IEEE Journal of
Microwaves

MTT-S
MHz to THz Community

Received 26 November 2024; revised 13 January 2025; accepted 17 January 2025.

Digital Object Identifier 10.1109/JMW.2025.3535077

Inverse Rendering of Near-Field mmWave MIMO Radar for Material Reconstruction

NIKOLAI HOFMANN [1], VANESSA WIRTH [1], JOHANNA BRÄUNIG [2] (Student Member, IEEE), INGRID ULLMANN [2] (Member, IEEE), MARTIN VOSSIEK [2] (Fellow, IEEE), TIM WEYRICH [1,3], AND MARC STAMMINGER [1]

(Regular Paper)

[1]Visual Computing Erlangen, Friedrich-Alexander-Universität Erlangen-Nürnberg, 91054 Erlangen, Germany
[2]Institute of Microwaves and Photonics, Friedrich-Alexander-Universität Erlangen-Nürnberg, 91054 Erlangen, Germany
[3]University College London, WC1E London, U.K.

CORRESPONDING AUTHOR: Nikolai Hofmann (e-mail: nikolai.hofmann@fau.de).

This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – SFB 1483 – Project-ID 442419336, EmpkinS. The authors gratefully acknowledge the scientific support and HPC resources provided by the Erlangen National High Performance Computing Center (NHR@FAU) of the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) under the NHR Project b201dc. NHR is supported by Federal and Bavarian State Authorities. NHR@FAU Hardware was supported by the DFG – Project-ID 440719683.

ABSTRACT Near-field multiple-input multiple-output (MIMO) radar systems allow for high-resolution spatial imaging by leveraging multiple antennas to transmit and receive signals across multiple perspectives. This capability is particularly advantageous in challenging environments, where optical imaging techniques struggle. We present a novel approach to inverse rendering for near-field MIMO radar systems, aimed at reconstructing material properties such as surface roughness, dielectric constants, and conductivity from radar and ground-truth mesh data, for example obtained from multi-view stereo. Drawing inspiration from physically based rendering techniques in computer graphics, we formalize an advanced inverse rendering algorithm that integrates electromagnetic wave propagation models directly into the optimization process. To avoid bias from conventional radar image reconstruction algorithms in the optimization process, we directly derive gradients from raw radar outputs, resulting in more accurate material characterization. We validate our approach through extensive experiments on both synthetic and real radar datasets, demonstrating its effectiveness in a multitude of scenarios.

INDEX TERMS Backpropagation, MIMO radar, radar simulation, ray tracing, scattering parameters.

I. MOTIVATION

Near-field multiple-input multiple-output (MIMO) imaging radar systems are capable of providing high-resolution spatial data due to their ability to simultaneously transmit and receive signals across multiple antennas and different perspectives. This enables highly detailed imaging of objects close to the radar, even in challenging environments with limited visibility from occlusions, clutter, or adverse weather. Inverse rendering in the context of near-field MIMO imaging radar is an important research direction due to its potential to significantly enhance object recognition, scene reconstruction, and material characterization. While traditional rendering techniques from computer graphics aim to generate realistic images from 3D models by simulating the physical behavior of light, inverse rendering reverses this process and aims to extract 3D models, material properties, or other scene characteristics from sensor data.

Our goal is to combine domain-specific knowledge from computer graphics with near-field MIMO imaging radar, where we investigate the potential of inverse rendering to reconstruct detailed material properties. However, the process of accurately reconstructing the material properties of an object from radar data is inherently challenging due to the complex nature of interactions between electromagnetic waves and the scene, which includes scattering, diffraction, and multi-path propagation, for example.

This paper derives a novel approach to inverse rendering for near-field MIMO imaging radar, with the specific goal of reconstructing material properties such as surface roughness, dielectric constants, and conductivity, in the form of complex indices of refraction. We propose an advanced inverse rendering algorithm inspired by physically based rendering from computer graphics, which leverages the rich data provided by MIMO imaging radars and their capacity to capture fine spatial details. By incorporating different models for electromagnetic wave propagation into the optimization process and directly deriving gradients from the raw outputs of a real MIMO radar, we achieve robust optimizations which enable a multitude of down-stream tasks. We validate and explore the strengths and weaknesses of our proposed framework through a series of experiments, both using synthetic and a diverse dataset of real radar data, evaluating its effectiveness in reconstructing material properties in various scenarios.

In summary, our contributions are:

- Formal derivation of an inverse rendering framework for near-field MIMO imaging radar using ray tracing
- Transfer of domain-specific knowledge from computer graphics and physically based rendering to radar simulation
- Direct derivation of inverse rendering gradients from raw radar outputs without bias from reconstruction algorithms
- Comprehensive evaluation of the proposed inverse rendering framework for near-field MIMO radar with the goal of material parameter reconstruction

II. PREVIOUS WORKS

II.

1) IMAGING RADAR

Analogously to photo cameras, imaging radars produce 2D images of their surroundings; however, while operating at radio wavelengths instead of visible light. Synthetic antenna aperture (SAR) imaging radars typically employ either multiple-input and multiple-output (MIMO) channels from multiple antennas at different positions [1], or rely on movement of the antenna or the observed object, which creates a virtually enlarged antenna aperture over time [2]. Elaborate reconstruction algorithms use this additional information to resolve radar data from multiple antennas or positions into one high-resolution image [3], [4], [5]. Leveraging both optical and radar data, Brinkmann et al. [6] proposed to estimate material properties and perform material characterization from measurements of a MIMO radar, which they then validate using a quasi-optical setup. While pursuing similar goals, they chose a more hardware-centric approach, whilst we rely on inverse rendering and seek a mostly software-based solution instead.

2) RADAR SIMULATION

The shooting and bouncing rays (SBR) [7] method is a proven approach to radar simulation, which models wave propagation using ray tracing and is additionally encouraged by hardware support in modern GPUs. With the upswing of machine learning and neural networks, radar simulation became an important tool to generate large-scale datasets to satisfy the need of data-driven approaches, such as deep learning. Yun and Iskander [8] provide a comprehensive overview over the basic principles of modeling radio wave propagation using ray tracing. Ortiz-Jimenez et al. [9] explore the applicability of microfacet scattering models from the realm of computer graphics to a 3D high-resolution imaging radar for hidden object detection at 300 GHz. In the context of autonomous driving and advanced driver assistance systems (ADAS), radar simulation on digital twins [10], [11] plays a crucial role to explore the design space of different sensor types and their applicability. Furthermore, Schüßler et al. [12] propose a radar simulator for MIMO arrays for ADAS, where they rely on a simplistic material model to both support diffuse and specular reflection, which provides realistic radar imagery even for large antenna arrays. They later extend their approach to allow for automatic data annotation via ray meta data [13], such as Doppler or hit geometry information, which enables advanced data-driven applications which would otherwise be impossible or too costly to realize on real data.

3) PHYSCIALLY BASED RENDERING

We recommend the excellent book by Pharr et al. [14], which provides an extensive description of state-of-the-art rendering techniques and algorithms. They cover both theory and implementation by providing code examples and an open-source implementation that runs on both CPU and GPU. Thus, we abstain from an elaborate survey and refer the interested reader to their work.

4) DIFFERENTIABLE RENDERING

Inverse, or differentiable, rendering receives increasing attention from the rendering community, enabled by differentiable rendering primitives, such as rasterization [15], ray tracing [16], or volumetric rendering [17]. This gave rise to a multitude of applications, for example in the context of novel view synthesis [18], [19], image-based rendering [20], and multi-view reconstruction [21]. A valuable tool to differentiate complex GPU-based rendering algorithms, or frameworks with a large codebase, is compiler-supported automatic differentiation [22], [23]. While automatic differentiation only yields gradients for continuous functions, discontinuous integrands can be handled using edge sampling [24], change of variables [25], or warped-area sampling [26].

5) DIFFERENTIABLE RADAR RENDERING

Inverse rendering has already been proven to be an equally valuable tool for radar simulation applications. Fu et al. [27] propose a differential renderer in the context of remote sensing using synthetic aperture radar (SAR), where they analytically derive gradients using probability maps, to help with information retrieval and object recognition. Analogously, Wilmanski and Tamir [28] employ a differential rasterizer, which produces feature maps which are then consumed by a neural renderer based on generative neural networks to produce SAR imagery. They demonstrate how to train their system end-to-end and apply to down-stream tasks such as object recognition on synthetic data. In the context of radio communications, Hoydis et al. [29] propose a differentiable ray tracing tool which simulates and optimizes material and sensor properties to visualize and maximize coverage. Inspired by neural radiance fields (NeRF) [17], Huang et al. [30] introduce a novel view synthesis algorithm, which is based on mmWave and LIDAR data and obeys radar-specific physics. After training on range-Doppler images, their implicit tomography representation can be rendered from novel viewpoints using a reflectance and transmittance-based rendering pipeline. Similarly, Borts et al. [31] propose to leverage neural scene representations to reconstruct dense 3D occupancy from 2D Frequency-Modulated Continuous-Wave (FMCW) radar data. They combine implicit neural geometry with physics-informed sensor and reflectance models, while operating in Fourier frequency space as opposed to relying on volumetric rendering, which is common for NeRF-based approaches.

III. RADAR RENDERING

III.

1) RADAR EQUATION

The radar equation [32] is a common model for ray tracing based radar simulation algorithms [8], [10]. It yields the ratio between transmitted, P_t, and received power, P_r, in a radar

$$\frac{P_r}{P_t} = \frac{G_t A_r \sigma}{(4\pi)^2 d_t^2 d_r^2} = \frac{G_t G_r \lambda^2 \sigma}{(4\pi)^3 d_t^2 d_r^2}, \tag{1}$$

where λ is the wavelength, G_t and G_r the transmitting and receiving antenna gains, and d_t, d_r the distances of the object to the transmitter and receiver, respectively. A_r is the effective aperture of the receiving antenna and is related to the antenna gain G_r by [33]

$$A_r = \frac{G_r \lambda^2}{4\pi}. \tag{2}$$

The radar cross section (RCS) σ $[m^2]$ defines the scattering properties of an observed object as a whole, which is approximated as a single point scatterer in the far field. The RCS depends on a multitude of complicated-to-model factors, such as, geometry, material, size relative to wavelength, incident and reflected angles, and polarization, which are aggregated into a single value. This is in contrast to reflectance models in computer graphics, where each point on the surface of an object is parameterized and evaluated individually.

The dimensionless antenna gain $G_i = \frac{S_{\text{real}}}{S_{\text{isotropic}}}$ is the ratio of the maximum signal strength S of the main lobe to the signal strength of an ideal (lossless) isotropic antenna. For highly directional antennas, such as antenna arrays with a non-uniform main lobe and multiple side lobes, simple scalar antenna gains G_t and G_r are not sufficient, especially in the near field. Thus, we employ directionally varying antenna gains $G_t(\omega_t)$ and $G_r(\omega_r)$, which are pre-computed using electromagnetic field simulation software and stored as look-up tables for each antenna model. Both the gain and power of a radar antenna are commonly expressed in decibels: $G_{\text{dBi}} = 10 \log_{10} G$ and $P_{\text{dBW}} = 10 \log_{10} P$.

