LEARNING FREQUENCY-TRANSFERABLE RADIO DIGITAL TWINS WITH GAUSSIAN RAY TRACING

Anonymous authors
Paper under double-blind review

ABSTRACT

Emerging multi-band systems (e.g. WiFi 7) raise the importance of the multi-band radio-frequency digital twin (RFDT), a virtual replica of how radio signals propagate within a physical site, which serves numerous applications. However, existing approaches only focus on a single frequency band, and directly predicting the signal of unseen bands with these DTs does not work because RF propagation varies with frequency. Building an RFDT with multi-band data, on the other hand, incurs huge measurement overhead as the number of bands increases. We therefore propose FresGRT, the first one-to-many-band RFDT built on single-band RF measurements with 3D Gaussian ray tracing. FresGRT first learns the frequency-invariant representation based on the Fresnel equations to achieve cross-band prediction with only single-band data. FresGRT then learns from the frequency-selective fading to decouple the physically coupled representation parameters, and optimizes them separately with a staged training pipeline. We extensively evaluate FresGRT on real testbed and ray-tracing datasets. Results on the testbed show that FresGRT achieves unseen band prediction with a mean error of 4.4 dB, a 45% improvement over the best baseline, while reducing the survey time by 55 %.

1 INTRODUCTION

Radio-frequency digital twin (RFDT) is a virtual replica of how radio signals propagate within a physical site, which predicts the signal for any placement of transmitters and receivers. Such an RFDT serves a wide range of tasks, including coverage and interference analysis (Hoydis et al., 2023; Zhang et al., 2023), access-point placement (Taufique et al., 2017), and beam management (Zhao et al., 2023; Zhang et al., 2025). RFDT provides the network manager an interface to understand the signal propagation behavior without any actual laborious RF measurements (Matson & Sundaresan, 2024; Yang et al., 2024). Moreover, the emerging integrated sensing and communication (ISAC) in 6G repurposes communication waveforms for environment understanding to improve the throughput, where RFDT plays an important role. WiFi is standardizing sensing and positioning (IEEE 802.11bf and 802.11az (IEEE, 2022; 2025)), leading to promising downstream applications, such as indoor localization (Zafari et al., 2019; Wang et al., 2020) and WiFi human sensing (Hsu & Hsieh, 2024; 2025). In 6G, ISAC moves from an add-on to a defining feature of the air interface (Liu et al., 2022), where RFDT becomes the site model that ISAC sensing and holographic digital-twin services are based on. Consequently, 6G architectures place a continuously maintained RFDT at their core (Zhang et al., 2025; Khan et al., 2022). The above use cases show that the accuracy of the twin bounds the accuracy of the decision made on top of it.

However, building an RFDT is challenging. RF signal interacts with objects in complicated ways, e.g. transmission and reflection, and the constructive and destructive interference caused by multi-path (different paths to the receiver) makes RF signals vary significantly with environment. While existing works have proposed model-based approaches to build an RFDT (Hoydis et al., 2023; Suga et al., 2023), the research community started to introduce learning-based approaches borrowed from computer vision to reconstruct more faithful twins. NeRF^2 (Zhao et al., 2023; Lu et al., 2024) brings neural radiance fields to radio by learning a volumetric field that re-radiates measured signals toward a receiver, and neural ray tracers such as WiNeRT (Orekondy et al., 2023) and RFScape (Chen et al., 2025) learn ray–surface interaction surrogates inside a tracer. WRF-GS (Wen et al., 2025), GSRF (Yang et al., 2025), RF-3DGS (Zhang et al., 2026), and GSpaRC (Nukapotula et al., 2025) replace the implicit field with explicit 3D Gaussians for speed and editability by attaching a learned radiance or attenuation to each Gaussian. RFIR (Wang et al., 2026) pushes toward inverse rendering that

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

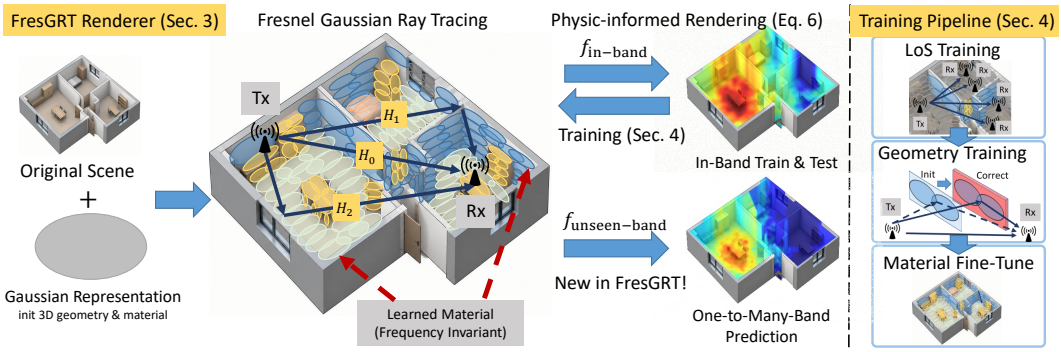

Figure 1: Overview of FresGRT. (a) FresGRT learns the frequency-invariant representation of the scene for one-to-many-band prediction. (b) FresGRT's staged-training pipeline decouples the scene parameters to separately optimize the representation step by step.

separates sources from scene. These models predict received power and spectra with few dB errors and reproduce fine spatial structure that classical ray tracing on a coarse mesh misses.

Meanwhile, recent wireless communication systems have shifted from single-band communication to multi-band communication. For example, WiFi 7 access point operates on the 2.4 GHz, 5 GHz, and 6 GHz bands at once and steers traffic between them through multi-link operation (IEEE, 2024). Given this trend, a DT must predict every used band for the sake of network planning or traffic steering. However, existing learning-based approaches only enable in-band prediction, i.e. training and testing the DT in the same frequency band. Directly predicting unseen bands with these DTs does not work well, since RF propagation varies with frequency. Signals of different frequencies undergo different attenuation from the same obstacle (6.9 dB difference between 2.4 GHz and 6 GHz for a 10 cm concrete wall (ITU-R, 2021)), and the multipath interference pattern changes with frequency. Although it is possible to build an RFDT with data measured from every targeted frequency band, the survey cost grows as $O(N)$ with the number of bands N, which takes hours in practice (5.3 h for three bands in a 108 m^2 site in our experiments, Sec. 5.1). Therefore, unseen band prediction becomes more and more important but still remains unsolved.

We propose *FresGRT*, Fresnel Gaussian Ray Tracing, to the best of our knowledge the first one-to-many-band RFDT, which predicts the RF characteristics of unseen bands by learning the frequency-invariant representation rather than a band-specific radiance (shown in Fig. 1). Reaching one-to-many-band prediction from single-band measurements raises three challenges. 1) What is the frequency-invariant representation that the DT should learn? Unlike prior works that learn frequency-specific signal radiance representation, the learned representation at one frequency should be able to transfer to another. 2) How can the complicated RF propagation be built into the representation? RF signals propagate to the receiver through multipath, undergoing transmissions and reflections, so the DT should incorporate these into the learned representation. 3) How can the representation be recovered from the measurements of a single band? All the materials and geometry of the representation are coupled together in the signal propagation equations (detailed in Eq. (8)), making it hard to optimize the parameters separately. Towards addressing these challenges, our contributions are as follows:

- We propose a frequency-invariant representation that carries the material parameters of the Fresnel equations and the ITU slab model (ITU-R, 2021), the relative permittivity η' and the conductivity σ. Because (η', σ) follow a known frequency law, a DT fitted at one frequency can be evaluated at another without adaptation. We adapt Gaussian ray tracing (Moenne-Loccoz et al., 2024) from vision to RF by replacing the optical interaction with these physical equations. Unlike Gaussian splatting and neural radiance fields, Gaussian ray tracing traces the reflections and the transmissions that RF signals actually undergo. We derive the gradients of these equations in closed form and implement them in CUDA to optimize the DT.
- We recover the representation from the measurements of a single band with staged training. We propose to learn from the frequency-selective fading, which is determined by the path lengths, to decouple the geometry and the materials of the representation. The stages exploit this to optimize these parameters step by step.

