Creation." In: *ICCV*.

Chu, Mengyu, Lingjie Liu, Quan Zheng, Aleksandra Franz, Hans-Peter Seidel, Christian Theobalt, and Rhaleb Zayer (2022). "Physics informed neural fields for smoke reconstruction with sparse data." In: *ACM Transactions on Graphics (TOG)*.

Condor, Jorge, Nicolai Hermann, Mehmet Ata Yurtsever, and Piotr Didyk (2026). "Gabor Fields: Orientation-Selective Level-of-Detail for Volume Rendering." In: *arXiv preprint arXiv:2602.05081*.

Condor, Jorge, 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." In: *ACM Transactions on Graphics (TOG)*.

Cook, Robert L, Thomas Porter, and Loren Carpenter (1984). "Distributed ray tracing." In: *ACM SIGGRAPH*.

Crow, Franklin C (1984). "Summed-area tables for texture mapping." In: *ACM SIGGRAPH*.

Deng, Xi, Fujun Luan, Bruce Walter, Kavita Bala, and Steve Marschner (2022). "Reconstructing translucent objects using differentiable rendering." In: *ACM SIGGRAPH*.

Deng, Xi, Lifan Wu, Bruce Walter, Ravi Ramamoorthi, Eugene d'Eon, Steve Marschner, and Andrea Weidlich (2024). "Reconstructing translucent thin objects from photos." In: *ACM SIGGRAPH*.

Dihlmann, Jan-Niklas, Arjun Majumdar, Andreas Engelhardt, Raphael Braun, and Hendrik Lensch (2024). "Subsurface Scattering for Gaussian Splatting." In: *NeurIPS*.

Diolatzis, Stavros, Jan Novak, Fabrice Rousselle, Jonathan Granskog, Miika Aittala, Ravi Ramamoorthi, and George Drettakis (2023). "Mesogan: Generative neural reflectance shells." In: *Computer Graphics Forum*. Wiley Online Library.

Du, Yilun, M. Katherine Collins, B. Joshua Tenenbaum, and Vincent Sitzmann (2021). "Learning Signal-Agnostic Manifolds of Neural Fields." In: *NeurIPS*.

Dupont, Emilien, Adam Goliński, Milad Alizadeh, Yee Whye Teh, and Arnaud Doucet (2021). "Coin: Compression with implicit neural representations." In: *ICLR (Neural Compression Workshop)*.

Dupont, Emilien, Hyunjik Kim, S. M. Ali Eslami, Danilo Jimenez Rezende, and Dan Rosenbaum (2022a). "From data to functa: Your data point is a function and you can treat it like one." In: PMLR.

Dupont, Emilien, Yee Whye Teh, and Arnaud Doucet (2022b). "Generative Models as Distributions of Functions." In: PMLR.

Erkoç, Ziya, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai (2023). "Hyperdiffusion: Generating implicit neural fields with weight-space diffusion." In: *ICCV*.

Essakine, Amer, Yanqi Cheng, Chun-Wun Cheng, Lipei Zhang, Zhongying Deng, Lei Zhu, Carola-Bibiane Schönlieb, and Angelica I Aviles-Rivero (2025). "Where Do We Stand with Implicit Neural Representations? A Technical and Performance Survey." In: *TMLR*.

Fan, Jiahui, Fujun Luan, Jian Yang, Milos Hasan, and Beibei Wang (2025). "RNG: Relightable Neural Gaussians." In: *CVPR*.

Farbman, Zeev, Raanan Fattal, and Dani Lischinski (2011). "Convolution pyramids." In: *ACM Transactions on Graphics (TOG)*.

Fathony, Rizal, Anit Kumar Sahu, Devin Willmott, and J Zico Kolter (2020). "Multiplicative filter networks." In: *ICLR*.

Fournier, Alain and Eugene Fiume (1988). "Constant-time filtering with space-variant kernels." In: *ACM Transactions on Graphics (TOG)*.

Freeman, William T, Edward H Adelson, et al. (1991). "The design and use of steerable filters." In: *TPAMI*.