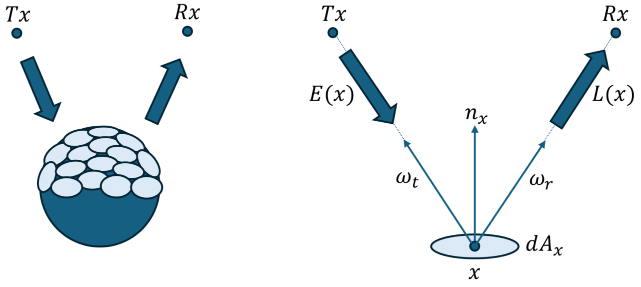


FIGURE 1. To compute the near field reflections of a scene object, we subdivide it into surface elements (left), and compute the reflection for each such surface element separately (right).

2) RADAR RENDERING

We employ the term *radar rendering* as a synonym for simulating the received returns of a given radar system and some scene object using geometrical optics and ray tracing, which is also referred to as the shooting and bouncing rays (SBR) method [12] in literature. The goal is to solve for the measured *flux* $[W]$ at each receiving antenna, which was previously emitted from a transmitting antenna and reflected from the scene object, analogous to a physical radar. Since the radar equation approximates each back-scattering object as a single point target, we need to extend this model for application in the near field, where objects are much closer to the radar and cover a larger solid angle. To this end, we transfer existing concepts from global illumination and physically based rendering [14] to radar simulation. We argue that, with the exception of particular wave effects such as diffraction, the core ideas from computer-graphics models designed for transport in the visible spectrum, i.e., $\lambda \in [380 \text{ nm}, 750 \text{ nm}]$, can be applied in the domain of mmWave radar as well.

Given is a scene object S in the near field of a MIMO radar. First, we subdivide S into surface elements visible to the radar, as shown in Fig. 1(left). We then separately compute the (direct) reflection from each surface element, and finally add all contributions. To achieve accurate results, the surface elements must be small enough to account for all relevant wave effects; we will elaborate on this later. For now, we assume the surface elements are infinitesimally small. Each such surface element has a position x, an orientation n_x, and an area $\mathrm{d}A_x$ (Fig. 1 right). In the following, we derive the radar cross section of such a surface element $\mathrm{d}A_x$, taking into account its orientation and distance, as well as its reflectance characteristics. To this end, we first compute the incident power density, or *irradiance* E $[Wm^{-2}]$, from a transmitting antenna Tx_i at x, according to (3):

$$E(x, \omega_t) = \frac{P_t G_t}{4\pi d_t^2} \; [Wm^{-2}], \tag{3}$$

where d_t^2 is the distance between the transmitting antenna and x and ω_t the direction towards the sender (see Fig. 1).

To model the reflection of this irradiance at a surface point x, we use the concept of a *bidirectional reflectance distribution function (BRDF)* [34]. A BRDF $f(\omega_t, x, \omega_r)$ describes which fraction of irradiance coming from an incident direction ω_t is reflected by surface point x into exitant direction ω_r. In Sect. V we will consider a variety of possible models to describe f. According to the definition of a BRDF, the reflected radiance is

$$L(x, \omega_r) = E(x, \omega_t)\cos(n_x, \omega_t) f(\omega_t, x, \omega_r), \tag{4}$$

where $\cos(n, \omega)$ is the cosine of the angle between the surface normal n and direction ω and accounts for the projected area of the surface element.

The resulting radiance $L(x, \omega_r)$ is power over projected area and solid angle, its unit is $[Wm^{-2}sr^{-1}]$, with sr being *steradian*, the unit of solid angle. To compute the resulting power arriving at the receiver, we need to integrate $L(x, \omega_r)$ over the (projected) surface element $\cos(n_x, \omega_r)\mathrm{d}A_x$, as well as the solid angle of the receiving antenna. As before, we model the receiving antenna as a surface with area A_r, computed from the antenna gain (2), resulting in a solid angle of A_r/d_r^2. Thus, the integrated power is

$$\begin{aligned} P_r &= L(x, \omega_r)\ \underbrace{A_r/d_r^2}_{\text{solid angle}}\ \underbrace{\cos(n_x, \omega_r)\mathrm{d}A_x}_{\text{projected area}} \\ &= P_t \frac{G_t G_r \lambda^2 \sigma(\mathrm{d}A_x)}{(4\pi)^3 d_t^2 d_r^2}, \end{aligned} \tag{5}$$

which is again equivalent to the radar equation from (1). According to this derivation, the radar cross section of the surface element is

$$\sigma(\mathrm{d}A_x) = \cos(n_x, \omega_t) 4\pi f(\omega_t, x, \omega_r)\cos(n_x, \omega_r)\mathrm{d}A_x\ [m^2]. \tag{6}$$

The 4π in this equation can be explained by the definition of BRDFs: the unit of a BRDF is sr^{-1}. Consequently, a BRDF that uniformly reflects all incident light into the sphere of all directions has value $\frac{1}{4\pi}$. With the factor of 4π in (6), this is normalized to one.

Now, to render a scene object S, we simply integrate (3)–(6) over all differential area elements $\mathrm{d}A_x \in S$,

$$P_r = \frac{P_t \lambda^2}{(4\pi)^3} \int_S \frac{G_t(\omega_t) G_r(\omega_r) \sigma(\mathrm{d}A_x) V(\mathrm{d}A_x)}{d_t^2 d_r^2} \mathrm{d}A_x\ [W]. \tag{7}$$

The additional visibility term $V(\mathrm{d}A_x)$ nullifies transport between surface elements that can not be seen from either the sending or receiving antenna, such as the back of the scene object in Fig. 1 left, and thus ensures valid integration. While all constants can be pulled from the integral, we need to retain the distances d_t, d_r and antenna gains G_t, G_r as integrands, as they change with respect to each individual area element. Note that (7) is limited to direct reflection without multipath effects. Leveraging Monte Carlo integration, we can numerically solve the integral $\int_S$ using a large number of stochastically sampled and ray-traced intersections with the scene object S

$$P_r = \frac{P_t \lambda^2 A_S}{(4\pi)^3 N} \sum_k^N \frac{G_t(\omega_k) G_r(\omega_r) \sigma(\mathrm{d}A_k) V(\mathrm{d}A_k)}{\rho(\omega_k) d_t^2 d_r^2}\ [W], \tag{8}$$

where N is the number of discrete samples, A_S the aggregate surface area of S, and $\mathrm{d}A_k$ the area element intersected by the k-th ray path. To improve convergence, we use importance sampling [35]: rather than selecting samples uniformly, more sample directions ω_k are drawn from regions expected to contribute more to the signal, expressed as a probability density function (PDF) over all directions. While such a PDF $\rho(\omega_k)$ can be freely chosen as long as samples are weighted by $\frac{1}{\rho(\omega_k)}$, convergence improves as $\rho(\omega_k)$ approximates the true energy distribution, which in our case is related to the antenna gain. Accordingly, we choose ρ (by definition normalized to one) proportional to G_t, thus more densely sampling rays in directions with large antenna gain, effectively mitigating variance due to G_t. Note that, since we sample outgoing ray paths according to $\rho(\omega_k)$, we never select rays with zero probability, and thus never divide by zero.

Using (8), our radar renderer (RR) computes P_r for all individual antenna pairs Tx_i and Rx_j in the antenna array. To this end, we integrate the energy emitted from the transmitting antenna Tx_i, reflected from all surface elements $\mathrm{d}A_k$, and received by the receiving antenna Rx_j

$$RR(S) = P_r(Tx_i, Rx_j)\ \ \forall i \in N_{Tx} \wedge \forall j \in N_{Rx}\ [W], \tag{9}$$

where N_{Tx} is the number of transmitting and N_{Rx} the number of receiving antennas in the array, respectively. This allows us to simulate the returns of a mmWave MIMO radar in the near field while considering both macroscopic (geometry) and microscopic surface detail (material).

3) IF SIGNAL GENERATION

Finally, to generate the intermediate frequency (IF) signal for a given time step t and a pair of transmitting and receiving antennas Tx_i and Rx_j, we accumulate the contributions from all individual ray paths during radar rendering.

$$IF(t) = \sum_k^N A_k(Tx_i, \mathrm{d}A_k, Rx_j)\ \exp(-2\pi j\ f_c(t)\tau_k), \tag{10}$$

where A_k is the signal amplitude of the k-th ray path between Tx_i, $\mathrm{d}A_k$, and Rx_j, which was sampled during radar rendering and can be approximated from the incident power at the receiving antenna using $A \approx \sqrt{P}$. j is the imaginary unit, $f_c(t)$ is the carrier frequency at the current time step t, and τ_k is the time of flight of the k-th ray path. Note that $f_c(t)$ may evaluate differently based on the type of radar modeled, for example continuously for a frequency-modulated continuous-wave (FMCW), or discretely for a frequency-stepped continuous-wave (FSCW) radar.

IV. INVERSE RADAR RENDERING

While the goal of traditional (*forward*) rendering is to synthesize 2D images from given 3D scene parameters such as meshes and materials, inverse (*backward*) rendering instead aims to reconstruct scene parameters from given image observations. Inverse rendering relies on the same gradient backpropagation [36] technique as deep learning-based methods, where neural networks are commonly employed to solve tasks in a data-driven approach, albeit replacing the neural network with a renderer. This enables data-driven inference of 3D scene parameters, such as geometry or materials, from 2D images where the gradients are derived from. Analogously to most deep learning frameworks, such as PyTorch [37] and TensorFlow [38], differentiable rendering can be achieved using automatic differentiation (*AD*). AD can differentiate arbitrarily complex functions, given they are composed of multiple simple (differentiable) operations such as addition and multiplication. By coupling the derivatives of each differentiable operation using the chain rule, the (partial) derivatives of any function with respect to its parameters can be computed.

We now apply this concept in a radar simulation context to infer properties of a known 3D scene under microwave radiation using inverse rendering. Thus, we optimize for

$$\theta^* = \arg\min_{\theta \in \Theta} L(IF(RR(\theta)), y), \tag{11}$$

where L is a loss function defining the optimization criterion with respect to the ground truth target y. RR and IF are the radar rendering and intermediate signal generation steps as described in Section III, and θ^* the optimal set of parameters θ from the superset of all possible parameters Θ. The parameter set with the lowest penalty, i.e., deviation from the ground truth as defined by the loss function, is then iteratively optimized for using gradient descent.

To compute the gradients, we store the intermediate results of each computation step during radar simulation.

$$\alpha_i = RR(\theta_i), \;\; \beta_i = IF(\alpha_i), \;\; \gamma_i = L(\beta_i, y) \tag{12}$$

Next, we compute the partial derivatives with respect to the inputs of each step in the reverse direction, starting from the loss and going back towards the original input parameter θ_i.

$$\begin{aligned} \frac{\partial \gamma_i}{\partial \beta_i} &= \frac{\partial L(\beta_i, y)}{\partial \beta_i} \\ \frac{\partial \beta_i}{\partial \alpha_i} &= \frac{\partial IF(\alpha_i)}{\partial \alpha_i} \\ \frac{\partial \alpha_i}{\partial \theta_i} &= \frac{\partial RR(\theta_i)}{\partial \theta_i} \end{aligned} \tag{13}$$

Finally, the gradient of the input parameter θ_i with respect to the loss can be determined using the chain rule.

$$\frac{\partial \gamma_i}{\partial \theta_i} = \frac{\partial \gamma_i}{\partial \beta_i} \frac{\partial \beta_i}{\partial \alpha_i} \frac{\partial \alpha_i}{\partial \theta_i} \tag{14}$$

This is called gradient backpropagation, as the direction of computations is backwards compared to the regular evaluation order. This enables us to iteratively optimize for any scene parameters, using the update rule $\nabla\theta_i = -\epsilon \frac{\partial L}{\partial \theta_i}$, where ϵ is called the learning rate, and $\frac{\partial L}{\partial \theta_i}$ are the first-order gradients of θ_i with respect to the loss function L.