- We set up a testbed to collect 3 multi-band datasets in different spaces, together with 24 ray-tracing datasets from 8 real floor plans for large-scale evaluation. FresGRT achieves cross-band prediction with a mean error of 4.4 dB (45% improvement) and 4.0 dB (52% improvement) on real and ray-tracing datasets over the best baseline, reducing the data collection to one band. The source code and the datasets of this work are released[1].

2 Related work

NeRF-based RF twins. NeRF^2 (Zhao et al., 2023) brought neural radiance fields (Mildenhall et al., 2020) to radio. It learns a volumetric field that re-radiates the signal of a transmitter toward a receiver and renders the spatial spectrum at the receiver. NeWRF (Lu et al., 2024) proposes a ray sampling strategy according to the direction of arrival (DoA) of the measured paths to reduce the required training data and accelerate rendering. WiNeRT (Orekondy et al., 2023) keeps a ray tracer as the forward model and learns neural surrogates of the ray–surface interactions. RFScape (Chen et al., 2025) adopts the signed distance function (SDF) and the neural representation to learn the attenuation per object from measurements. EchoNeRF (Amballa et al., 2025) learns the representation of environment opacity from the signal power with neural ray tracing. Nevertheless, all of them are trained and evaluated at one carrier frequency.

3DGS-based RF twins. Gaussian-based twins replace the implicit field with explicit 3D Gaussians (Kerbl et al., 2023) for rendering speedup. WRF-GS (Wen et al., 2025) splats the wireless radiation field to synthesize spatial spectra in milliseconds, and GSRF (Yang et al., 2025) makes the Gaussians complex-valued for rendering the phase of the field. RF-3DGS (Zhang et al., 2026) replaces the color of a Gaussian by spherical-harmonic channels that encode its spatial channel state information (CSI) (path gain, delay, and angle of departure), and splats the Gaussians onto the angular domain of the receiver to render the spatial spectrum. GSpaRC (Nukapotula et al., 2025) adds physics-informed features, e.g., distance attenuation, to each Gaussian and a hemispherical projection around the receiver to reach sub-millisecond rendering. RFIR (Wang et al., 2026) moves toward inverse rendering by giving each Gaussian surface normal, complex reflection coefficient, and roughness, and renders radar cross section and received power through an RF scattering function. In all of these models, every Gaussian learns radiance, attenuation, or reflection coefficient from data, where those properties are coupled with the training carrier frequency, making them hard to transfer to unseen frequency bands. In contrast, FresGRT's Gaussians carry the relative permittivity and conductivity of its material, whose frequency dependence is anchored by electromagnetics, so the twin is evaluated at an unseen band without adaptation.

Gaussian ray tracing. 3D Gaussian splatting (Kerbl et al., 2023) and its surfel variant 2DGS (Huang et al., 2024) established Gaussians as an optimizable scene representation. 3DGRT (Moenne-Loccoz et al., 2024) traces rays against Gaussian particles instead of rasterizing them, and IRGS (Gu et al., 2025) builds inverse rendering with inter-reflections on top of Gaussian ray tracing in the optical domain. Nonetheless, RF signals interact with objects differently from light. While a light ray stops at the first object it hits, an RF signal penetrates the object because of its long wavelength. RF signals are also sensitive to phase, which creates constructive and destructive interference between multipath components. FresGRT inherits the representation from Gaussian ray tracing and adapts the ray–object interaction to RF signals. Gaussian ray tracing determines the rays that bounce off and penetrate the objects in the environment, and FresGRT's physics-informed rendering equation, derived from the Fresnel equations, governs the interaction between rays and objects, which makes the learned representation transferable across frequencies (Sec. 3).

3 FresGRT renderer

FresGRT renders the channel frequency response between a transmitter and a receiver by tracing rays through a scene of Gaussians and applying the Fresnel equations, as shown in Fig. 1. FresGRT also imposes the slab model (ITU-R, 2021) on Gaussians for precise signal propagation modeling. The rendering equations are fully differentiable and implemented on CUDA kernels.

3.1 Scene representation

FresGRT builds on Gaussian ray tracing rather than Gaussian splatting, because transmissions and reflections can be traced with the ray tracer, while rasterizing the Gaussians only accumulates them

[1] https://anonymous.4open.science/r/FresGRT-061C

along the ray to the receiver. While traditional Gaussian ray tracer gives every Gaussian its own appearance, it incurs more freedom than an indoor radio scene needs. Most surfaces in a building are planar walls, floors, and ceiling with single radio parameter. Therefore, according to their coplanarity and adjacency, we group the Gaussians into planar *faces* with the same normal and into *material groups* with the same radio parameters by assuming every object contains single evenly-distributed material. Inspired by Gaussian surfels (Dai et al., 2024), we apply the surfel idea to a face instead of a single Gaussian, making all member Gaussians of a face share one plane, one thickness, and one material. A face w is a planar part of an object, whose member Gaussians lie on one plane with unit normal n_w, and the face carries a learnable thickness t_w. Member g has a center μ_g and a learnable occupancy $\rho_g \in (0, 1)$ (Fig. 2). We optimize geometry at the face level, the translation along the normal and the thickness, and keep only the occupancy per Gaussian for openings on the face, e.g. doors and windows. Each Gaussian member whose covariance $\Sigma_g = R_w S_w S_w^\top R_w^\top$ has its scale S_w and rotation R_w shared by the whole face,

$$S_w = \mathrm{diag}\big(t_w/2,\ s_t,\ s_t\big), \qquad R_w = [\, n_w \ \ e_{w,1} \ \ e_{w,2} \,]. \tag{1}$$

The Gaussian is a flat ellipsoid whose thin axis is the face normal and whose two in-plane axes $e_{w,1}, e_{w,2}$ span the face. A material group b carries a relative permittivity η_b' and a conductivity σ_b at the reference frequency f_0. At frequency f the complex relative permittivity is

$$\eta_b(f) = \eta_b' - \mathrm{j}\,\frac{\sigma_b\,(f/f_0)^\gamma}{2\pi f\,\varepsilon_0}, \tag{2}$$

which is the power-law form of the ITU-R P.2040 material model (ITU-R, 2021; Conrat et al., 2024). η_b' is constant over 1–10 GHz for building materials, and σ_b grows as f^γ with an exponent that differs by material, from 0.78 to 1.34. Single-band power cannot identify γ, so we fix $\gamma = 0.9$ (the median among the materials) for all groups. Note that the relative permittivity in Eq. (2) is a function of f, meaning that the learned representation can be transferred to another frequency.

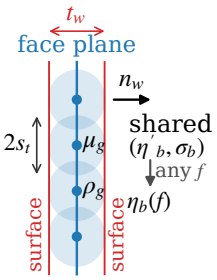

Figure 2: Scene representation.