Gao, Jian, Chun Gu, Youtian Lin, Zhihao Li, Hao Zhu, Xun Cao, Li Zhang, and Yao Yao (2024). "Relightable 3d gaussians: Realistic point cloud relighting with brdf decomposition and ray tracing." In: *ECCV*.

Gkioulekas, Ioannis, Anat Levin, and Todd Zickler (2016). "An evaluation of computational imaging techniques for heterogeneous inverse scattering." In: *ECCV*. Springer.

Gkioulekas, Ioannis, Shuang Zhao, Kavita Bala, Todd Zickler, and Anat Levin (2013). "Inverse volume rendering with material dictionaries." In: *ACM Transactions on Graphics (TOG)*.

Gonzalez, Rafael C (2009). *Digital image processing*. Pearson education india.

Hadadan, Saeed, Shuhong Chen, and Matthias Zwicker (2021). "Neural radiosity." In: *ACM Transactions on Graphics (TOG)*.

Hadadan, Saeed, Geng Lin, Jan Novák, Fabrice Rousselle, and Matthias Zwicker (2023). "Inverse global illumination using a neural radiometric prior." In: *ACM SIGGRAPH*.

Heckbert, Paul S (1986). "Filtering by repeated integration." In: *ACM Transactions on Graphics (TOG)*.

Henyey, Louis G and Jesse Leonard Greenstein (1941). "Diffuse radiation in the galaxy." In: *Astrophysical Journal, vol. 93, p. 70-83 (1941)*.

Henzler, Philipp, Niloy J Mitra, and Tobias Ritschel (2019). "Escaping plato's cave: 3d shape from adversarial rendering." In: *ICCV*.

Hermosilla, Pedro, Tobias Ritschel, Pere-Pau Vázquez, Àlvar Vinacua, and Timo Ropinski (2018). "Monte carlo convolution for learning on non-uniformly sampled point clouds." In: *ACM Transactions on Graphics (TOG)*.

Hertz, Amir, Or Perel, Raja Giryes, Olga Sorkine-Hornung, and Daniel Cohen-Or (2021). "Sape: Spatially-adaptive progressive encoding for neural optimization." In: *NeurIPS*.

Heusel, Martin, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter (2017). "Gans trained by a two time-scale update rule converge to a local nash equilibrium." In: *NeurIPS*.

Isaac-Medina, Brian K. S., Chris G. Willcocks, and Toby P. Breckon (2023). "Exact-NeRF: An Exploration of a Precise Volumetric Parameterization for Neural Radiance Fields." In: *CVPR*.

Jakob, Wenzel, Adam Arbree, Jonathan T Moon, Kavita Bala, and Steve Marschner (2010). "A radiative transfer framework for rendering materials with anisotropic structure." In: *ACM SIGGRAPH*.

JangaFX (2026). "smoke assets." In: *JangaFX*. URL: https://jangafx.com/.

Jiang, Kaiwen, Shu-Yu Chen, Hongbo Fu, and Lin Gao (2023). "Nerffacelighting: Implicit and disentangled face lighting representation leveraging generative prior in neural radiance fields." In: *ACM Transactions on Graphics (TOG)*.

Jin, Haian, Isabella Liu, Peijia Xu, Xiaoshuai Zhang, Songfang Han, Sai Bi, Xiaowei Zhou, Zexiang Xu, and Hao Su (2023). "Tensoir: Tensorial inverse rendering." In: *CVPR*.

JojoV (2018). "Henyey–Greenstein." In: *Wikimedia Commons*. URL: https://commons.wikimedia.org/w/index.php?title=User:Jojo_V&action=edit&redlink=1.

Kajiya, James T. (1986). "The rendering equation." In: *ACM SIGGRAPH*. Association for Computing Machinery.

Kajiya, James T and Brian P Von Herzen (1984). "Ray tracing volume densities." In: *ACM SIGGRAPH*.

Kallweit, Simon, Thomas Müller, Brian Mcwilliams, Markus Gross, and Jan Novák (Nov. 2017). "Deep scattering: rendering atmospheric clouds with radiance-predicting neural networks." In: *ACM Transactions on Graphics (TOG)*.