Automatic differentiation, however, only yields gradients for continuous functions. This is fortunately the case for all functions in our use-case, except for the visibility term V in (7) and (8). While there exist techniques to differentiate discontinuous functions, for example by change of variables [26], we failed to see a notable benefit for our use-case. The inherent setup of a radar, where emitters and receivers are co-located, strongly attenuates the impact of a visibility term, especially when not considering multi-bounce effects during rendering. This allows us to treat the visibility term as a constant during backpropagation, and thus significantly simplify the implementation without loss of generality.

V. ALGORITHM

The goal of our inverse radar simulation pipeline is to infer information about a scene from real radar data. In theory, we could optimize for arbitrary parameters, such as scene geometry, material parameters, or antenna configurations. In this paper, however, we assume that the geometry of a scene is known – in fact it is reconstructed independently using photogrammetry – and we optimize only for the material parameters of the scene, i.e., parameters of a model that describes how surface points reflect incident radar waves. To this end, we iteratively adjust our simulation to produce output as close as possible to a real radar, with the aim to reconstruct material properties which best explain the observations of the real mmWave radar.

As input, we require scene geometry, a material model and regularization, a description of the radar to be simulated, and ground truth radar returns. Scene geometry and ground truth data is provided by the MAROON [39] dataset, which we further detail in Section VI. By tracing rays from each transmitting antenna into the scene, which are then reflected and returned to the receiving antennas, we compute the IF signal of the simulated radar system. The output of our simulation is of identical format to the real radar, i.e., an array of complex numbers depending on the antenna layout of the radar, which can be used interchangeably in signal processing algorithms. A high-level overview of our proposed inverse radar simulation pipeline is given in Fig. 2.

Since MAROON [39] provides mesh geometry using multiview stereo (MVS), some residual error from the domain gap between optical and radar sensors, and a small registration error [40], are to be expected. Note that our approach is not limited to MVS and supports any source of mesh geometry. A perfect match between the outputs of a real radar and a radar simulator should not be expected either, due to the inherent complexity of the electromagnetic processes involved. Nevertheless, we demonstrate that our optimization yields simulated returns that very closely match a real radar.

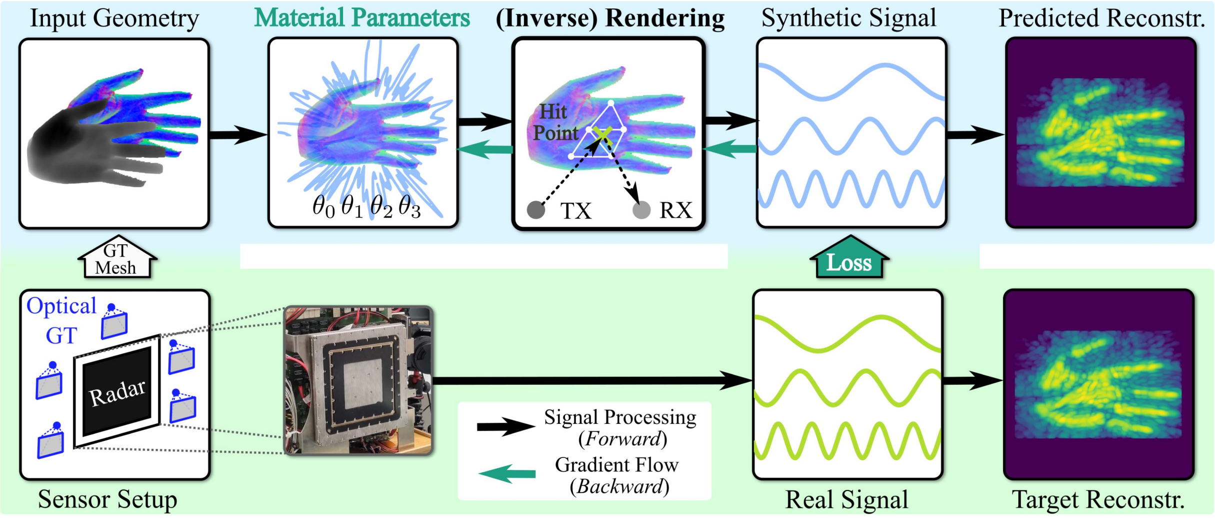


FIGURE 2. Our inverse rendering pipeline for near-field mmWave MIMO radar, where we iteratively optimize material properties until the simulated phasor data matches the real data as closely as possible. This way, we can directly infer material properties of known scene geometry under microwave radiation from observations of a real radar, which in turn enables further down-stream tasks such as object recognition and material characterization.

A. SIMULATED RADAR SETUP

We rely on information provided by the radar manufacturer and electromagnetic field simulation software to compute the antenna characteristics based on CAD models of the antenna array. During ray tracing, we consider the positions of the transmitting and receiving antennas, carrier frequency, bandwidth, transmission losses and directional antenna gains.

1) RESIDUAL RADAR GAIN

To handle any unknown or difficult-to-model effects, such as losses on transmission lines and antenna-specific impedance, we introduce a residual gain value per radar, which we optimize alongside any other scene parameters. This way, we enable the optimization process to more closely match the magnitudes of the signal measured by the real radar. We initialize this residual gain factor to 6 dB, which we empirically found to be a good starting point for most scenes we experimented on. However, it is ambiguous to the optimization process whether the material of the scene object should reflect more energy, or the residual gain value should be increased. Resolving ambiguities, such as disentangling reflectance and gain, is an ill-posed problem that is still actively researched in the context of inverse rendering [41].

2) ANTENNA CHARACTERISTICS

Leveraging CST Studio Suite, we pre-compute antenna characteristics for the transmitting and receiving antennas respectively. During rendering, we read the directional antenna gains $G_t(\omega_i)$ and $G_r(\omega_o)$ from the resulting look-up tables and plug them into (8). We rely on these pre-computed characteristics when shooting rays from TX antennas to focus more rays towards areas with higher output gain and thus focus computations on well illuminated areas.

3) FINE REGISTRATION

Since the fidelity of our radar simulation directly depends on the accuracy of the virtual scene representation, even a small registration error may strongly interfere with the results. Especially phase information is strongly dependent on the accuracy of the registration, where even sub-millimeter errors may cause significant phase shifts in the resulting signal. Thus, we introduce a per-scene correction term in the range of $\pm\frac{\lambda}{2}$ along each principal axis to compensate for any registration errors, for example introduced by the photometric reconstruction. This correction term is initialized to zero and optimized analogous to any other parameters. We limit this offset to $|\frac{\lambda}{2}|$, where λ is set to the longest wavelength employed by the radar, to avoid large offsets from exploding gradients and local minima in the optimization process due to the recurrent nature of waves. The MAROON [39] dataset reports registration errors up to 1–2 millimeters, which is well handled by our correction term.

B. MATERIAL

Once a ray hits an object in the scene, the material defines how much energy is either reflected or absorbed. In our system, the material consists of two components: the material model, which yields the reflection properties of a surface, and the regularization, which yields how the parameters of a model may vary spatially or by frequency. In the following, we introduce multiple variants for each, which we later evaluate in Section VII.

1) MATERIAL MODELS

We tested five different material models of varying complexity, which can all interchangeably be substituted into f in (6). We informally include equations for each model to provide the reader an outline of each model's complexity. There is no need to follow every minute detail of these computer graphics inspired models, but we refer any interested readers to the physically based rendering (PBR) [14] book. Instead, our goal is to focus on the conceptual differences between the models we evaluated, for example how the reconstruction quality changes with the addition of rough specular reflection versus purely diffuse scattering, or when incorporating complex indices of refraction.

2) f_0: Baseline: As a baseline, we employ a simple diffuse material model, whose sole reflectivity parameter $\theta_0 \in [0, 1]$ determines the scalar reflectivity of a surface. This parameter can either be interpreted as albedo in computer graphics terms, or as (normalized) radar cross section in radar terms.

$$f_0(\omega_t, \omega_r) = \theta_0 \tag{15}$$

3) f_1: Mixed Phong: We adapt the idea from Schüßler at al. [12], where a linear blending factor $\theta_1 \in [0, 1]$ determines how diffuse or specular an object appears under microwave radiation. They employ linear interpolation between diffuse and specular mirror reflection. Perfect mirror reflection, however, is a Dirac delta function which has a singularity at the reflection direction and is zero everywhere else. As we simultaneously connect each ray-traced hit point with all transmitting and receiving antennas during radar simulation, it would require the receiving antenna to perfectly align with the reflection direction, which is statistically impossible for a Dirac delta function. Thus, we employ the specular Blinn-Phong [42] reflection model $\mathcal{P}$ instead, which introduces the exponent $\theta_0 \in [1, 100]$ as a parameter. This effectively narrows or widens the angle around the peak at the perfect reflection direction, and thus allows for transport over a wider range of angles. Note that we normalize both diffuse and specular reflection terms to obey energy conservation and $\mathcal{I}(x, y, a)$ implements linear interpolation between x and y.

$$f_1(\omega_t, \omega_r) = \mathcal{I}\left(\frac{1}{\pi}, \mathcal{P}(\omega_t, \omega_r, \theta_0)\frac{(\theta_0 + 1)}{2\pi}, \theta_1\right) \tag{16}$$

4) f_2: Layered BRDF: This model employs a bidirectional reflectance distribution function (BRDF), similar to physically based rendering (PBR) [14] models in computer graphics. This BRDF consists of two layers, a specular coating layer and a diffuse base layer with albedo $\theta_0 \in [0, 1]$. The (real-valued) Fresnel term $\mathcal{F}_r$, parameterized by the (real-valued) index of refraction $\theta_1 \in [1, 10]$, determines the fraction of photons which reflect from each layer, where reflectance corresponds to the specular coating layer and transmittance corresponds to the diffuse base layer. Specular reflection is modeled using the GGX microfacet distribution $\mathcal{D}_{\text{GGX}}$ [43] and Smith shadowing term $\mathcal{G}_1$ [44], which are parameterized by roughness $\theta_2 \in [0, 1]$ and allow for both mirror-like and glossy reflection. As before, we blend between diffuse and specular reflection using $\theta_3 \in [0, 1]$.

$$\begin{aligned} F &= \mathcal{F}_r(\cos(\omega_t), \theta_1) \\ D &= \mathcal{D}_{\text{GGX}}(\omega_h, \theta_2) \\ G &= \mathcal{G}_1(\omega_r, \theta_2)\mathcal{G}_1(\omega_t, \theta_2) \\ f_2(\omega_t, \omega_r) &= \mathcal{I}\left(\frac{\theta_0}{\pi}, \frac{FDG}{4\cos(\omega_r)\cos(\omega_t)}, \theta_3\right) \end{aligned} \tag{17}$$

5) f_3: Fresnel Smooth: While most visible light can safely be assumed to be incoherently polarized, polarization plays a more crucial role in the context of microwave radiation, as most hardware emits coherently polarized waves. Thus, we employ a material model with complex indices of refraction, composed of both a real part $\theta_0 \in [1, 100]$ (permittivity) and complex $\theta_1 \in [0, 100]$ part (conductivity). The horizontal and vertical polarization components are handled separately via the the two Fresnel terms $\mathcal{F}_s$ and $\mathcal{F}_p$, and a linear combination of both is returned using $\theta_3 \in [0, 1]$. This removes the need for an explicit albedo parameter, since absorption is implicitly modeled via complex arithmetic. Additionally, Brewster's angles are introduced, where the vertical polarization is fully refracted and all reflected energy is horizontally polarized. Note that this material model assumes dielectric interfaces and smooth surfaces under microwave radiation.