3.2 Rendering Equation

Let p_{tx} and p_{rx} be the positions of the transmitter and the receiver. The channel state information (CSI) $H(f)$ at the receiver is the sum of the line-of-sight path and the reflected paths (see Fig. 1a),

$$H(f) = H_0(f) + \sum_r H_r(f), \tag{3}$$

where r indexes the specular reflection paths. Each path is attenuated by free-space spreading, by the transmission through every Gaussian it crosses, and by the reflection coefficient of every face it bounces on. Let $T(p \to p')$ be the transmission along the straight segment from position p to p' (defined in Eq. (8)), and let $p_{r,1}, \ldots, p_{r,n_r}$ be the reflection points of path r on the faces $w_{r,1}, \ldots, w_{r,n_r}$, with $p_{r,0} = p_{\mathrm{tx}}$ and $p_{r,n_r+1} = p_{\mathrm{rx}}$. The direct path and a reflected path are

$$H_0 = \frac{\lambda\, T(p_{\mathrm{tx}} \to p_{\mathrm{rx}})}{4\pi d_0}\, \mathrm{e}^{-\mathrm{j}k_0 d_0}, \qquad H_r = \frac{\lambda}{4\pi d_r} \prod_{i=1}^{n_r} \Gamma_{w_{r,i}}(\theta_{r,i}) \prod_{i=0}^{n_r} T(p_{r,i} \to p_{r,i+1})\, \mathrm{e}^{-\mathrm{j}k_0 d_r}, \tag{4}$$

where $k_0 = 2\pi f/c$ is the wavenumber, $\lambda = c/f$ the wavelength, $d_0 = \|p_{\mathrm{rx}} - p_{\mathrm{tx}}\|$ and $d_r = \sum_{i=0}^{n_r} \|p_{r,i+1} - p_{r,i}\|$ are the path lengths, $\theta_{r,i}$ is the incidence angle at the i-th reflection, and Γ_w is the reflection coefficient of face w (defined in Eq. (8)). The reflected paths in Eq. (3) interfere with the direct path, causing increase or decrease in received power. These interferences depend on the phase of the reflection, determined by the path length, and the inaccuracy of path length can cause a 3 dB power difference (analyzed in Sec. A.3). We therefore design two factors, $c_r(f)$ and $c_x(f)$, to reduce the impact of the path length variation,

$$c_r(f) = \exp\big(-\tfrac{1}{2}(2k_0 s_w \cos\theta_w)^2\big) \in [0, 1], \qquad c_x(f) = \exp\big(-\tfrac{1}{2}(k_0\sigma_x)^2 n\big) \in [0, 1]. \tag{5}$$

$c_r(f)$ sets how much a reflection on face w interferes with other paths (1: full interference, 0: pure power addition), where s_w is a learnable coherence parameter of face w, the standard deviation of its position error along the normal, and θ_w the reflection incident angle. $c_x(f)$ attenuates a reflection whose path crosses n objects. The phase of the transmission coefficient T in Eq. (8) depends on

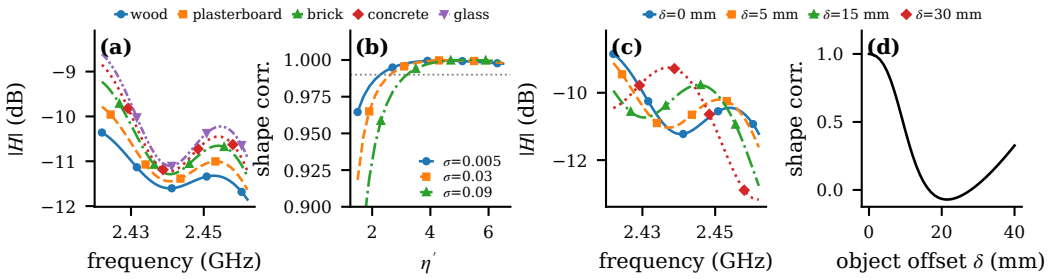

Figure 3: Example of frequency selective fading among different frequencies in a 2.4 GHz band with (a)(b) material variation (c)(d) path length variation

the thickness and the material of the object, and the path length errors lead to large phase errors (analyzed in Sec. A.4). Here σ_x is the standard deviation of the transmission path length error, estimated once from the training data (detailed in Sec. C). While it does not faithfully reconstruct the interference, $c_r(f)$ and $c_x(f)$ prevent the degradation from inaccurate geometry. Applying c_r and c_x gives the rendering equation:

$$P(f) = \kappa\Big(\big|H_0(f) + \textstyle\sum_r c_r(f)\, c_x(f)\, H_r(f)\big|^2 + \sum_r \big(1 - c_r(f)^2\big)\, c_x(f)^2\, |H_r(f)|^2\Big), \quad (6)$$

where κ denotes the learned equivalent gain of transmit power, antenna, and radio-chain. We leave the derivation of Eq. (6) in Sec. A.3 and Sec. A.4. FresGRT's rendering equation is a function of frequency f, which helps render the signal at any frequency, unlike the prior work in Sec. 2 that only learns a representation fitted to the training band.

3.3 Transmission and Reflection

The material a ray segment from p to p' crosses in Gaussian g is the line integral of its occupancy-weighted density,

$$\ell_g = \rho_g \int_0^L \exp\Big(-\tfrac{1}{2}\,(p + s\,u - \mu_g)^\top \Sigma_g^{-1} (p + s\,u - \mu_g)\Big)\, ds, \qquad u = \frac{p' - p}{L},\ L = \|p' - p\|, \quad (7)$$

which has a closed form in the error function (see Sec. A.1). ℓ_g is the length of material that the segment crosses in member Gaussian g. Summed over the Gaussian members of an object w, $\ell_w = \sum_{g \in w} \ell_g$ is the length of material the segment crosses in that object, which equals $t_w / \cos\theta$ for a full crossing at incidence θ to the normal n_w. We model the transmission and the reflection of an object with the ITU-R slab model (ITU-R, 2021), formulated as

$$T_w = \frac{(1 - r^2)\, \mathrm{e}^{-\mathrm{j}\phi_w}}{1 - r^2 \mathrm{e}^{-2\mathrm{j}\phi_w}}, \quad \phi_w = k_0 q_m \ell_w \cos\theta, \qquad \Gamma_w = \frac{r\,(1 - \mathrm{e}^{-2\mathrm{j}\psi_w})}{1 - r^2 \mathrm{e}^{-2\mathrm{j}\psi_w}}, \quad \psi_w = k_0 q_m t_w, \quad (8)$$

where r is the Fresnel reflection coefficient (Eq. (22)) and $q_m = \sqrt{\eta - \sin^2\theta}$. The transmission of a segment is $T(p \to p') = \prod_w T_w$ over the objects it crosses, and Γ_w reduces to r when the thickness of the face is unknown. Reflection points are found by mirroring the transmitter in the face planes, and only faces wider than the first Fresnel zone are allowed to reflect (see Sec. B), since a smaller surface scatters the signal rather than reflecting it specularly.

The predicted power of Eq. (6) is therefore a closed-form composition of erf, square roots, exponentials, and rational functions of the face planes and thicknesses, the occupancies, the materials, and the frequency, so its gradients exist in closed form, and we implement the forward pass and these gradients as CUDA kernels (derivations in Sec. A).

4 FresGRT Training Pipeline

Since the material and geometry of the representation are coupled (detailed in Sec. 4.2), making them unidentifiable from the measurements, FresGRT proposes to learn from the frequency-selective fading to decouple the parameters. FresGRT therefore trains in three stages: line of sight (LoS), geometry, and fine-tuning, to learn the representation step by step.

4.1 LINE-OF-SIGHT TRAINING

The initialized scene representation can be noisy, and every material group starts at a default value. We therefore begin by tracing the LoS path only, with the geometry of the Gaussians locked, and fit the occupancy and the materials. The receiver receives the multipath signal, but the direct path carries most of its energy, which is enough to move the wrong initial parameters close to their correct values. To limit the degrees of freedom and avoid overfitting, occupancy and material are updated in two separate steps: occupancy is trained first to remove the spurious Gaussians, followed by the optimization of materials. Gaussians whose occupancy falls below 0.02 after the occupancy step are pruned. The error between the rendered and the measured power in dB is long-tailed, because a few links are blocked or sit in deep fades. We therefore minimize the Huber loss (Huber, 1964) for both steps on band-mean power,

$$\mathcal{L}_{\text{level}} = \frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \text{huber}\left(\hat{\bar{p}}_i - \bar{p}_i\right), \qquad \text{huber}(x) = \min\left(\tfrac{1}{2}x^2,\ \delta|x| - \tfrac{1}{2}\delta^2\right), \tag{9}$$

over the set $\mathcal{B}$ of training Tx–Rx pairs i. Here $\bar{p}_i = 10\log_{10}\text{mean}\left(10^{\mathbf{p}_i/10}\right)$ is the measured band-mean power, with $\mathbf{p}_i = [P_i(f_1), \ldots, P_i(f_K)]^\top$ the measured power in dB at the K frequencies of the band, and $\hat{\bar{p}}_i$ and $\hat{\mathbf{p}}_i$ are their rendered counterparts. Band-mean power is the average attenuation of a link over the band, i.e. its power level, which is why we call $\mathcal{L}_{\text{level}}$ the "level loss", and it is insensitive to the frequency-selective fading introduced in Sec. 4.2. Although the measurements at the individual frequencies carry more information, at this stage that information cannot be explained by the materials and the occupancy, so we choose the band-mean power as the training target.