Kerbl, Bernhard, Georgios Kopanas, Thomas Leimkühler, and George Drettakis (2023). "3D Gaussian Splatting for Real-Time Radiance Field Rendering." In: *ACM Transactions on Graphics (TOG)*.

Kettunen, Markus, Marco Manzi, Miika Aittala, Jaakko Lehtinen, Frédo Durand, and Matthias Zwicker (2015). "Gradient-domain path tracing." In: *ACM Transactions on Graphics (TOG)*.

Kingma, Diederik P. and Jimmy Ba (2015). "Adam: A Method for Stochastic Optimization." In: *ICLR*.

Kopanas, Georgios, Thomas Leimkühler, Gilles Rainer, Clément Jambon, and George Drettakis (2022). "Neural Point Catacaustics for Novel-View Synthesis of Reflections." In: *ACM Transactions on Graphics (TOG)*.

Lehtinen, Jaakko, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila (2018). "Noise2Noise: Learning Image Restoration without Clean Data." In: *ICML*.

Leimkühler, Thomas, Hans-Peter Seidel, and Tobias Ritschel (2018). "Laplacian Kernel Splatting for Efficient Depth-of-field and Motion Blur Synthesis or Reconstruction." In: *ACM Transactions on Graphics (TOG)*.

Levis, Aviad, Yoav Y Schechner, Amit Aides, and Anthony B Davis (2015). "Airborne three-dimensional cloud tomography." In: *ICCV*.

Levis, Aviad, Yoav Y Schechner, and Anthony B Davis (2017). "Multiple-scattering microphysics tomography." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Li, Jia, Lu Wang, Lei Zhang, and Beibei Wang (2024). "Tensosdf: Roughness-aware tensorial representation for robust geometry and material reconstruction." In: *ACM Transactions on Graphics (TOG)*.

Li, Tzu-Mao, Miika Aittala, Frédo Durand, and Jaakko Lehtinen (2018). "Differentiable monte carlo ray tracing through edge sampling." In: *ACM Transactions on Graphics (TOG)*.

Liang, Zhihao, Qi Zhang, Ying Feng, Ying Shan, and Kui Jia (2024). "Gs-ir: 3d gaussian splatting for inverse rendering." In: *CVPR*.

Lindeberg, Tony (2013). *Scale-space theory in computer vision*. Springer Science & Business Media.

Lindell, David B, Julien NP Martel, and Gordon Wetzstein (2021). "Autoint: Automatic integration for fast neural volume rendering." In: *CVPR*.

Lindell, David B, Dave Van Veen, Jeong Joon Park, and Gordon Wetzstein (2022). "Bacon: Band-limited coordinate networks for multiscale scene representation." In: *CVPR*.

Liu, Yibo, Zhixin Fang, Sune Darkner, Noam Aigerman, Kenny Erleben, Paul Kry, and Teseo Schneider (2025). "Neural Kinematic Bases for Fluids." In: *ACM SIGGRAPH*.

Liu, Zhen, Hao Zhu, Qi Zhang, Jingde Fu, Weibing Deng, Zhan Ma, Yanwen Guo, and Xun Cao (2024). "Finer: Flexible spectral-bias tuning in implicit neural representation by variable-periodic activation functions." In: *CVPR*.

Lyu, Linjie, Ayush Tewari, Thomas Leimkuehler, Marc Habermann, and Christian Theobalt (2022). "Neural Radiance Transfer Fields for Relightable Novel-view Synthesis with Global Illumination." In: *ECCV*.

McCormick, NJ (1992). "Inverse radiative transfer problems: a review." In: *Nuclear science and Engineering*.

Mildenhall, Ben, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng (2020). "Nerf: Representing scenes as neural radiance fields for view synthesis." In: *ECCV*.

Mishra, Siddhartha and Roberto Molinaro (2021). "Physics informed neural networks for simulating radiative transfer." In: *Journal of Quantitative Spectroscopy and Radiative Transfer*.

Mitchel, Thomas W., Benedict Brown, David Koller, Tim Weyrich, Szymon Rusinkiewicz, and Michael Kazhdan (June 2020). *Efficient Spatially Adaptive Convolution and Correlation*. Tech. rep. arXiv preprint.