$$\begin{aligned} n_i &= \theta_0 - \theta_1 i \\ F_s &= \mathcal{F}_s(\cos(\omega_t), n_i) \\ F_p &= \mathcal{F}_p(\cos(\omega_t), n_i) \\ f_3(\omega_t, \omega_r) &= \mathcal{I}(F_s, F_p, \theta_3) \end{aligned} \tag{18}$$

6) f_4: Fresnel Rough: Finally, we aim to relax the smooth surface requirement of the previous Fresnel-based material model, by re-introducing a GGX microfacet lobe to incorporate both smooth and rough structures on the surface into the model. Similarly, we omit an albedo parameter with this material model, since absorption is implicitly modeled by the Fresnel equations. We limit the roughness parameter $\theta_2 \in [0, 1]$ to modulate the amplitude of the reflected waves and ignore phase differences from variable microfacet heights, since we noticed severe overfitting when introducing phase delays in the material model. Similarly, a linear combination of the horizontal and vertical polarization component is returned via $\theta_3 \in [0, 1]$.

$$f_4(\omega_t, \omega_r) = \frac{\mathcal{I}(F_s, F_p, \theta_3)DG}{4\cos(\omega_r)\cos(\omega_t)} \tag{19}$$

2) MATERIAL PARAMETER REGULARIZATION

Next, we address how to represent and regularize the individual parameters of the material models. The main question is whether to allow for the parameters to vary spatially, and how, or to only optimize for a single set of parameters per scene. In the following, we introduce four variants: *Global*, *Voxelgrid*, *Hashgrid* and *Vertex*. Independent of the spatial regularization, we linearly interpolate all material parameters across the emitted wavelengths to cover frequency-dependent effects.

2) Global: The most straightforward, but at the same time strongest regularization, is to optimize for a single set of material parameters per scene. This limits the capabilities of the model and assumes the material in the scene to be spatially constant, which may not suffice for more complex materials with spatially varying features, such as the grain pattern in wood, for example. Overfitting, however, is nearly impossible due to gradients being averaged over the whole scene.

2) Voxelgrid: Alternatively, we subdivide the scene into a uniform grid of 3D voxels and store material parameters at each voxel's center. This allows for the parameters to vary spatially without any assumption of scene geometry and ensures constant spacing between parameters. The coarseness of the subdivision scheme trades between spatial resolution and memory requirements and we perform tri-linear interpolation to ensure smooth and continuous transitions.

2) Hashgrid: A popular choice of underlying scene representation in the context of novel view synthesis [17] is a hash-based grid encoding combined with a multilayer perceptron (MLP) [18]. The multi-resolution hash-based encoding scheme replaces the fixed mapping between points in space and voxels, which automatically adapts to the underlying geometry and simultaneously encodes material parameters at multiple spatial resolutions, which allows for crosstalk between parameters at different scales. The hierarchical encoding scheme combined with an MLP allows for a compact representation, albeit at the cost of high-frequency noise due to hash collisions and high computational cost per parameter lookup.

2) Vertex: Finally, we opted to store material parameters with the scene geometry, specifically at each triangle vertex. This approach couples the spatial resolution, and thus regularization, of the material parameters with the geometry, which is the resolution limit of the input data. The main drawback, however, is the direct dependency on tessellation, as the number of stored parameters and their spacing fully depends on the mesh. To circumvent tessellation impacting our experiments, we pre-process each scene to equalize the tessellation level, which will be further discussed in Section VI-B. We leverage the barycentric coordinates computed by ray tracing hardware to smoothly interpolate between the material parameters at the vertices of the intersected triangle.

3) NORMAL MAP

Since every material model, except for the *Baseline* model, strongly depends on the surface normal, we optionally also optimize for a normal map. This normal map stores a small offset to the normal at each vertex of the mesh to locally fix inaccuracies in the MVS reconstruction. We initialize these offsets to zero and optimize them alongside all other material parameters, if enabled.

C. LOSS FUNCTION

The choice of loss function is critical for any gradient backpropagation algorithm, as it defines the optimization criterion where gradients are derived from. In the context of imaging radar, the obvious candidate for inverse rendering is to apply the loss function to the reconstructed (2D) radar images. However, we found that the maximum projection, which is often used to project the resulting 3D volume to 2D [4], does not yield viable gradients in practice, and causes the optimization to either converge slowly, or not at all. Thus, we instead propose to compute the loss on the full 3D volume, before any projection is applied. To this end, we compare the reconstructed voxels from the simulated and real radar data using the mean squared error

$$L_{2\text{-REC}} = \frac{\sum_i^N [REC(y)_i - REC(IF(RR(\theta)))_i]^2}{N}, \tag{20}$$

where $REC()$ is a 3D radar reconstruction algorithm, such as the Fourier transform or backprojection [4]. Note that when computing $L_{2\text{-REC}}$ on decibel values, the resulting volumes must use the same (absolute) reference or the optimization will diverge. While this improves convergence, we still found the optimization process to be volatile and the results unsatisfactory. Thus, we further propose to skip the radar reconstruction algorithm entirely and compute the loss directly from the IF signal, either using the L1 or L2 norm.

$$L_{1\text{-IF}} = \frac{\sum_i^N |y_i - IF(RR(\theta))_i|}{N}$$
$$L_{2\text{-IF}} = \frac{\sum_i^N [y_i - IF(RR(\theta))_i]^2}{N} \tag{21}$$

This approach, however, requires the simulated phasor to match the real radar data more closely, both in terms of amplitude and phase, as we directly compare (absolute) phasor values instead of (relative) correlation values from the radar reconstruction. The residual radar gain, as previously introduced in Section V-A, enables our simulation to closely match the real phasor. We provide a thorough evaluation of the choice of loss function for inverse radar rendering in Section VII.

D. IMPLEMENTATION

For this paper, we implemented a radar simulation framework from scratch. Each performance-critical part is executed on the GPU, which yields significant speed-ups compared to a purely CPU-based implementation, due to massive parallelism. For ease of use, the required (inverse) radar rendering kernels were integrated into a Python-based PyTorch [37] framework using custom extensions written in the CUDA [45] programming language. Furthermore, to realize automatic differentiation for large custom kernels, such as radar rendering or IF signal generation, we employ Slang-D [23], which is a shader pre-compilation tool that expands the feature set of common shading languages. They recently extended their tool with automatic differentiation support for CUDA kernels, which allow for fast iteration times during development of inverse rendering kernels, by omitting the need for explicit derivatives which are time-consuming and error prone to write by hand. A native implementation in Python using PyTorch would not have been feasible due to excessive storage requirements of intermediate states in the computational graph and

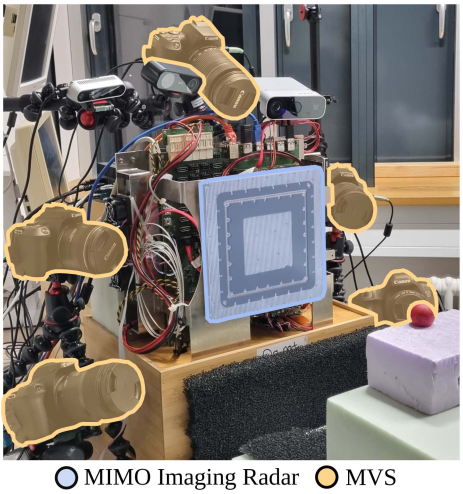


FIGURE 3. The MAROON dataset employs a multiple-input multiple-output (MIMO) imaging radar as well as a multi view stereo system (MVS), which we use to acquire ground truth information for our simulation. The image is taken from [39].

exceptionally high runtime. For fast ray tracing, we leverage OptiX [46], which provides high-performance ray intersection tests on the GPU while making use of modern ray tracing hardware. The code of our simulation framework is publicly available.[1]

VI. HARDWARE AND DATASET

While an evaluation of real-world use-cases is notably more difficult compared to synthetic examples, we demonstrate the applicability to real-world scenarios and base our experiments on the MAROON [39] dataset. This dataset provides optical ground truth reconstructions as well as sensor data collected from a MIMO imaging radar in conjunction with four optical depth sensors. Within the sensor variety, we particularly rely on the radar captures which were spatially calibrated with a ground truth MVS system of five DSLR cameras. The corresponding hardware setup is depicted in Fig. 3, of which further details are given in Section VI-A. Moreover, in Section VI-B we explain the details of the dataset, which are relevant for our experiments.

A. MIMO RADAR AND MVS

The employed MIMO radar is a submodule of a near-field imaging system from Rohde & Schwarz [47], as pictured in Fig. 3. The square-shaped antenna aperture consists of 94 transmitting (TX) and 94 receiving (RX) antennas, and the signal is sampled at 128 discrete frequencies in frequency stepped continuous wave (*FSCW*) manner. The frequency band reaches from 72 GHz to 82 GHz, i.e. 10 GHz of bandwidth, which results in a maximum unambiguous range of 1.9 m and a spatial resolution of $4 \times 4 \times 11$ mm at 30 cm depth [39]. Time-division multiplexing avoids signal interference and ensures that the signal is separated into its pairwise Tx-Rx antenna and respective frequency components.

While the dataset contains reconstructions from multiple sensor types, we solely employ the data of the MVS system, of which the five distinct viewpoints allow for accurate, high-resolution reconstructions of the observed objects.

B. DATASET

The MAROON dataset comprises 45 objects captured from five depth sensors, namely one imaging radar, three RGB-D cameras, and a ground-truth MVS system. Each object is recorded at three different distances to the radar sensor, of which we utilize the sensor data of nearest distance (30 cm).

1) SENSOR DATA

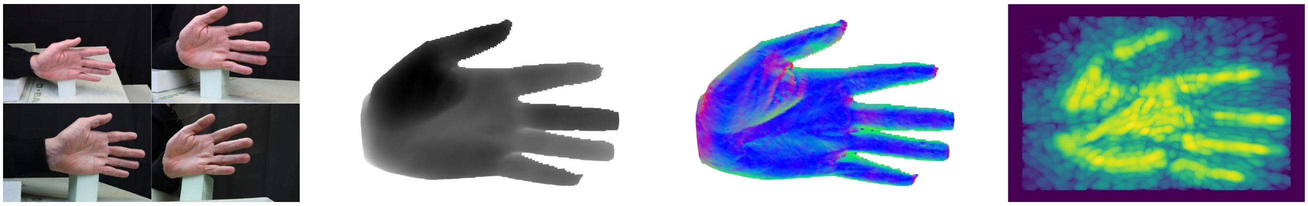
In accordance with the previously described antenna arrangement, the capture output of the MIMO imaging radar is a $94 \times 94 \times 128$ tensor of complex phasors. To reconstruct an object from the raw data, backprojection [1] can be applied as suggested by [39]. The result of this algorithm is a 3D volumetric representation in form of a $N \times M \times D$ voxel grid, where each entry contains a confidence value about the object's presence at the respective voxel center. By applying maximum projection [48], a dimensionality reduction is achieved such that the result can be visualized as an intensity-encoded 2D radar depth image, which is depicted in Fig. 4. Furthermore, the depth values are commonly filtered using a cutoff threshold in decibel. We explicitly note that the presented depth and radar images are only used for visualizing the presented data in our experiments. Instead, the gradients of the optimization are either computed from the complex tensor of phasors or from the 3D volumetric grid, as described in Section V-C.