4.2 GEOMETRY TRAINING

A geometric error changes the length of a reflected path and therefore its phase, which changes the interference at the receiver (Sec. 3.2). The geometry stage turns the reflections on and updates only the geometry of the Gaussians. Since only the objects that the direct paths cross were updated in the LoS stage, the objects that only the reflections hit still maintain their initialized geometry and materials. However, reflection coefficient in Eq. (8) couples both parameters, making them hard to learn accurately from reflections. We therefore use frequency selectivity (Tse & Viswanath, 2005) to decouple geometry and material. Frequency selectivity depends only on the phases of the multipath components, so the CSI relation between different frequencies of a band does not largely change with the material but changes strongly with path length variation. We show this in Fig. 3 with a Tx–Rx pair in the room but varying the materials and face offsets, and we can see that only the face offsets incur huge correlation variation among different frequencies. This inspires us to train the geometry on the similarity between the rendered and the measured CSI across frequencies to adjust the geometry of Gaussians, i.e. μ_g and t_w:

$$\mathcal{L}_{\text{shape}} = \frac{1}{|\mathcal{B}_{\text{u}}|} \sum_{i \in \mathcal{B}_{\text{u}}} w_i \left(1 - \text{corr}(\hat{\mathbf{p}}_i, \mathbf{p}_i)\right), \qquad w_i \propto \text{std}(\mathbf{p}_i). \tag{10}$$

Here $\mathcal{B}_{\text{u}}$ is the set of training pairs with an unobstructed direct path, while the other pairs are excluded since the phase of the transmission through an object depends on its thickness and permittivity, neither of which is accurate at this point. The function corr is the Pearson correlation coefficient between the rendered and the measured power vectors, and std is the standard deviation of a vector over its K entries. The weight w_i, proportional to the standard deviation of the measured power across the band, gives more importance to the positions where a larger frequency selectivity is observed. We include single-bounce reflections in early training for coarse training and add double-bounce reflections afterwards for fine tuning.

4.3 MATERIAL FINE-TUNING

With the geometry fixed, this stage renders the line of sight and all reflections and learns the materials (η'_b, σ_b), the gain κ, and the coherence parameters s_w of Eq. (5) together by minimizing

$$\mathcal{L} = \mathcal{L}_{\text{level}} + \mathcal{L}_{\text{depth}} + \mathcal{L}_{\text{anchor}}, \tag{11}$$

$$\mathcal{L}_{\text{depth}} = \frac{1}{|\mathcal{B}_{\text{u}}|} \sum_{i \in \mathcal{B}_{\text{u}}} \text{huber}\left(\text{std}(\hat{\mathbf{p}}_i) - \text{std}(\mathbf{p}_i)\right), \qquad \mathcal{L}_{\text{anchor}} = \frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \text{huber}\left(\hat{\bar{p}}_i^{\text{LoS}} - \hat{\bar{p}}_{i,0}^{\text{LoS}}\right), \tag{12}$$

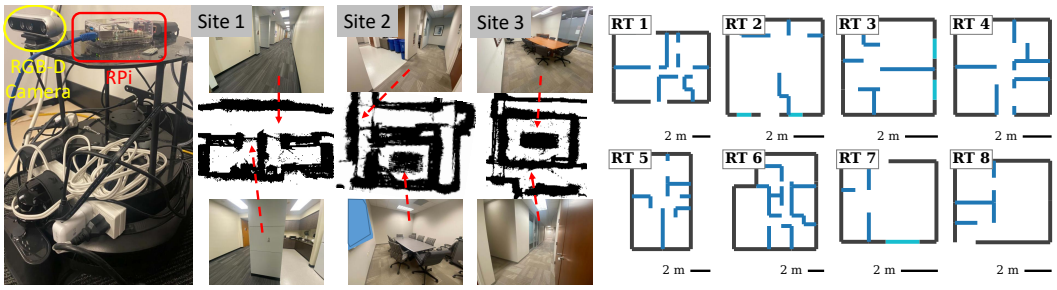

Figure 4: Evaluation environments: (a) Robotic testbed (b) measured sites (c) ray-tracing floor plans

where $\hat{\bar{p}}_i^{\mathrm{LoS}}$ is the band-mean power rendered with the direct path only and $\hat{\bar{p}}_{i,0}^{\mathrm{LoS}}$ is its value cached after the LoS stage. We introduce three losses in this stage. Since the reflections are now rendered to adjust the materials, $\mathcal{L}_{\mathrm{level}}$ of Eq. (9) is kept to learn the attenuation caused by the materials of the reflecting surfaces from the band-mean power. We further introduce the depth loss $\mathcal{L}_{\mathrm{depth}}$ to learn from the frequency-selective fading how the materials shape it: whereas the correlation in $\mathcal{L}_{\mathrm{shape}}$ of Eq. (10) discards the amplitude of the fading, which the materials determine, the standard deviation in $\mathcal{L}_{\mathrm{depth}}$ keeps the amplitude, giving one more information to train on the weak reflected signals as shown in Fig. 3a. Both losses are still evaluated on $\mathcal{B}_{\mathrm{u}}$ pairs only to avoid the wrong interference that transmission with inaccurate material can cause. To keep the first two losses from over-adjusting the materials, we introduce the anchor loss $\mathcal{L}_{\mathrm{anchor}}$ on all pairs $\mathcal{B}$, which caches the direct-path prediction of every pair at its value $\hat{\bar{p}}_{i,0}^{\mathrm{LoS}}$ from that stage.

5 Experiments

5.1 Experiment setup

Testbed. We build the testbed on a TurtleBot 4 (Clearpath Robotics and Open Robotics, 2022) that carries an Intel RealSense D455 RGB-D camera (Keselman et al., 2017) and a Raspberry Pi (Fig. 4a). The robot localizes with visual SLAM (Labbé & Michaud, 2019) and takes a depth panorama from 8 directions at every position to build the point cloud that initializes the Gaussians (Sec. B). Three Raspberry Pis placed in the environment communicate with the one on the robot, and all of them run the Nexmon CSI firmware (Gringoli et al., 2019) to collect CSI.

Ray-tracing simulation. For large-scale experiments, we build 3D models of indoor scenes from the ZInD dataset (Cruz et al., 2021) and simulate the channels with Sionna ray tracing (Aoudia et al., 2025). The simulation traces up to five reflections per path with transmission through objects enabled, and assigns ITU-R P.2040 materials, e.g. concrete and plasterboard (ITU-R, 2021), to the walls for a realistic simulation.

Datasets. We collect three datasets with the testbed in three indoor sites shown in Fig. 4b, which contain walls and furniture. Among these sites, the robot stops at 141 to 174 positions per site, and measures the links to the three anchors at 2.4, 5, and 5.8 GHz with a 40 MHz bandwidth. Since the radios need about 90 s to measure bands at every position, one site takes 4.3 to 5.3 hours to survey. We also build 24 ray-tracing datasets on 8 floor plans of 22 to 118 m^2 composed only of walls (Fig. 4c). Each floor plan is assigned with 3 materials, and we permute the walls with different materials. We generate 784 to 27,378 measurements at 1.0, 2.4, 5.0, and 5.8 GHz. All the data positions in each scene are uniformly sampled over the floor, and we detail the datasets in Sec. D.

Baselines. We compare FresGRT with existing RF neural rendering and 3D Gaussian splatting approaches. Neural radiance approaches include RFScape (Chen et al., 2025), NeRF^2 (Zhao et al., 2023), and EchoNeRF (Amballa et al., 2025), while 3D Gaussian splatting methods include WRF-GS+ (Wen et al., 2026) and GSpaRC (Nukapotula et al., 2025).