Mujkanovic, Felix, Ntumba Elie Nsampi, Christian Theobalt, Hans-Peter Seidel, and Thomas Leimkühler (July 2024). "Neural Gaussian Scale-Space Fields." In: *ACM Transactions on Graphics (TOG)*.

Müller, Norman, Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulo, Peter Kontschieder, and Matthias Nießner (2023). "Diffrf: Rendering-guided 3d radiance field diffusion." In: *CVPR*.

Müller, Thomas, Alex Evans, Christoph Schied, and Alexander Keller (2022). "Instant Neural Graphics Primitives with a Multiresolution Hash Encoding." In: *ACM Transactions on Graphics (TOG)*.

Müller, Thomas, Alex Evans, Christoph Schied, and Alexander Keller (2022). "Instant neural graphics primitives with a multiresolution hash encoding." In: *ACM transactions on graphics (TOG)*.

Nicolet, Baptiste, Fabrice Rousselle, Jan Novak, Alexander Keller, Wenzel Jakob, and Thomas Müller (2023). "Recursive control variates for inverse rendering." In: *ACM Transactions on Graphics (TOG)*.

Niederreiter, Harald (1992). "Low-discrepancy point sets obtained by digital constructions over finite fields." In: *Czechoslovak Mathematical Journal*.

Nimier-David, Merlin, Thomas Müller, Alexander Keller, and Wenzel Jakob (2022). "Unbiased inverse volume rendering with differential trackers." In: *ACM Transactions on Graphics (TOG)*.

Nimier-David, Merlin, Sébastien Speierer, Benoît Ruiz, and Wenzel Jakob (2020). "Radiative backpropagation: An adjoint method for lightning-fast differentiable rendering." In: *ACM Transactions on Graphics (TOG)*.

Nimier-David, Merlin, Delio Vicini, Tizian Zeltner, and Wenzel Jakob (2019). "Mitsuba 2: A retargetable forward and inverse renderer." In: *ACM Transactions on Graphics (TOG)*.

Nsampi, Ntumba Elie, Adarsh Djeacoumar, Hans-Peter Seidel, Tobias Ritschel, and Thomas Leimkühler (Dec. 2023). "Neural Field Convolutions by Repeated Differentiation." In: *ACM Transactions on Graphics (TOG)*.

– (2026). "Volumetric Inverse Rendering via Neural Radiative Transfer." In: *Computer Graphics Forum*. DOI: 10.1111/cgf.70541.

Owen, Art B. (2013). *Monte Carlo theory, methods and examples*. https://artowen.su.domains/mc/.

Pan, Xingang, Xudong Xu, Chen Change Loy, Christian Theobalt, and Bo Dai (2021). "A shading-guided generative implicit model for shape-accurate 3d-aware image synthesis." In: *NeurIPS*.

Park, Jeong Joon, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove (2019). "Deepsdf: Learning continuous signed distance functions for shape representation." In: *CVPR*.

Park, Keunhong, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla (2021). "Nerfies: Deformable neural radiance fields." In: *ICCV*.

Paszke, Adam, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer (2017). "Automatic differentiation in pytorch." In.

Perlin, Kenneth (1984). Personal communication with Paul Heckbert, mentioned in Heckbert [1986].

Peter Shirley Trevor David Black, Steve Hollasch (2025). *Ray Tracing in One Weekend*.

Pharr, Matt, Wenzel Jakob, and Greg Humphreys (2016). *Physically Based Rendering: From Theory to Implementation*. 3rd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Pharr, Matt, Wenzel Jakob, and Greg Humphreys (2023). *Physically based rendering: From theory to implementation*. MIT Press.

Rahaman, Nasim, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville (2019). "On the spectral bias of neural networks." In: *ICML*.

Rahimi, Ali and Benjamin Recht (2007). "Random features for large-scale kernel machines." In: *NeurIPS*.

Raissi, Maziar, Paris Perdikaris, and George E Karniadakis (2019). "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." In: *Journal of Computational physics*.

Ramachandran, Prajit, Barret Zoph, and Quoc V Le (2017). "Searching for activation functions." In: *arXiv preprint arXiv:1710.05941*.