For the MVS system, reconstructions are provided in form of a 3D triangle mesh. Further processing and sanitation of this mesh yields a high-resolution ground truth version of each object. After spatial alignment with the radar coordinate system, we get a 3D mesh that is placed in front of the radar aperture alongside the reconstructed radar signal. As mentioned for the vertex representation in Section V-B2, the tessellation level of the mesh is equalized across all object reconstructions. Similar to the evaluation in [39], we rasterize the spatially aligned MVS mesh with respect to the 2D radar pixel grid and, subsequently, triangulate the resulting depth map using the topology defined by the pixel neighborhood. In this way, the tessellation level is kept consistent with the sample resolution of the radar reconstruction.

2) EMPTY ROOM MEASUREMENT

A straightforward approach to reduce clutter and side-lobe artifacts in measurements from the real radar is to perform an empty room measurement. Due to our radar rendering framework being limited to a single bounce and the MVS reconstructions in the MAROON dataset only including the

[1] https://github.com/nihofm/inverse-radar-rendering



MVS Input Views (4/5) MVS Depth MVS Surface Normals MIMO Radar Image

FIGURE 4. We showcase the (tessellated) multi-view stereo (MVS) and radar data at 35 dB dynamic range from the `S1 Hand Open` scene, which we utilize as ground truth for our algorithm. The mesh and radar data is pre-processed and aligned, which allows us to directly input scenes from the MAROON dataset into our pipeline without any modifications, except for the additional tessellation to equalize the mesh for the *Vertex* regularization.

object in front of the radar, we do not have any means to simulate the multi-bounce returns and background clutter from the room the radar setup was placed in. Thus, we utilize the empty room measurements in MAROON that were made without any object in front of the radar. We then subtract this data from the measurement with an object in front of the radar. In this way, we emphasize the returns from the object and mask clutter returned from the room, which cannot be simulated given our framework and the dataset.

VII. RESULTS

In the following, we provide a comprehensive qualitative and quantitative evaluation of our inverse radar simulation framework on both synthetic and real-world data. For each of the 45 scenes in the MAROON dataset, we evaluated over 15 variants of our algorithm, performing 250 radar rendering and gradient update steps each, which summed up to approximately 1500 GPU hours across multiple NVIDIA A40 GPUs. Unless stated otherwise, we employed the following configuration. We perform ray tracing until 2^{18} stochastically sampled intersections with the scene were found, totaling $\approx 296 \times 10^9$ connections between transmitting and receiving antennas for each simulation step. Backprojection was performed with a sample resolution of $N = 128$, $M = 128$, and $D = \lceil \frac{D_S}{D_V} \rceil$ voxels, where D_S is the depth extent of the scene and D_V of a voxel, respectively. We visualize reconstructions with a dynamic range of 35 dB to also show clutter and sidelobes from the reconstruction in the intensity-encoded radar images. Optimizations were performed using the `L1-IF` loss function and the Adam [49] optimizer with a learning rate of 0.1 and betas of (0.5, 0.9).

A. SYNTHETIC VALIDATION

To validate our inverse radar rendering system and the computed gradients for correctness, we execute the optimization process on synthetic data generated within our own simulation software. In this setting, the target data matches the simulation perfectly and we should be able to reproduce the input parameters very accurately, given our system works correctly and we do not get stuck in local minima. Stil, a perfect reconstruction should not be expected due to noise from stochastic sampling, inherent noisiness of iterative gradient-based optimization and bias from the Adam optimizer. To this end, we load the geometry from the `S1 Hand Open` scene, randomly choose target material parameters, simulate the target radar data, reset all material parameters to different random values, and start the optimization process to try to recover the original set of parameters. We report results for both using a single material (*Global*) and a spatially varying material (*Vertex*) in Fig. 5. After 250 iterations, all runs achieved convergence with a final loss value of < 0.01, which supports our approach.

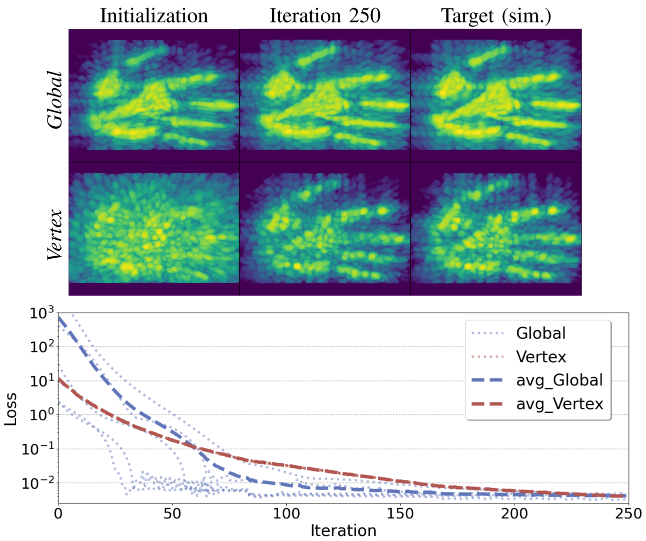


FIGURE 5. We validate our inverse radar rendering framework on synthetic data using the geometry from the `S1 Hand Open` scene, while using a single set of material parameters (Global) and spatially varying material parameters (Vertex). For each variant, we executed five optimization runs with fully randomized starting and target states. We show reconstructions at 35 dB dynamic range for two example runs (top) and loss curves for all optimizations (bottom). All runs converged with a final loss value of < 0.01, which supports our implementation. A loss value of exactly zero is not achievable in practice due stochastic noise in the sampling and optimization process.

B. QUALITATIVE EVALUATION

We now qualitatively evaluate our inverse radar rendering framework on real-world data using the MAROON [39] dataset. We showcase four selected scenes using the *Fresnel Rough* material model and *Vertex* regularization in Fig. 6

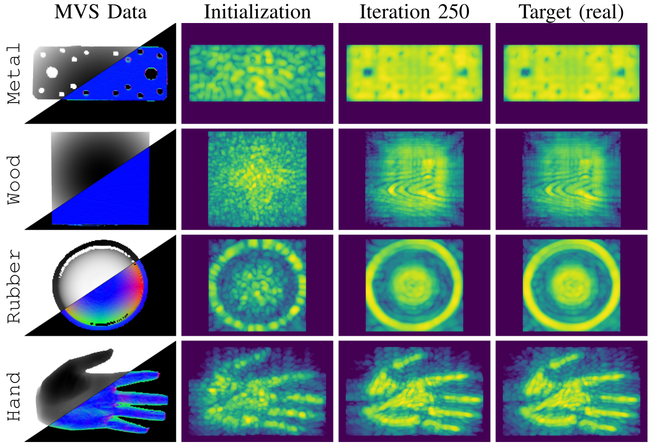


FIGURE 6. We showcase inverse radar rendering results at 35 dB dynamic range on four exemplary scenes: `V1 Metal Plate`, `Wood Plane`, `Plunger`, and `S1 Hand Open`. The first column shows the MVS input data (depth and normals) from the MAROON dataset, while the second and third columns show the simulated radar returns just after initialization (second column) and after inverse radar rendering (third column). The real radar returns are visualized in the fourth column.

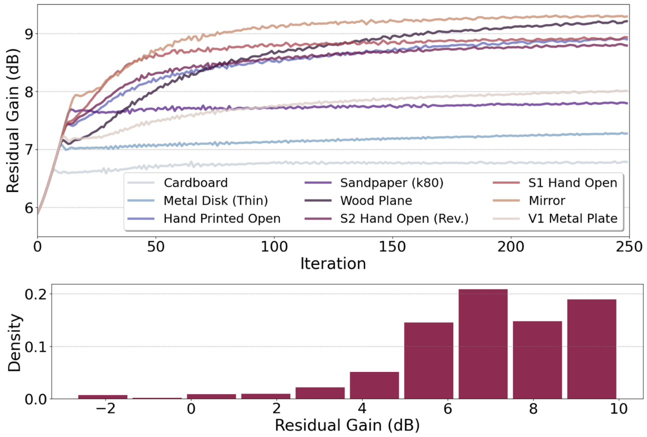


FIGURE 7. We visualize residual antenna gains for each step in the optimization process on nine example scenes from the MAROON dataset while using the *Fresnel Rough* material model and *Vertex* parameter regularization (top). We additionally plot a density histogram over the resulting residual gain values of $\approx$750 converged optimization runs across all scenes and variants we executed (bottom).

and further scenes in the supplemental material. While the radar images with randomly initialized parameters (Iteration 0) exhibit notable differences to the real-word counterpart, we show that our simulation is able to achieve very similar results after optimization (Iteration 250). For a multitude of scenes and materials, such as metal, wood, rubber, and a human hand as depicted in Fig. 6, the returns of our simulated radar closely match the real-world radar after optimization. Note that these images were created for visualization purposes only, as inverse rendering was performed on the IF signal directly instead of the reconstructed radar images, which yielded better results and will be further examined in Section VII-C. Thus, our inverse radar-rendering approach is both applicable to synthetic and real-word data.

In Fig. 7, we additionally evaluate the residual gain term, which was previously introduced in Section V, for a multitude of scenes and configurations. Except for a few outliers with particularly low gain, most optimization runs converged to a residual gain value in the range of 5 to 10 dB, as apparent from Fig. 7. We argue for the residual gain values to be of reasonable magnitude, as the estimated transmission losses of our employed MIMO radar are $\approx$ 6 dB.

Next, we visualize intermediate material parameters during two optimization runs with varying spatial regularization strength in Fig. 8, while utilizing the *Fresnel Rough* material model and *Voxelgrid* regularization. With all parameter regularization options that allow spatial variation, we generally noticed a trend of increased spatial variation in the material parameters with iteration count. While this may stem from the inherent complexity of the observed material, it may also stem from overfitting the model to the target data. In such cases, non-material related effects are also incorporated into the material properties due to the lazy nature of the optimization process. When trying to recover fine surface details, such as the grain pattern of the `Wood Plane` scene shown in Fig. 6, however, high spatial variation of the material properties is required.

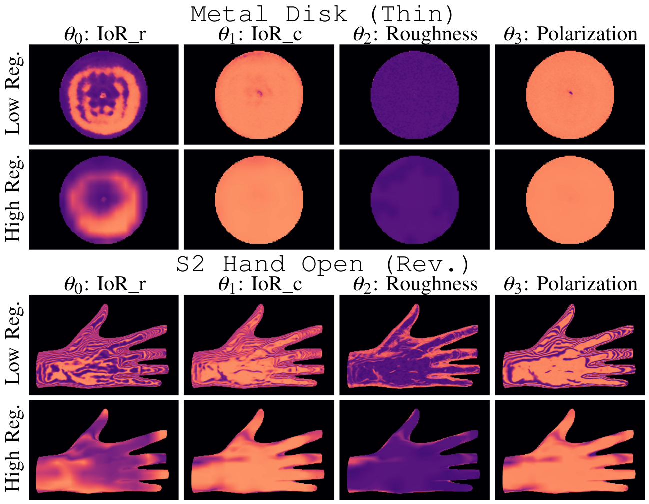


FIGURE 8. We visualize the effect of spatial regularization strength on the resulting material parameters from inverse rendering on two scenes from MAROON. The material model is *Fresnel Rough* with low and high spatial regularization modes using the *Voxelgrid* with $\approx 2e^6$ parameters for the low and $\approx 1e^3$ parameters for the high mode. The first two columns show the real and complex parts of the refractive indices, with their original range of $[0, 100]$ mapped to $[0, 1]$ for visualization purposes. The third and fourth columns show the roughness and polarization parameters with their range in $[0, 1]$. Note how the material parameters for the low regularization mode exhibit very strong spatial variation, which indicates overfitting to the target data, while the high regularization mode ensures spatially coherent parameters.