5.2 Evaluation

We evaluate FresGRT with randomly sampled 80% of the positions as training data (rest for in-band testing, all positions for cross-band testing). FresGRT is respectively trained on one of the bands and tested on the rest bands to evaluate the cross-band prediction by inputting the tested band frequency f' and the positions of the transmitter and receiver. The remaining training details are given in Sec. C. We report the mean absolute error (MAE) of the band-mean power defined in Eq. (9) as the

evaluation metric, which is the commonly used index, i.e. received signal strength indicator (RSSI), for wireless network management.

Overall performance. Tabs. 1 and 2 report the in-band and the cross-band errors of FresGRT and of every baseline on the real sites and on the ray-tracing datasets, trained at 2.4 GHz and at 5 GHz (the results trained on 1 GHz and 5.8 GHz are in Sec. E).

For in-band scenario, where training and testing are in the same band, all models perform similarly. On the real sites, the largest error is 3.89 dB (EchoNeRF) when trained at 2.4 GHz and 4.25 dB (EchoNeRF) when trained at 5 GHz, while in simulation the maximum errors are 4.55 dB (GSpaRC) and 4.62 dB (WRF-GS+). However, in cross-band scenario, where the models trained on one band directly do zero-shot prediction on other bands, FresGRT clearly outperforms every baseline. On the real sites, FresGRT outperforms the best baseline by 47% and 50% at 5 GHz and 5.8 GHz when trained at 2.4 GHz, and by 44% and 38% at 2.4 GHz and 5.8 GHz when trained at 5 GHz. The same advantage still appears in the ray-tracing datasets, where the scenes are larger and the measurements denser. FresGRT outperforms the best baseline by 62%, 47% and 46% at 1 GHz, 5 GHz and 5.8 GHz when trained at 2.4 GHz, and by 65%, 53% and 39% at 1 GHz, 2.4 GHz and 5.8 GHz when trained at 5 GHz. Averaged over all cross-band testing, FresGRT reaches 4.4 dB on the real sites and 4.0 dB on the ray-tracing datasets, 45% and 52% below the best baseline. One key finding here is that the prediction of lower bands is more accurate. The reason is that the wavelength at lower frequency is longer, and the phase is therefore less sensitive to path length variation, making the multipath interference easier to predict as discussed in Sec. 3.2.

Table 1: MAE of the band-mean power (dB) averaged on the real sites.

MAE (dB) ↓	Trained at 2.4 GHz			Trained at 5 GHz		
Band (GHz)	2.4	5	5.8	5	2.4	5.8
RFScape	3.66	8.08	9.91	**3.65**	10.26	8.69
NeRF^2	**3.45**	9.06	11.11	4.00	9.19	7.16
EchoNeRF	3.89	10.67	12.73	4.25	7.06	6.97
GSpaRC	3.74	9.44	11.40	3.92	8.81	7.22
WRF-GS+	3.50	9.17	11.33	3.74	9.37	7.86
FresGRT	3.60	**4.31**	**4.96**	3.76	**3.92**	**4.32**

FresGRT learns the Gaussian material field that recovers the material properties of the environment from measurements at a single band, and since these properties transfer across frequency, the learned scene is able to render at any other band without new data. Each baseline instead learns a representation tied to the training frequency, which does not extend to a band absent from the training data.

Table 2: MAE of the band-mean power (dB) averaged on the ray-tracing datasets

MAE (dB) ↓	Trained at 2.4 GHz				Trained at 5.0 GHz			
Band (GHz)	2.4	1.0	5.0	5.8	5.0	1.0	2.4	5.8
RFScape	4.51	7.14	8.62	9.28	4.17	9.16	8.43	7.58
NeRF^2	3.24	10.12	9.16	11.13	**3.60**	19.05	9.11	7.55
EchoNeRF	4.02	7.52	13.13	15.07	4.39	14.39	7.73	9.31
GSpaRC	4.55	10.53	10.53	12.20	4.48	19.08	10.04	7.97
WRF-GS+	4.23	10.63	11.04	12.68	4.62	19.00	10.22	8.20
FresGRT	**3.14**	**2.74**	**4.55**	**5.03**	4.24	**3.17**	**3.63**	**4.62**

Data collection reduction. An advantage of the cross-band prediction of FresGRT is that the model trained on only one band can predict any other band, which saves the time of collecting the measurements of the other bands to train the digital twin. Tabs. 1 and 2 show that the cross-band prediction is only slightly worse than the in-band prediction, by 0.7 dB on the real sites. In our testbed, it takes 90 s to measure the 3 bands at every position, whereas FresGRT only needs the measurements of one band (30 s), saving $2/3$ of the time. FresGRT therefore saves 2.4 to 2.9 hours of RF measurement overhead per site, which is a 55 % reduction of the total survey time. Details of the measurement time can be found in Sec. D.

Learned scene representation. Fig. 5 shows the scene representation, i.e. η' and σ fields, that FresGRT learns in part of real sites and ray-tracing datasets on 2.4GHz (we show the ray-tracing dataset with plasterboard interior walls in Fig. 5, and more results are shown in Fig. 8). Although the ground-truth materials of the real sites are not available, we report in Tab. 3 the recognition accuracy of the material classified into pre-defined categories on the ray-tracing datasets, whose materials are fully controllable. Results show that FresGRT recognizes 72% to 73% of the wall materials correctly. We further split the walls into interior walls (observed through both transmitted and reflected signals) and exterior walls (observed only through reflections) for further discussion. The interior walls reach 92% to 98% accuracy, whereas the exterior walls reach only 44% to 62%. Since the materials of exterior walls are learned from the reflected signals alone, the weak reflection power constrains the materials loosely. Nonetheless, as the reflection power only accounts for a

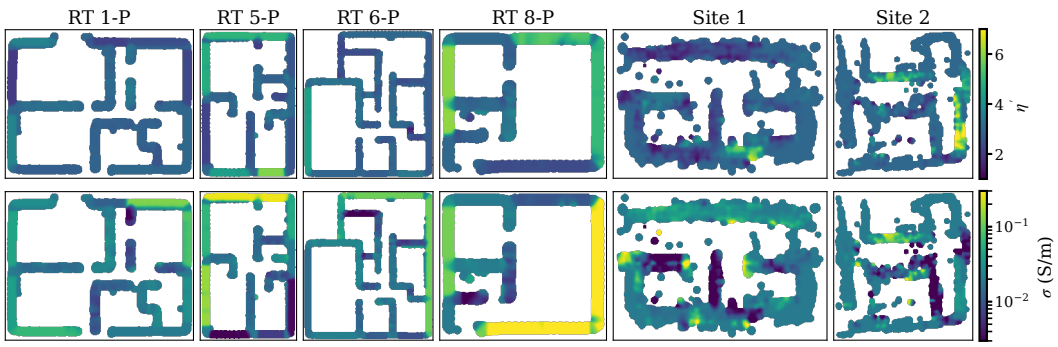

Figure 5: FresGRT's learned scene representation of real sites and ray-tracing datasets. More results are in Sec. E. Top row: the relative permittivity η'. Bottom row: the conductivity σ.

Table 3: Material recognition accuracy (%) of the Gaussians in simulation.

Scene	Interior	Exterior	All
RT 1	97	48	73
RT 5	96	54	72
RT 6	98	44	72
RT 8	92	62	73

Table 4: MAE of the band-mean power (dB) of the ablation of the training stages, trained at 2.4 GHz.

Training stage			Real sites		Ray tracing	
LoS	Geometry	Fine-tuning	In-band	Cross-band	In-band	Cross-band
✓			4.28	5.27	3.69	4.73
✓	✓		3.95	4.97	3.31	4.46
✓		✓	3.86	4.81	3.25	4.32
✓	✓	✓	**3.60**	**4.64**	**3.14**	**4.11**

small part of the total energy, lower accuracy on the exterior wall material has little effect on the overall predicted power.