Ramasinghe, Sameera and Simon Lucey (2022). "Beyond periodicity: Towards a unifying framework for activations in coordinate-mlps." In: *ECCV*.

Rebain, Daniel, Soroosh Yazdani, Kwang Moo Yi, and Andrea Tagliasacchi (2024). "Neural fields as distributions: Signal processing beyond Euclidean space." In: *CVPR*.

Riganti, Roberto and L Dal Negro (2023). "Auxiliary physics-informed neural networks for forward, inverse, and coupled radiative transfer problems." In: *Applied Physics Letters*.

Rubab, Fizza, Ntumba Elie Nsampi, Martin Balint, Felix Mujkanovic, Hans-Peter Seidel, Tobias Ritschel, and Thomas Leimkühler (2025). "Learning Neural Antiderivatives." In: *Vision, Modeling, and Visualization*. Ed. by Bernhard Egger and Tobias Günther. The Eurographics Association.

Saragadam, Vishwanath, Daniel LeJeune, Jasper Tan, Guha Balakrishnan, Ashok Veeraraghavan, and Richard G Baraniuk (2023). "Wire: Wavelet implicit neural representations." In: *CVPR*.

Saratchandran, Hemanth, Sameera Ramasinghe, Violetta Shevchenko, Alexander Long, and Simon Lucey (2024). "A sampling theory perspective on activations for implicit neural representations." In: *arXiv preprint arXiv:2402.05427*.

Sharp, Nicholas and Alec Jacobson (2022). "Spelunking the Deep: Guaranteed Queries on General Neural Implicit Surfaces via Range Analysis." In: *ACM Transactions on Graphics (TOG)*.

Shi, Yahao, Yanmin Wu, Chenming Wu, Xing Liu, Chen Zhao, Haocheng Feng, Jian Zhang, Bin Zhou, Errui Ding, and Jingdong Wang (2025). "GIR: 3D Gaussian Inverse Rendering for Relightable Scene Factorization." In: *TPAMI*.

Shocher, Assaf, Ben Feinstein, Niv Haim, and Michal Irani (2020). "From discrete to continuous convolution layers." In: *arXiv preprint arXiv:2006.11120*.

Simard, Patrice, Léon Bottou, Patrick Haffner, and Yann LeCun (1998). "Boxlets: a fast convolution algorithm for signal processing and neural networks." In: *NeurIPS*.

Singh, Gurprit, Cengiz Öztireli, Abdalla GM Ahmed, David Coeurjolly, Kartic Subr, Oliver Deussen, Victor Ostromoukhov, Ravi Ramamoorthi, and Wojciech Jarosz (2019). "Analysis of sample correlations for Monte Carlo rendering." In: *Computer Graphics Forum*. Wiley Online Library.

Sitzmann, Vincent, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein (2020). "Implicit neural representations with periodic activation functions." In: *NeurIPS*.

Sitzmann, Vincent, Semon Rezchikov, Bill Freeman, Josh Tenenbaum, and Fredo Durand (2021). "Light field networks: Neural scene representations with single-evaluation rendering." In: *NeurIPS*.

Sitzmann, Vincent, Michael Zollhöfer, and Gordon Wetzstein (2019). "Scene Representation Networks: Continuous 3D-Structure-Aware Neural Scene Representations." In: *NeurIPS*.

Sobol, Ilya Meerovich (1967). "On the distribution of points in a cube and the approximate evaluation of integrals." In: *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*.

Solomon, Justin (2015). *Numerical algorithms: methods for computer vision, machine learning, and graphics*. CRC press.

Song, Ying, Jiaping Wang, Li-Yi Wei, and Wencheng Wang (Dec. 2016). "Vector Regression Functions for Texture Compression." In: *ACM Transactions on Graphics (TOG)*.

Speierer, Sébastien and Adrian Jarabo (2026). "Generalized non-exponential Gaussian splatting." In: *arXiv preprint arXiv:2603.02887*.

Srinivasan, Pratul P, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T Barron (2021). "Nerv: Neural reflectance and visibility fields for relighting and view synthesis." In: *CVPR*.