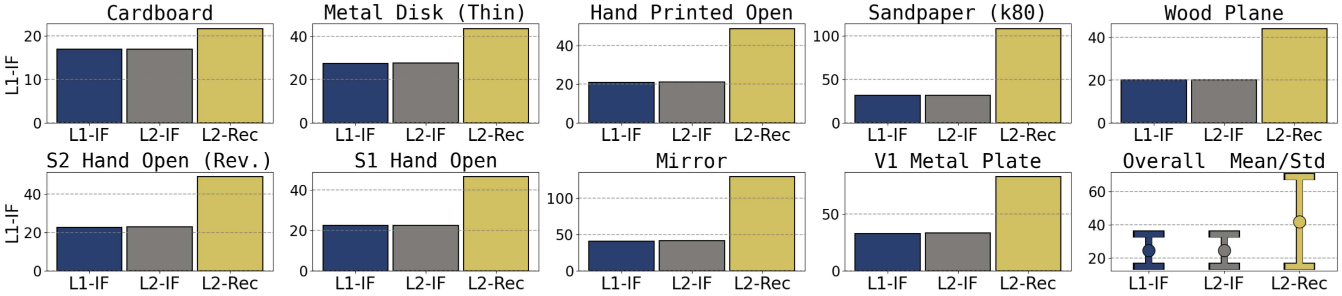


FIGURE 9. We quantitatively evaluate different optimization criteria on the MAROON dataset. We report `L1-IF` loss (lower is better) for nine representative scenes and report statistics over the overall dataset in the last panel. The ***L1-IF*** and ***L2-IF*** optimization criteria, which both operate on the intermediate frequency signal directly, did not exhibit any notable differences in the reconstruction fidelity in our tests. However, when additionally incorporating the radar reconstruction algorithm (***L2-Rec***), we observed a significant drop in reconstruction quality. Due to ambiguities in the signal reconstruction pipeline, and thus ambiguities in the gradients, the optimization regressed notably compared to the IF-based loss functions.

C. QUANTITATIVE EVALUATION: LOSS FUNCTIONS

To quantitatively evaluate the impact of the loss functions introduced in Section V on the optimization process, we executed our pipeline on all scenes in the MAROON dataset while using each loss function as optimization criterion. However, when the loss function is not only used during optimization, but also during evaluation, training on loss A while evaluating on loss A will yield better results than training on loss A and evaluating on loss B due to overfitting. For the remainder of this work, we report `L1-IF` loss as the main evaluation criterion, which makes the comparison between `L2-IF` and `L2-REC` metrics unbiased, but introduces a small bias for the `L1-IF` loss. Individual results for nine select scenes of the dataset are visualized in Fig. 9 and we tabularize the mean ($\overline{\mu}_{L1}$) and standard deviation ($\hat{\sigma}_{L1}$) of the `L1-IF` loss across all 45 scenes in MAROON here:

L1-IF	$L_{1\text{-IF}}$	$L_{2\text{-IF}}$	$L_{2\text{-REC}}$
$\overline{\mu}_{L1}$	**24.74**	24.83	42.06
$\hat{\sigma}_{L1}$	**9.65**	9.69	27.02

Both loss functions $L_{1\text{-IF}}$ and $L_{2\text{-IF}}$, which compare raw radar returns, robustly yield near-identical results of high quality and could be used interchangeably in our tests. When training on $L_{2\text{-REC}}$, however, which includes the FSCW reconstruction algorithm [48], we observed notably degraded results. We infer that optimizing on the reconstructed signal introduces twofold issues. Firstly, any artifacts introduced by the reconstruction algorithm are automatically incorporated into the optimization process, which negatively impacts reconstruction fidelity. Since the reconstruction assumes validity of the Born approximation [4] and there are numerous cases in practice where this requirement is not met, systematic artifacts are introduced, such as ambiguities and ringing, for example. This directly impacts the computed gradients for each iterative update, as they are immediately derived from the loss. Secondly, due to the relative nature of correlation, constructive or destructive interference and subsequent normalization, numerical stability becomes an issue and may cause vanishing or exploding gradients, which occurred after only a few epochs in our tests if not accounted for. Thus, we strongly argue for inverse radar rendering applications to directly optimize on the intermediate frequency signal instead of reconstructed images or volumes.

D. QUANTITATIVE EVALUATION: MATERIAL REGULARIZATION

To evaluate the effect of the spatial regularization variants from Section V-B2 on reconstruction quality, we ran our algorithm for each scene with constant configuration and varied the material parameter regularization. The only exception being the *Hashgrid* representation, which was trained with a lower learning rate of 0.01 instead to ensure convergence. We visualize the results in Fig. 10 and summarize the overall results for all 45 scenes in the MAROON dataset here:

L1-IF	Global	Voxelgrid	Hashgrid	Vertex
$\overline{\mu}_{L1}$	30.99	**24.68**	25.52	24.74
$\hat{\sigma}_{L1}$	15.08	**9.60**	9.88	9.65

In terms of loss, the *Voxelgrid* variant performed best, with the *Vertex* and *Hashgrid* variants closely behind. The *Global* variant performed worst in terms of `L1-IF` loss, which is to be expected given the comparably strong regularization versus the other variants. One can clearly see the effect of spatial variation in the material parameters, as the *Global* option is an outlier. For simple scenes which can be described using only a single set of material parameters, which is largely the case for the MAROON dataset, we consider the *Global* to be the most useful regularization for real-world applications due to its high interpretability and low susceptibility to overfitting. While any variant with spatially varying material parameters achieves lower overall loss, we observe strong and repetitive fluctuations in the resulting material properties for areas which are not prominently backscattering under microwave radiation and thus contribute less to the overall signal. This indicates overfitting, where the optimization only strives to match the target IF signal without considering physical plausibility, which is not particularly useful in practice. Thus,

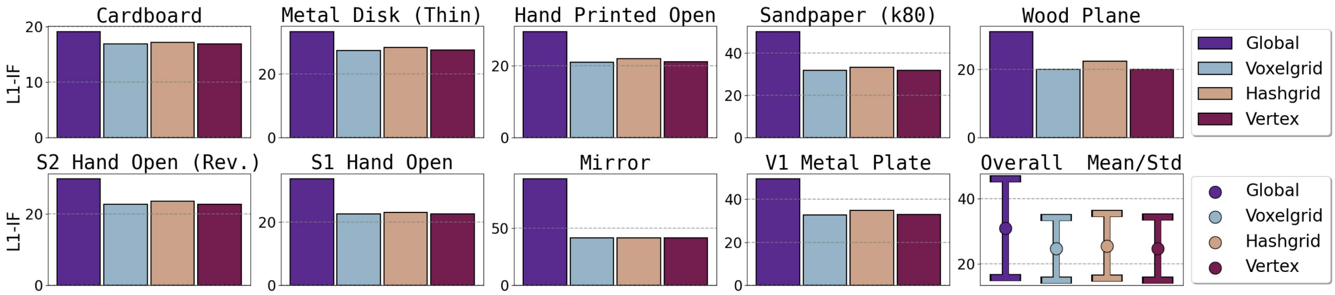


FIGURE 10. We quantitatively evaluate different material parameter regularization techniques on the MAROON dataset. We report `L1-IF` loss (lower is better) for nine representative scenes and report statistics over the overall dataset in the last panel. A detailed overview over all scenes is provided in the supplemental material. All variants which allow for spatially varying parameters (*Voxelgrid*, *Hashgrid*, and *Vertex*) performed relatively similar in terms of the reconstruction loss, with the *Voxelgrid* option performing best. The *Global* option, which uses a single set of parameters, performed notably worse for all scenes; however, in practice this option might be a very valuable option due to the strong regularization and low susceptibility to overfitting.

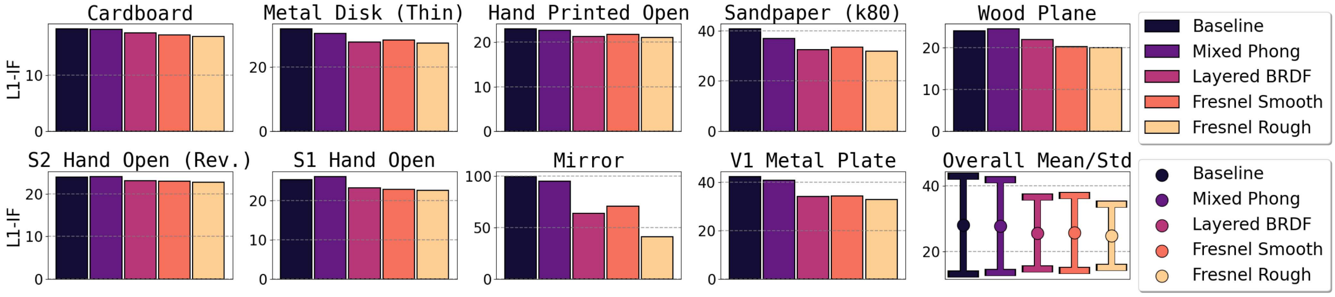


FIGURE 11. We quantitatively evaluate different material models on the MAROON dataset. We report `L1-IF` loss (lower is better) for nine representative scenes and report statistics over the overall dataset in the bottom right panel. A detailed overview over all scenes is provided in the supplemental material. While the *Baseline* and *Mixed Phong* models performed worst, the *Layered BRDF* and *Fresnel Smooth* yielded very similar reconstruction fidelity, and the *Fresnel Rough* produced the best reconstruction quality in terms of `L1-IF` loss compared to returns of a real radar.

we advise to use the strongest regularization applicable, either via a single set of parameters if the scene permits, or to adaptively penalize spatial variation in the resulting material parameters. Determining the ideal regularization term, however, is a scene-dependent trade-off and requires further research.

E. QUANTITATIVE EVALUATION: MATERIAL MODELS

To evaluate the effect of the material models from Section V-B1, we executed our algorithm with constant configuration for each scene of the MAROON dataset and only varied the underlying material model. The results are visualized in Fig. 11 and summarized for all 45 scenes in the following:

`L1-IF`	Baseline	Phong	BRDF	Fr. Smth.	Fr. Rgh.
$\overline{\mu}_{L1}$	27.98	27.74	25.62	25.62	**24.74**
$\hat{\sigma}_{L1}$	14.87	14.11	10.97	11.42	**9.65**

While the conceptually simple *Baseline* material performed worst overall, the *Mixed Phong* material model did improve over the baseline in terms of average `L1-IF` loss. A notable improvement over the Phong-based mixture model was achieved by the computer graphics-inspired *Layered BRDF* model, which first introduced both a roughness term for non-smooth surfaces and (real-valued) Fresnel term for dielectrics. The *Fresnel Smooth* material model, which further expands on the Fresnel term while assuming smooth surfaces, achieved very similar results compared to the *Layered BRDF* model. The *Fresnel Rough* model, which extends the *Fresnel Smooth* model by a microfacet-based roughness term to incorporate rough surfaces, performed best overall in our experiments. This indicates that the smooth surface approximation for Fresnel-based transport does not always hold true in practice.