Ablation study. We conduct the ablation study of the training pipeline of FresGRT in Sec. 4 by leaving one stage out at a time, trained at 2.4 GHz. The results shown in Tab. 4 report the MAE in dB. We can see that the performance increases gradually. The model trained with only the LoS stage performs the worst, and every added stage brings a clear improvement. Only the training that includes the full pipeline performs the best. On the real sites, the in-band error drops from 4.28 dB to 3.60 dB (a 16% improvement), and the cross-band error falls from 5.27 dB to 4.64 dB (a 12% improvement). The results on the ray-tracing datasets show the same trend, with the in-band error going from 3.69 dB to 3.14 dB (reduced by 15%) and the cross-band error from 4.73 dB to 4.11 dB (reduced by 13%). This indicates that every stage of the pipeline has its own effect, and only running all of them reaches the best performance.

Computation time. To make the rendering fast enough for real-time digital-twin applications, the ray tracing of FresGRT is implemented in CUDA. We measure the computation time of the CUDA kernels on every dataset with a single NVIDIA Quadro RTX 4000 and report the largest scenes in Tab. 5 (the rest are in Sec. F, with a breakdown of the time by component). FresGRT renders the band power at one position in 76 ms to 113 ms, meeting the 100 ms requirement for real-time WiFi tracking (Xie et al., 2019).

Table 5: Rendering time of one new position on one NVIDIA Quadro RTX 4000.

Scene	Gaussians	Faces	Subcarriers	Time
RT 1	4,092	23	51	106 ms
RT 6	4,494	22	51	113 ms
Site 1	4,226	15	112	76 ms

6 Conclusion

We proposed FresGRT, the first one-to-many-band RFDT, which learns a frequency-invariant scene representation with 3D Gaussian ray tracing. FresGRT learns the material parameters of the environment through the Fresnel equations, decouples them from the geometry using the frequency-selective fading, and optimizes both step by step with a staged training pipeline. FresGRT is evaluated on real measurements and on ray-tracing datasets. Trained on a single band, FresGRT predicts the unseen bands with a mean error of 4.4 dB on the real sites and 4.0 dB on the ray-tracing datasets, a 45% and a 52% improvement over the best baseline, while reducing the survey time by 55 %.

REPRODUCIBILITY STATEMENT

The renderer and the closed forms of its gradients are derived in Sec. A, the initialization of the Gaussians from the vision data in Sec. B, and the training stages, the losses, and the hyperparameters in Sec. C and Tab. 7. The collection of the real site datasets and the generation of the ray-tracing datasets are described in Sec. D. The code, the trained models, and the datasets are available at `https://anonymous.4open.science/r/FresGRT-061C`.

AI USE STATEMENT

In this work, we used generative AI tools for method implementation and for verifying the derivations originally derived by the authors. We have not used generative AI tools for generating synthetic data, developing the theoretical model, formulating the mathematical claims, proposing the hypotheses, designing the experiments, or interpreting the results, and translation, dataset cleaning, and qualitative data analysis are not applicable to this work. Additionally, we used generative AI tools for literature search and for correcting the grammar and improving the clarity of the text. We have reviewed all AI-assisted work. All AI-generated code was verified and tested by the authors, and every reference found by the tools was read by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Chaitanya Amballa, Yu-Lin Wei, Sattwik Basu, Zhijian Yang, Mehmet Ergezer, and Romit Roy Choudhury. Can NeRFs "See" without Cameras? In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 124182–124209, 2025. doi: 10.52202/085713-3743.

Fayçal Aït Aoudia, Jakob Hoydis, Merlin Nimier-David, Baptiste Nicolet, Sebastian Cammerer, and Alexander Keller. Sionna RT: Technical Report, 2025. URL `https://arxiv.org/abs/2504.21719`.

Xingyu Chen, Zihao Feng, Kun Qian, and Xinyu Zhang. Radio Frequency Ray Tracing with Neural Object Representation for Enhanced RF Modeling. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21339–21348, 2025. doi: 10.1109/CVPR52734.2025.01988.

Clearpath Robotics and Open Robotics. TurtleBot 4 User Manual. `https://turtlebot.github.io/turtlebot4-user-manual/`, 2022.

Jean-Marc Conrat, Mohamed Abdelbasset Aliouane, Jean-Cristophe Cousin, and Xavier Begaud. Material Permittivity and Conductivity Estimation From 2 to 260 GHz and Extension of the ITU-R P.2040 Model at Frequency Above 100 GHz. In *2024 IEEE 35th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–6, 2024. doi: 10.1109/PIMRC59610.2024.10817451.

Steve Cruz, Will Hutchcroft, Yuguang Li, Naji Khosravan, Ivaylo Boyadzhiev, and Sing Bing Kang. Zillow Indoor Dataset: Annotated Floor Plans With 360° Panoramas and 3D Room Layouts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2133–2143, 2021. doi: 10.1109/CVPR46437.2021.00217.

Pinxuan Dai, Jiamin Xu, Wenxiang Xie, Xinguo Liu, Huamin Wang, and Weiwei Xu. High-quality Surface Reconstruction using Gaussian Surfels. In *ACM SIGGRAPH 2024 Conference Papers*, SIGGRAPH '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400705250. doi: 10.1145/3641519.3657441. URL `https://doi.org/10.1145/3641519.3657441`.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981. ISSN 0001-0782. doi: 10.1145/358669.358692. URL `https://doi.org/10.1145/358669.358692`.

Francesco Gringoli, Matthias Schulz, Jakob Link, and Matthias Hollick. Free Your CSI: A Channel State Information Extraction Platform For Modern Wi-Fi Chipsets. In *Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*, WiNTECH '19, pp. 21–28, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450369312. doi: 10.1145/3349623.3355477. URL https://doi.org/10.1145/3349623.3355477.

Chun Gu, Xiaofei Wei, Zixuan Zeng, Yuxuan Yao, and Li Zhang. IRGS: Inter-Reflective Gaussian Splatting with 2D Gaussian Ray Tracing. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10943–10952, 2025. doi: 10.1109/CVPR52734.2025.01022.

Jakob Hoydis, Sebastian Cammerer, Fayçal Ait Aoudia, Avinash Vem, Nikolaus Binder, Guillermo Marcus, and Alexander Keller. Sionna: An Open-Source Library for Next-Generation Physical Layer Research, 2023. URL https://arxiv.org/abs/2203.11854.

Ting-Wei Hsu and Hung-Yun Hsieh. Robust Multi-User Pose Estimation Based on Spatial and Temporal Features from WiFi CSI. In *ICC 2024 - IEEE International Conference on Communications*, pp. 1600–1605, 2024. doi: 10.1109/ICC51166.2024.10623053.

Ting-Wei Hsu and Hung-Yun Hsieh. On Using Spatial and Temporal Features for Robust Multiuser Pose Estimation Based on WiFi CSI. *IEEE Internet of Things Journal*, 12(15):29897–29912, 2025. doi: 10.1109/JIOT.2025.3572208.

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. In *ACM SIGGRAPH 2024 Conference Papers*, SIGGRAPH '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400705250. doi: 10.1145/3641519.3657428. URL https://doi.org/10.1145/3641519.3657428.

Peter J. Huber. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964.

IEEE. IEEE Std 802.11az-2022: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 4: Enhancements for Positioning. IEEE Standards Association, 2022.

IEEE. IEEE Std 802.11be-2024: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 8: Enhancements for Extremely High Throughput (EHT). IEEE Standards Association, 2024. Multi-link operation across the 2.4, 5, and 6 GHz bands.

IEEE. IEEE Std 802.11bf-2025: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment: Enhancements for Wireless LAN Sensing. IEEE Standards Association, 2025.

ITU-R. Effects of building materials and structures on radiowave propagation above about 100 MHz. Recommendation ITU-R P.2040-2, International Telecommunication Union, September 2021.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkuehler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.*, 42(4), July 2023. ISSN 0730-0301. doi: 10.1145/3592433. URL https://doi.org/10.1145/3592433.