Stamnes, Knut, Si-Chee Tsay, Warren Wiscombe, and Istvan Laszlo (2000). "DISORT, a general-purpose Fortran program for discrete-ordinate-method radiative transfer in scattering and emitting layered media: documentation of methodology." In.

Stanley, Kenneth O (2007). "Compositional pattern producing networks: A novel abstraction of development." In: *Genetic programming and evolvable machines*.

Tancik, Matthew, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Bar-

ron, and Ren Ng (2020). "Fourier features let networks learn high frequency functions in low dimensional domains." In: *NeurIPS*.

Technologies, Tanso (2026). "smoke chemney." In: *Tanso Technologies*. URL: https://www.tanso.de/en.

Tewari, Ayush, Justus Thies, Ben Mildenhall, Pratul Srinivasan, Edgar Tretschk, W Yifan, Christoph Lassner, Vincent Sitzmann, Ricardo Martin-Brualla, Stephen Lombardi, et al. (2022). "Advances in neural rendering." In: *Computer Graphics Forum*.

Tomasi, Carlo and Roberto Manduchi (1998). "Bilateral filtering for gray and color images." In: *ICCV*.

Tretschk, Edgar, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt (2021). "Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video." In: *ICCV*.

TungArt7 (2026). "clouds." In: *pixabay*. URL: https://pixabay.com/users/tungart7-38741244/.

Vasconcelos, Cristina, Kevin Swersky, Mark Matthews, Milad Hashemi, Cengiz Oztireli, and Andrea Tagliasacchi (2023). "CUF: Continuous Upsampling Filters." In: *CVPR*.

Vicini, Delio, Sébastien Speierer, and Wenzel Jakob (2021). "Path replay backpropagation: Differentiating light paths using constant memory and linear time." In: *ACM Transactions on Graphics (TOG)*.

– (2022). "Differentiable signed distance function rendering." In: *ACM Transactions on Graphics (TOG)*.

Viola, Paul and Michael Jones (2001). "Rapid object detection using a boosted cascade of simple features." In: *CVPR*.

Violante, Nicolás, Alban Gauthier, Stavros Diolatzis, Thomas Leimkühler, and George Drettakis (2024). "Physically-Based Lighting for 3D Generative Models of Cars." In: *Computer Graphics Forum*.

Wang, Shenlong, Simon Suo, Wei-Chiu Ma, Andrei Pokrovsky, and Raquel Urtasun (2018). "Deep parametric continuous convolutional neural networks." In: *CVPR*.

Wang, Sifan, Xinling Yu, and Paris Perdikaris (2022a). "When and why PINNs fail to train: A neural tangent kernel perspective." In: *Journal of Computational Physics*.

Wang, Yinhuai, Shuzhou Yang, Yujie Hu, and Jian Zhang (2022b). "NeRFocus: Neural Radiance Field for 3D Synthetic Defocus." In: *arXiv preprint arXiv:2203.05189*.

Wang, Zhou, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli (2004). "Image quality assessment: from error visibility to structural similarity." In: *IEEE transactions on image processing*.

Weier, Philippe, Jérémy Riviere, Ruslan Guseinov, Stephan Garbin, Philipp Slusallek, Bernd Bickel, Thabo Beeler, and Delio Vicini (2025). "Practical Inverse Rendering of Textured and Translucent Appearance." In: *ACM Transactions on Graphics (TOG)*.

Williams, Lance (1983). "Pyramidal parametrics." In: *ACM SIGGRAPH*.

Witkin, Andrew P (1987). "Scale-space filtering." In: *Readings in Computer Vision*.

Worchel, Markus, Ugo Finnendahl, and Marc Alexa (2025). "Radiative Backpropagation with Non-Static Geometry." In: *Proc. EGSR*.

Wu, Sean, Shamik Basu, Tim Broedermann, Luc Van Gool, and Christos Sakaridis (2025). "PBR-NeRF: Inverse Rendering with Physics-Based Neural Fields." In: *CVPR*.

Xie, Yiheng, Towaki Takikawa, Shunsuke Saito, Or Litany, Shiqin Yan, Numair Khan, Federico Tombari, James Tompkin, Vincent Sitzmann, and Srinath Sridhar (2022). "Neural fields in visual computing and beyond." In: *Computer Graphics Forum*.