It should be noted that in the context of gradient backpropagation, optimization algorithms always perform lazy, and their only aim is to reduce the loss without adhering to physical constraints. Thus, when giving a model more freedom, which is usually equivalent to an increase in parameters, it may reduce the overall loss without any practical advantage, i.e., overfitting the model to the data. While further research is needed to confirm the physical plausibility of our material models and parameters, we argue that the previously shown improvements over a wide range of scenes and materials

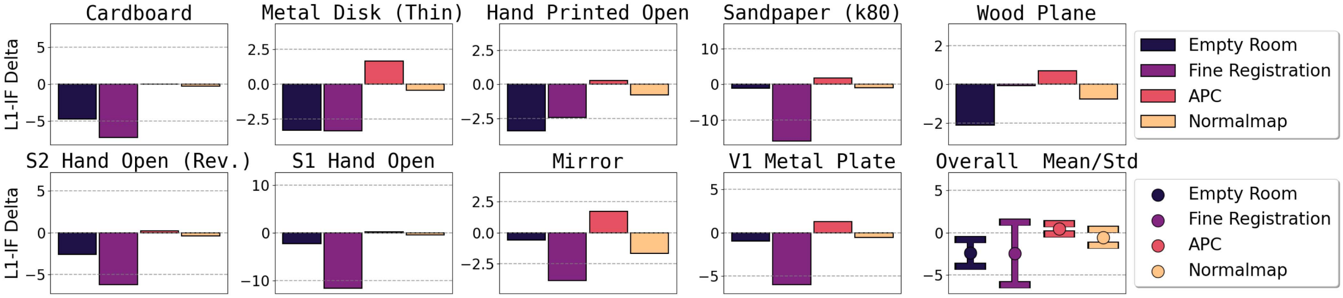


FIGURE 12. **We performed ablation studies regarding multiple variants of our inverse radar rendering framework on the MAROON dataset. We report differences in `L1-IF` loss (lower is better) with respect to a baseline for nine representative scenes and report statistics across the full dataset in the last panel. A detailed overview over all scenes is provided in the supplemental material. The additionally optimized *Fine Registration* offset, which compensates for a small registration error in the scene, yielded the largest overall benefit in our tests. For scenes with very accurate registration, however, such as `Wood Plane`, this feature did not provide much benefit. Secondly, performing an empty room measurement to reduce background clutter (*Empty Room*), also turned out be a very valid approach, with every scene either improving or at least staying consistent. For the *APC* and *Normalmap* variants, however, we did not see as consistently improved results, where the *APC* option even regressed reconstruction quality for some scenes.**

provide a valuable stepping stone towards practical material property inference for mmWave radar using inverse rendering.

F. ABLATION STUDIES

We further performed multiple ablation studies of our system with respect to empty-room measurements, fine registration, directional antenna power characteristic (*APC*), and normal map optimization. To this end, we evaluate each scene of the MAROON dataset with and without the feature in question to investigate the effect on reconstruction fidelity. We used the *Global* material regularization for the *Empty Room Measurement*, *Fine Registration*, and *APC* tests to reduce overfitting to a minimum and single out the effect of the respective feature. For the *Normal Map* variant, which is inherently spatially varying, we used the *Vertex* material regularization as a baseline instead to isolate its effect. We visualize results in Fig. 12 and tabularize the average `L1-IF` loss μ_{L1} without and with the respective features in the following.

`L1−IF`	Room	Fine Reg.	APC	Norm. Map
$\overline{\mu}_{L1}$ *w/o*	30.99	30.99	**30.99**	24.74
$\overline{\mu}_{L1}$ *w/*	**28.64**	**28.57**	31.45	**24.19**
$\Delta\overline{\mu}_{L1}$	-2.36	-2.43	+0.45	-0.55

1) EMPTY ROOM MEASUREMENT

We observed a reduction of the average loss across all scenes from 30.99 to 28.64 when optimizing against a target where the empty room measurement has been subtracted from, which corresponds to an improvement of ≈7.6%.

2) FINE REGISTRATION

When additionally optimizing for a small registration offset, as described in Section V-A, we observed a reduction of the average loss across all scenes from 30.99 to 28.57, which corresponds to the largest observed improvement of ≈ 7.8%.

3) ANTENNA POWER CHARACTERISTIC

When considering the pre-computed APCs during radar simulation, we noted improvements in some scenes and deterioration in others, which resulted in a regression of the average loss by ≈1.5% overall. This might be due to all scenes in MAROON consisting of objects directly in front of the radar, which are all covered by the main lobe of the antennas, and thus reduce the effectiveness of the APC. When additionally simulating returns of the room with multi-bounce reflections, for example, we expect a larger impact of the antenna power characteristics due to more ray paths actually covering the side lobes.

4) NORMAL MAP

When additionally optimizing a normal map, as described in Section V-B2, we report a small reduction of the overall average loss for all scenes from 24.74 to 24.19, which corresponds to an improvement of ≈2.2%. Note that this experiment was conducted while allowing for spatially varying material parameters to isolate the effect of the normal map, which is spatially varying by design.

VIII. CONCLUSION

We conclude that our inverse near-field MIMO radar rendering framework, which we formally derived and implemented on the GPU, is able to reconstruct material parameters from the phasor data of both synthetic and real radar datasets. We outlined any extra steps required to realize such a system with real radar data in practice, such as optimizing for residual gain and a registration offset. Our comprehensive evaluation showed that directly deriving the loss from the phasor data is a clear improvement compared to incorporating reconstruction algorithms, such as backprojection. Furthermore, the introduced material models and regularization options notably improved reconstruction fidelity in terms of loss compared to a baseline material implementation. Finally, we hope that our findings prove to be an important stepping stone towards inverse radar rendering for near-field MIMO radar and enable a multitude of downstream tasks, such as object recognition and material characterization.

VIII. LIMITATIONS AND FUTURE WORK

Naturally, our system comes with its own set of limitations. Due to the introduction of a residual gain value and the lazy nature of the optimization, ambiguities between scattering behaviour and residual gain can not be fully resolved. We also noticed severe overfitting of material models with spatially varying parameters and significant degrees of freedom with respect to phase information, such as phase delays from microfacet distributions, which requires further research w.r.t. parameter regularization. Furthermore, we utilized a simple polarization model, and disabled both visibility gradients and multi-bounce effects in our renderer, which reduces the complexity of the implementation and keeps the number of optimized parameters in check, but may be improved upon in the future. Since we limit our fine registration offset to $\pm\frac{\lambda}{2}$, the employed dataset must obey to this registration error, which fortunately is the case for the MAROON dataset. Finally, reconstructing both geometry and materials from phasor data poses an interesting area of future research. Initial experiments, however, suggest this avenue to pose a major challenge with a geometrical optics approach, due to highly irregular loss landscapes generated by waves reflected from infinitesimally small point targets.

ACKNOWLEDGMENT

The authors would like to thank the Rohde & Schwarz GmbH & Co. KG (Munich, Germany) for providing the radar imaging device and technical support which made this work possible.

REFERENCES

[1] S. S. Ahmed, "Microwave imaging in security—two decades of innovation," *IEEE J. Microwaves*, vol. 1, no. 1, pp. 191–201, Jan. 2021.

[2] I. H. Woodhouse, *Introduction to Microwave Remote Sensing*. Boca Raton, FL, USA: CRC Press, 2017.

[3] S. Bertl, *Abbildung mit Millimeterwellen für die Personenkontrolle*. Berlin, Germany: Logos Verlag Berlin GmbH, 2009.

[4] S. S. Ahmed, *Electronic Microwave Imaging With Planar Multistatic Arrays*. Berlin, Germany: Logos Verlag Berlin GmbH, 2014.

[5] X. Li, X. Wang, Q. Yang, and S. Fu, "Signal processing for TDM MIMO FMCW millimeter-wave radar sensors," *IEEE Access*, vol. 9, pp. 167959–167971, 2021.

[6] M. Brinkmann, F. Gumbmann, G. F. Hamberger, and B. Simper, "Material characterization using high-resolution multiple-input multiple-output imaging radarsyun2015ray," in *Proc. 19th Eur. Radar Conf.*, 2022, pp. 229–232.

[7] H. Ling, R.-C. Chou, and S.-W. Lee, "Shooting and bouncing rays: Calculating the RCS of an arbitrarily shaped cavity," *IEEE Trans. Antennas Propag.*, vol. 37, no. 2, pp. 194–205, Feb. 1989.

[8] Z. Yun and M. F. Iskander, "Ray tracing for radio propagation modeling: Principles and applications," *IEEE Access*, vol. 3, pp. 1089–1100, 2015.

[9] G. Ortiz-Jimenez, F. García-Rial, L. A. Úbeda-Medina, R. Pages, N. García, and J. Grajal, "Simulation framework for a 3-D high-resolution imaging radar at 300 GHz with a scattering model based on rendering techniques," *IEEE Trans. THz Sci. Technol.*, vol. 7, no. 4, pp. 404–414, Jul. 2017.

[10] J. Thieling, S. Frese, and J. Roßmann, "Scalable and physical radar sensor simulation for interacting digital twins," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3184–3192, Feb. 2021.

[11] Y. Zhou, L. Liu, H. Zhao, M. López-Benítez, L. Yu, and Y. Yue, "Towards deep radar perception for autonomous driving: Datasets, methods, and challenges," *Sensors*, vol. 22, no. 11, 2022, Art. no. 4208.

[12] C. Schüßler, M. Hoffmann, J. Bräunig, I. Ullmann, R. Ebelt, and M. Vossiek, "A realistic radar ray tracing simulator for large MIMO-arrays in automotive environments," *IEEE J. Microwaves*, vol. 1, no. 4, pp. 962–974, Oct. 2021.

[13] C. Schüßler et al., "Achieving efficient and realistic full-radar simulations and automatic data annotation by exploiting ray meta data from a radar ray tracing simulator," in *Proc. IEEE Radar Conf.*, 2023, pp. 1–6.

[14] M. Pharr, W. Jakob, and G. Humphreys, *Physically Based Rendering: From Theory to Implementation*. Cambridge, MA, USA: MIT Press, 2023.

[15] S. Laine, J. Hellsten, T. Karras, Y. Seol, J. Lehtinen, and T. Aila, "Modular primitives for high-performance differentiable rendering," *ACM Trans. Graph.*, vol. 39, no. 6, pp. 1–14, 2020.

[16] M. Nimier-David, D. Vicini, T. Zeltner, and W. Jakob, "Mitsuba 2: A retargetable forward and inverse renderer," *ACM Trans. Graph.*, vol. 38, no. 6, pp. 1–17, 2019.

[17] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *Proc. Computer Vis.–ECCV*, 2020, pp. 405–421.

[18] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Trans. Graph.*, vol. 41, no. 4, pp. 1–15, 2022.

[19] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D Gaussian splatting for real-time radiance field rendering," *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[20] L. Fink, D. Rückert, L. Franke, J. Keinert, and M. Stamminger, "Livenvs: Neural view synthesis on live RGB-d streams," in *Proc. SIGGRAPH Asia 2023 Conf. Papers*, 2023, pp. 1–11.

[21] J. Munkberg et al., "Extracting triangular 3D models, materials, and lighting from images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8280–8290.

[22] W. Jakob, S. Speierer, N. Roussel, and D. Vicini, "Dr Jit: A just-in-time compiler for differentiable rendering," *ACM Trans. Graph.*, vol. 41, no. 4, pp. 1–19, 2022.