Leonid Keselman, John Iselin Woodfill, Anders Grunnet-Jepsen, and Achintya Bhowmik. Intel RealSense Stereoscopic Depth Cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1267–1276, 2017. doi: 10.1109/CVPRW.2017.167.

Latif U. Khan, Walid Saad, Dusit Niyato, Zhu Han, and Choong Seon Hong. Digital-Twin-Enabled 6G: Vision, Architectural Trends, and Future Directions. *IEEE Communications Magazine*, 60 (1):74–80, 2022. doi: 10.1109/MCOM.001.21143.

Mathieu Labbé and François Michaud. RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *Journal of Field Robotics*, 36(2):416–446, 2019. doi: 10.1002/rob.21831. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21831.

Fan Liu, Yuanhao Cui, Christos Masouros, Jie Xu, Tony Xiao Han, Yonina C. Eldar, and Stefano Buzzi. Integrated Sensing and Communications: Toward Dual-Functional Wireless Networks for 6G and Beyond. *IEEE Journal on Selected Areas in Communications*, 40(6):1728–1767, 2022. doi: 10.1109/JSAC.2022.3156632.

Haofan Lu, Christopher Vattheuer, Baharan Mirzasoleiman, and Omid Abari. NeWRF: a deep learning framework for wireless radiation field reconstruction and channel prediction. In *Proceedings of the 41st International Conference on Machine Learning*. JMLR.org, 2024.

N. Cameron Matson and Karthikeyan Sundaresan. Online Radio Environment Map Creation via UAV Vision for Aerial Networks. In *IEEE INFOCOM 2024 - IEEE Conference on Computer Communications*, pp. 81–90, 2024. doi: 10.1109/INFOCOM52122.2024.10621219.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I*, volume 12346 of *Lecture Notes in Computer Science*, pp. 405–421. Springer, 2020. doi: 10.1007/978-3-030-58452-8_24. URL https://doi.org/10.1007/978-3-030-58452-8_24.

Nicolas Moenne-Loccoz, Ashkan Mirzaei, Or Perel, Riccardo de Lutio, Janick Martinez Esturo, Gavriel State, Sanja Fidler, Nicholas Sharp, and Zan Gojcic. 3D Gaussian Ray Tracing: Fast Tracing of Particle Scenes. *ACM Trans. Graph.*, 43(6), November 2024. ISSN 0730-0301. doi: 10.1145/3687934. URL https://doi.org/10.1145/3687934.

Bhavya Sai Nukapotula, Rishabh Tripathi, Seth Pregler, Dileep Kalathil, Srinivas Shakkottai, and Theodore S. Rappaport. GSpaRC: Gaussian Splatting for Real-time Reconstruction of RF Channels. *arXiv preprint arXiv:2511.22793*, 2025.

Tribhuvanesh Orekondy, Pratik Kumar, Shreya Kadambi, Hao Ye, Joseph Soriaga, and Arash Behboodi. WiNeRT: Towards Neural Ray Tracing for Wireless Channel Modelling and Differentiable Simulations. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=tPKKXeW33YU.

Theodore S. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall, 2nd edition, 2002.

Norisato Suga, Yoshihiro Maeda, and Koya Sato. Indoor Radio Map Construction via Ray Tracing With RGB-D Sensor-Based 3D Reconstruction: Concept and Experiments in WLAN Systems. *IEEE Access*, 11:24863–24874, 2023. doi: 10.1109/ACCESS.2023.3254912.

Azar Taufique, Mona Jaber, Ali Imran, Zaher Dawy, and Elias Yacoub. Planning Wireless Cellular Networks of Future: Outlook, Challenges and Opportunities. *IEEE Access*, 5:4821–4845, 2017. doi: 10.1109/ACCESS.2017.2680318.

David Tse and Pramod Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, 2005. doi: 10.1017/CBO9780511807213.

Fuhai Wang, Zihan Jin, Lehang Wang, Xuehui Dong, Tiebin Mi, Robert Caiming Qiu, and Zenan Ling. Radio-Frequency Inverse Rendering for Wireless Environment Modeling. *arXiv preprint arXiv:2604.07086*, 2026.

Xiangyu Wang, Xuyu Wang, Shiwen Mao, Jian Zhang, Senthilkumar C. G. Periaswamy, and Justin Patton. Indoor Radio Map Construction and Localization With Deep Gaussian Processes. *IEEE Internet of Things Journal*, 7(11):11238–11249, 2020. doi: 10.1109/JIOT.2020.2996564.

Chaozheng Wen, Jingwen Tong, Yingdong Hu, Zehong Lin, and Jun Zhang. WRF-GS: Wireless Radiation Field Reconstruction with 3D Gaussian Splatting. In *IEEE INFOCOM 2025 - IEEE Conference on Computer Communications*, pp. 1–10, 2025. doi: 10.1109/INFOCOM55648.2025.11044513.

Chaozheng Wen, Jingwen Tong, Yingdong Hu, Zehong Lin, and Jun Zhang. Neural Representation for Wireless Radiation Field Reconstruction: A 3D Gaussian Splatting Approach. *IEEE Transactions on Wireless Communications*, 25:7490–7504, 2026. doi: 10.1109/TWC.2025.3631663.

Yaxiong Xie, Jie Xiong, Mo Li, and Kyle Jamieson. mD-Track: Leveraging Multi-Dimensionality for Passive Indoor Wi-Fi Tracking. In *The 25th Annual International Conference on Mobile Computing and Networking*, MobiCom '19, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361699. doi: 10.1145/3300061.3300133. URL https://doi.org/10.1145/3300061.3300133.

Kang Yang, Gaofeng Dong, Sijie Ji, Wan Du, and Mani Srivastava. GSRF: Complex-Valued 3D Gaussian Splatting for Efficient Radio-Frequency Data Synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 70361–70390, 2025. doi: 10.52202/085713-2119.

Zhaohui Yang, Mingzhe Chen, Yuchen Liu, and Zhaoyang Zhang. A Joint Communication and Computation Framework for Digital Twin Over Wireless Networks. *IEEE Journal of Selected Topics in Signal Processing*, 18(1):6–17, 2024. doi: 10.1109/JSTSP.2023.3347931.

Faheem Zafari, Athanasios Gkelias, and Kin K. Leung. A Survey of Indoor Localization Systems and Technologies. *IEEE Communications Surveys & Tutorials*, 21(3):2568–2599, 2019. doi: 10.1109/COMST.2019.2911558.

Haijun Zhang, Ziyang Zhang, Xiangnan Liu, Wei Li, Haojin Li, and Chen Sun. Integrated Sensing and Communication for 6G Holographic Digital Twins. *IEEE Wireless Communications*, 32(2): 104–112, 2025. doi: 10.1109/MWC.001.2400178.

Lihao Zhang, Haijian Sun, Samuel Berweger, Camillo Gentile, and Rose Qingyang Hu. RF-3DGS: Wireless Channel Modeling With Radio Radiance Field and 3D Gaussian Splatting. *IEEE Transactions on Wireless Communications*, 25:10419–10433, 2026. doi: 10.1109/TWC.2026.3652154.

Songyang Zhang, Achintha Wijesinghe, and Zhi Ding. RME-GAN: A Learning Framework for Radio Map Estimation Based on Conditional Generative Adversarial Network. *IEEE Internet of Things Journal*, 10(20):18016–18027, 2023. doi: 10.1109/JIOT.2023.3278235.

Xiaopeng Zhao, Zhenlin An, Qingrui Pan, and Lei Yang. NeRF2: Neural Radio-Frequency Radiance Fields. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, ACM MobiCom '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450399906. doi: 10.1145/3570361.3592527. URL https://doi.org/10.1145/3570361.3592527.

A Derivations

This appendix derives the closed forms used by the renderer and their gradients with respect to the scene parameters. All quantities are complex-analytic in the material parameters and smooth in the geometry away from grazing incidence, indicating that the chain rule can be applied term by term. Tab. 6 lists every learnable parameter, the quantities through which it enters the rendered power, and the equations that give its gradient. The derivations also include the effect of the interference under an uncertain face position, which gives the coherence factor and the rendering equation of Sec. 3.2 (Sec. A.3).