Xu, Dejia, Peihao Wang, Yifan Jiang, Zhiwen Fan, and Zhangyang Wang (2022). "Signal Processing for Implicit Neural Representations." In: *NeurIPS*.

Yaldiz, Mustafa B., Ishit Mehta, Nithin Raghavan, Andreas Meuleman, Tzu-Mao Li, and Ravi Ramamoorthi (2025). "Spectral Prefiltering of Neural Fields." In: *ACM SIGGRAPH*.

Yang, Guandao, Serge Belongie, Bharath Hariharan, and Vladlen Koltun (2021). "Geometry processing with neural fields." In: *NeurIPS*.

Yang, Guandao, Sagie Benaim, Varun Jampani, Kyle Genova, Jonathan Barron, Thomas Funkhouser, Bharath Hariharan, and Serge Belongie (2022). "Polynomial neural fields for subband decomposition and manipulation." In: *NeurIPS*.

Yao, Yao, Jingyang Zhang, Jingbo Liu, Yihang Qu, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan (2022). "Neilf: Neural incident light field for physically-based material estimation." In: *ECCV*. Springer.

Yu, Hong-Xing, Yang Zheng, Yuan Gao, Yitong Deng, Bo Zhu, and Jiajun Wu (2024). "Inferring hybrid neural fluid fields from videos." In: *NeurIPS*.

Yuan, Yu-Jie, Yang-Tian Sun, Yu-Kun Lai, Yuewen Ma, Rongfei Jia, and Lin Gao (2022). "NeRF-editing: geometry editing of neural radiance fields." In: *ICCV*.

Zaal, Greg (2026). "Env Map." In: *POlyheaven*. URL: https://polyhaven.com/alla=Greg20Zaal.

Zeltner, Tizian, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob (2021). "Monte Carlo estimators for differential light transport." In: *ACM Transactions on Graphics (TOG)*.

Zhang, Cheng, Lifan Wu, Changxi Zheng, Ioannis Gkioulekas, Ravi Ramamoorthi, and Shuang Zhao (2019). "A differential theory of radiative transfer." In: *ACM Transactions on Graphics (TOG)*.

Zhang, Cheng, Zihan Yu, and Shuang Zhao (2021a). "Path-space differentiable rendering of participating media." In: *ACM Transactions on Graphics (TOG)*.

Zhang, Kai, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely (2021b). "Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting." In: *CVPR*.

Zhang, Richard, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang (2018). "The unreasonable effectiveness of deep features as a perceptual metric." In: *CVPR*.

Zhang, Xian-Da (2022). "Modern signal processing." In: *Modern Signal Processing*. De Gruyter.

Zhang, Xiuming, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron (2021c). "Nerfactor: Neural factorization of shape and reflectance under an unknown illumination." In: *ACM Transactions on Graphics (TOG)*.

Zhang, Youjia, 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: *ICCV*.

Zhang, Yuqing, Yuan Liu, Zhiyu Xie, Lei Yang, Zhongyuan Liu, Mengzhou Yang, Runze Zhang, Qilong Kou, Cheng Lin, Wenping Wang, et al. (2024). "Dreammat: High-quality pbr material generation with geometry-and light-aware diffusion models." In: *ACM Transactions on Graphics (TOG)*.

Zheng, Quan, Gurprit Singh, and Hans-Peter Seidel (2021). "Neural relightable participating media rendering." In: *NeurIPS*.

Zucker, Shai, Dmitry Batenkov, and Michal Segal Rozenhaimer (2025). "Physics-informed neural networks for modeling atmospheric radiative transfer." In: *Journal of Quantitative Spectroscopy and Radiative Transfer*.

dexmac (2026). "underwater." In: *pixabay*. URL: https://pixabay.com/users/dexmac-12233086/.

hamist (2026). "fog." In: *pixabay*. URL: https://pixabay.com/users/hamist-571446/.

shogun (2026). "smoke." In: *pixabay*. URL: https://pixabay.com/users/shogun-1310047/.

xiSerge (2026). "flower." In: *pixabay*. URL: https://pixabay.com/de/users/xiserge15871962/.