[23] S. P. Bangaru et al., "Slang. d: Fast, modular and differentiable shader programming," *ACM Trans. Graph.*, vol. 42, no. 6, pp. 1–28, 2023.

[24] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen, "Differentiable Monte Carlo ray tracing through edge sampling," *ACM Trans. Graph.*, vol. 37, no. 6, pp. 1–11, 2018.

[25] G. Loubet, N. Holzschuch, and W. Jakob, "Reparameterizing discontinuous integrands for differentiable rendering," *ACM Trans. Graph.*, vol. 38, no. 6, pp. 1–14, 2019.

[26] S. P. Bangaru, T.-M. Li, and F. Durand, "Unbiased warped-area sampling for differentiable rendering," *ACM Trans. Graph.*, vol. 39, no. 6, pp. 1–18, 2020.

[27] S. Fu and F. Xu, "Differentiable SAR renderer and image-based target reconstruction," *IEEE Trans. Image Process.*, vol. 31, pp. 6679–6693, 2022.

[28] M. C. Wilmanski and J. I. Tamir, "Differentiable rendering for synthetic aperture radar imagery," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 60, no. 1, pp. 82–93, Feb. 2024.

[29] J. Hoydis et al., "Sionna RT: Differentiable ray tracing for radio propagation modeling," in *Proc. 2023 IEEE Globecom Workshops (GC Wkshps)*, Kuala Lumpur, Malaysia, 2023, pp. 317–321, doi: 10.1109/GCWkshps58843.2023.10465179.

[30] T. Huang et al., "DART: Implicit doppler tomography for radar novel view synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 24118–24129.

[31] D. Borts et al., "Radar fields: Frequency-space neural scene representations for FMCW radar," in *Proc. ACM SIGGRAPH 2024 Conf. Papers*, 2024, pp. 1–10.

[32] M. I. Skolnik, *Radar Handbook*. New York, NY, USA: McGraw-Hill Inc., 1970.

[33] G. W. Stimson, *Introduction to Airborne Radar*. Bellingham, WA, USA: SPIE, 1998.

[34] F.E. Nicodemus, "Directional reflectance and emissivity of an opaque surface," *Appl. Opt.*, vol. 4, no. 7, pp.767–775, 1965.

[35] M. H. Kalos and P. A. Whitlock, *Monte Carlo Methodsvolume I: Basics*. Hoboken, NJ, USA: Wiley-Interscience, 1986.

[36] D. E. Rumelhart, R. Durbin, R. Golden, and Y. Chauvin, "Backpropagation: The basic theory," in *Backpropagation*. Hove, U.K.: Psychology Press, 2013, pp. 1–34.

[37] A. Paszke et al., "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, vol. 32, pp. 7994–8006.

[38] M. Abadi et al., "{TensorFlow}: A system for {Large-Scale} machine learning," in *Proc. 12th USENIX Symp. Operating Syst. Des. Implementation*, 2016, pp. 265–283.

[39] V. Wirth, J. Bräunig, M. Vossiek, T. Weyrich, and M. Stamminger, "MAROON: A. framework for the joint characterization of near-field high-resolution radar and optical depth imaging techniques," 2024, *arXiv:2411.00527*.

[40] V. Wirth et al., "Automatic spatial calibration of near-field MIMO radar with respect to optical depth sensors," in *Proc. 2024 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Abu Dhabi, United Arab Emirates, 2024, pp. 8322–8329, doi: 10.1109/IROS58592.2024.10801705.

[41] G. Kouros, M. Wu, S. Nagesh, X. Zhang, and T. Tuytelaars, "Unveiling the ambiguity in neural inverse rendering: A parameter compensation analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 2832–2841.

[42] B. T. Phong, "Illumination for computer generated pictures," in *Seminal Graphics: Pioneering Efforts that Shaped the Field*. New York, NY, USA: ACM, 1998, pp. 95–101.

[43] B. Walter, S. R. Marschner, H. Li, and K. E. Torrance, "Microfacet models for refraction through rough surfaces," in *Proc. 18th Eurographics Conf. Rendering Techn.*, 2007, pp. 195–206.

[44] B. Smith, "Geometrical shadowing of a random rough surface," *IEEE Trans. Antennas Propag.*, vol. TAP-15, no. 5, pp. 668–671, Sep. 1967.

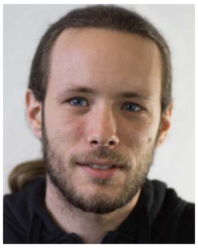
[45] D. Luebke, "CUDA: Scalable parallel programming for high-performance scientific computing," in *Proc. 5th IEEE Int. Symp. Biomed. Imaging, From Nano Macro*, 2008, pp. 836–838.

[46] S. G. Parker et al., "Optix: A general purpose ray tracing engine," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1–13, 2010.

[47] Rohde & Schwarz, "R&S QAR50 quality automotive radome tester," 2019. Accessed: Oct. 29, 2024. [Online]. Available: https://www.rohde-schwarz.com/products/test-and-measurement/radome-tester/rs-qar50-quality-automotive-radome-tester_63493-1138625.html?change_c=true

[48] J. Bräunig et al., "An ultra-efficient approach for high-resolution MIMO radar imaging of human hand poses," *IEEE Trans. Radar Syst.*, vol. 1, pp. 468–480, 2023.

[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2014.



NIKOLAI HOFMANN received the M.Sc. degree in computer science from Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany, in 2020. He is currently working toward the Ph.D. degree with the Chair of Visual Computing, FAU. He was Intern with the Real-Time Graphics Research Team, NVIDIA. His research interests include the intersection of computer graphics, machine learning, and radar simulation, with topics including ray tracing, volumetric rendering, global illumination, data-driven denoising, inverse rendering, and imaging radar.

VANESSA WIRTH was born in Schwabach, Germany, in 1996. She received the M.Sc. degree in computer science from Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany, in collaboration with the Technical University of Munich, Munich, Germany. She is currently working toward the Ph.D. degree with the Chair of Visual Computing, FAU. Her research interests include the intersection of computer graphics, computer vision, and machine learning. She is particularly interested in surface reconstruction and motion tracking of nonrigid objects using radar-based and optical sensor systems.

JOHANNA BRÄUNIG (Student Member, IEEE) was born in Bamberg, Germany, in 1993. She received the M.Sc. degree in electrical engineering from Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany, in 2020, and the Ph.D. degree from the Institute of Microwaves and Photonics, FAU, in 2025. Her research focuses on radar imaging and signal processing, particularly in radar-based motion tracking and the application of radar technology in medical engineering and machine learning. She was the recipient of the Young Engineer Prize at the European Radar Conference in 2023.

INGRID ULLMANN (Member, IEEE) received the M.Sc. degree in electrical engineering and the Ph.D. degree from Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany, in 2016 and 2021, respectively. She is currently a Postdoc and the Head of the research group "Wave-Based Sensing Techniques" with the Institute of Microwaves and Photonics, FAU. In 2022, she spent one month as a Visiting Researcher with the Microwave Sensing, Signals and Systems Group, Delft University of Technology, Delft, The Netherlands. Her research interests include radar sensing, signal processing and imaging for nondestructive testing, security screening, and medical applications. She was a reviewer for the European Radar Conference and various journals in the field of microwaves. Since 2022, she has been an Associate Editor for IEEE TRANSACTIONS ON RADAR SYSTEMS. Dr. Ullmann was the recipient of EuRAD Conference Prize in 2019 and IEEE SENSORS Best Paper Award in 2023.

MARTIN VOSSIEK (Fellow, IEEE) received the Ph.D. degree from Ruhr-Universität Bochum, Bochum, Germany, in 1996. In 1996, he joined Siemens Corporate Technology, Munich, Germany, where he led the Microwave Systems Group from 2000 to 2003. Since 2003, he has been a Full Professor with the Clausthal University of Technology, Clausthal-Zellerfeld, Germany. Since 2011, he has been the Chair of the Institute of Microwaves and Photonics (LHFT), Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany. He has authored or coauthored more than 450 publications, and his research has resulted in more than 100 granted patents. His research interests include radar, microwave systems, wave-based imaging, transponders, RF identification, communication, and wireless sensor and localization systems. He is the Spokesperson for the Collaborative Research Centre (CRC 1483) EmpkinS, where more than 80 researchers are developing innovative wireless and wave-based sensor technologies for applications in medicine and psychology. He is also a member of the German National Academy of Science and Engineering and the German Research Foundation (DFG) Review Board 4.42-02, Communication Technology and Networks, Microwave Technology and Photonic Systems, Signal Processing and Machine Learning for Information Technology. Since 2024, he has been Spokesperson for the DFG review board 4.42 for electrical engineering and information technology. Dr. Vossiek is actively involved with the IEEE Microwave Theory and Technology (MTT) Technical Committees, including MTT-24 (Microwave/mm-wave Radar, Sensing, and Array Systems), MTT-27 (Connected and Autonomous Systems, where he was the founding chair), and MTT-29 (Microwave Aerospace Systems). Additionally, he is on the Advisory Board of the IEEE CRFID Technical Committee on Motion Capture & Localization. He was the recipient of numerous best paper awards and other distinctions. In 2019, the IEEE MTT Society (MTT-S) honored him with the Microwave Application Award for his pioneering research in wireless local positioning systems. He was on organizing committees and technical program committees for many international conferences and has contributed as a reviewer for various technical journals. From 2013 to 2019, he was an Associate Editor for IEEE TRANSACTIONS ON MICROWAVE THEORY AND TECHNIQUES, and since October 2022, he has been an Associate Editor-in-Chief for IEEE TRANSACTIONS ON RADAR SYSTEMS.

TIM WEYRICH received the Ph.D. degree from ETH Zurich, Zurich, Switzerland, in 2006. He was a Postdoctoral Teaching Fellow with Princeton University, Princeton, NJ, USA, working in the Princeton Computer Graphics Group. He was with University College London (UCL), London, U.K., in 2008. In addition, he retains a part-time assignment as Professor of visual computing in the Virtual Environments and Computer Graphics Group, Department of Computer Science, UCL. He is currently the Chair of Digital Reality with the Department of Computer Science, Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany, and is an Associate Member of FAU's Department for Digital Humanities and Social Studies. His research interests include physically-based rendering, computational photography, appearance modeling and fabrication, point-based graphics, cultural heritage analysis and digital humanities. He was an Associate Editor for IEEE TRANSACTIONS OF VISUALIZATION AND COMPUTER GRAPHICS, *Computer and Graphics*, Elsevier, and *Computer Graphics Forum*, Wiley/Eurographics. He is a Member of the Executive Committee of Eurographics, Member of the steering board of the Eurographics Workshop on Graphics and Cultural Heritage, and Member of the steering committee EPSRC Doctoral Training Centre of Science and Engineering in Arts, Heritage and Archaeology.

MARC STAMMINGER is currently the Chair of Visual Computing with Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany. His research interests include various areas of visual computing, such as real-time computer graphics, neural and inverse rendering, physical light and microwave simulation, and virtual and augmented reality. He has authored or coauthored more than 200 papers in leading journals and conference proceedings. He is an Associate Editor for the *Journal of Computer Graphics Techniques*. He was a Program Chair for Eurographics, High Performance Graphics, and the Eurographics Symposium of Rendering, and participated in the program committees of all major computer graphics conferences. He has been elected to the Eurographics Executive Committee and is the Eurographics Professional Board chair.