A.1 Line integral through a Gaussian

Let a Gaussian have center μ, rotation R with columns (e_1, e_2, e_3), and scales (s_1, s_2, s_3) (Fig. 6a), and let the ray be $x(s) = a + s\,\hat{d}$ for $s \in [0, L]$, where a is its start point, $\hat{d}$ its unit direction and L its length (the distance between the transmitter and the receiver for the direct path). In the Gaussian frame, the ray is $y(s) = y_0 + s\,\hat{v}$, where $y_0 = R^\top(a - \mu)$ is the ray origin and $\hat{v} = R^\top \hat{d}$ the ray

Table 6: Learnable parameters of FresGRT, the quantities through which each one enters the rendered power, and the equations that give its gradient.

Parameter	Enters through	Gradient equations
Occupancy ρ_g	$\ell_g \to \phi_w \to T_w$	$\partial\ell/\partial\rho$ (Sec. A.1), Eq. (27)
Face translation	$\mu_g \to \ell_g \to \phi_w \to T_w$, $\theta \to r \to \Gamma_w$, and d_r	Eqs. (4), (17) and (25)
Thickness t_w	$s_1 \to \ell_g \to \phi_w \to T_w$, $t_w \to \phi_w \to \Gamma_w$	Eqs. (17) and (30)
Materials η_b', σ_b	Eq. (31) $\to \eta \to r, \phi_w \to T_w, \Gamma_w$	Eqs. (23), (24) and (29)
Coherence parameter s_w	c_r in Eq. (6)	Sec. A.3
Gain κ	Eq. (6)	Eq. (6)

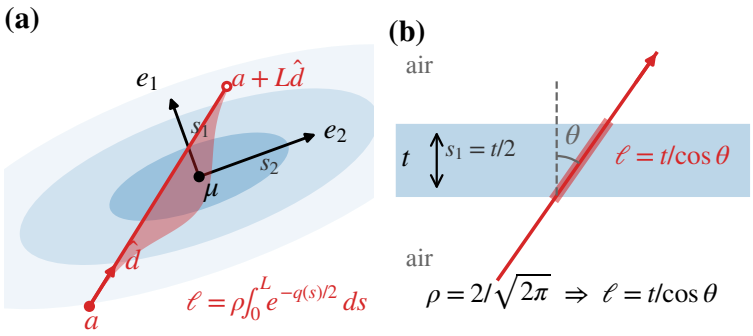

Figure 6: (a) visualization of a ray crossing a Gaussian (b) A wall of thickness t with air on both sides, represented by a Gaussian with $s_1 = t/2$ along the normal, and a ray crossing it at incidence θ, whose path inside the wall has length $t/\cos\theta$.

direction, both expressed in the frame of the Gaussian, and we calculate the quadratic form

$$q(s) = \sum_{i=1}^{3} \frac{(y_{0,i} + s\,v_i)^2}{s_i^2} = a_2 s^2 + 2b_1 s + c_0, \quad a_2 = \sum_i \frac{v_i^2}{s_i^2},\ b_1 = \sum_i \frac{y_{0,i} v_i}{s_i^2},\ c_0 = \sum_i \frac{y_{0,i}^2}{s_i^2}. \tag{13}$$

Completing the square, $q(s) = a_2\,(s + b_1/a_2)^2 + c_0 - b_1^2/a_2$, so

$$\begin{aligned} \ell = \rho \int_0^L \mathrm{e}^{-q(s)/2}\,ds &= \rho\,\mathrm{e}^{-\frac{1}{2}\left(c_0 - b_1^2/a_2\right)} \int_0^L \mathrm{e}^{-\frac{a_2}{2}(s + b_1/a_2)^2} ds \\ &= \rho\,\mathrm{e}^{-\frac{1}{2}\left(c_0 - b_1^2/a_2\right)} \sqrt{\frac{\pi}{2a_2}} \Big[\operatorname{erf}(z_L) - \operatorname{erf}(z_0)\Big], \end{aligned} \tag{14}$$

where $z_L = (a_2 L + b_1)/\sqrt{2a_2}$, and $z_0 = b_1/\sqrt{2a_2}$. The integrand is positive, meaning that $\ell \geq 0$ for every ray.

Initialize ρ for a slab. The occupancy ρ scales the density of a Gaussian, so it sets how much material a ray sees when it crosses the Gaussian. We model every object as a slab, and for a solid slab the ray should see exactly the material along its path inside the slab, whose length is $t/\cos\theta$ for thickness t and incidence θ as shown in Fig. 6b. We therefore initialize ρ so that the line integral through the Gaussian of a slab equals this length. Model the slab as one Gaussian with $s_1 = t/2$ along the normal and $s_2, s_3 \to \infty$ in the plane, so that the quadratic form reduces to its normal component $q(s) = (y_{0,1} + s\cos\theta)^2/s_1^2$, and integrate the density over the whole ray:

$$\ell = \rho \int_{-\infty}^{\infty} \mathrm{e}^{-q(s)/2}\,ds = \frac{\rho\sqrt{2\pi}\,s_1}{\cos\theta} = \frac{\rho\sqrt{2\pi}\,t}{2\cos\theta}. \tag{15}$$

Choosing $\rho = 2/\sqrt{2\pi} \approx 0.798$, we have $\ell = t/\cos\theta$, the length of the ray path inside the slab. In other words, a Gaussian with peak density $\rho = 0.798$ holds the same amount of material along the normal as a solid slab with $\rho = 1$ and thickness t, and this is the value we assign to every Gaussian of an object before training.

Gradients. The line integral of Eq. (14) is a product of three factors:

$$\ell = \rho\, A\, S\, D, \qquad A = \mathrm{e}^{-\frac{1}{2}\left(c_0 - b_1^2/a_2\right)}, \quad S = \sqrt{\frac{\pi}{2a_2}}, \quad D = \mathrm{erf}(z_L) - \mathrm{erf}(z_0), \tag{16}$$

where A is the density at the point of the ray closest to the center μ of the Gaussian, S the width of the Gaussian along the ray, and D the fraction of that width that lies within the ray segment of length L. The gradient with respect to the occupancy is $\partial\ell/\partial\rho = A\,S\,D$. For a geometry parameter $p \in \{\mu, R, s_1, s_2, s_3\}$, the product rule gives

$$\frac{\partial \ell}{\partial p} = \ell\left(\frac{\partial \ln A}{\partial p} + \frac{\partial \ln S}{\partial p}\right) + \rho\, A\, S\, \frac{\partial D}{\partial p}, \tag{17}$$

and the chain rule through a_2, b_1 and c_0 gives the three pieces

$$\frac{\partial \ln A}{\partial p} = -\frac{1}{2}\left(\frac{\partial c_0}{\partial p} - \frac{2b_1}{a_2}\frac{\partial b_1}{\partial p} + \frac{b_1^2}{a_2^2}\frac{\partial a_2}{\partial p}\right), \tag{18}$$

$$\frac{\partial \ln S}{\partial p} = -\frac{1}{2a_2}\frac{\partial a_2}{\partial p}, \tag{19}$$

$$\frac{\partial D}{\partial p} = \frac{2}{\sqrt{\pi}}\left(\mathrm{e}^{-z_L^2}\frac{\partial z_L}{\partial p} - \mathrm{e}^{-z_0^2}\frac{\partial z_0}{\partial p}\right), \tag{20}$$

and

$$\frac{\partial z_L}{\partial p} = \frac{1}{\sqrt{2a_2}}\left(L\frac{\partial a_2}{\partial p} + \frac{\partial b_1}{\partial p}\right) - \frac{z_L}{2a_2}\frac{\partial a_2}{\partial p}, \qquad \frac{\partial z_0}{\partial p} = \frac{1}{\sqrt{2a_2}}\frac{\partial b_1}{\partial p} - \frac{z_0}{2a_2}\frac{\partial a_2}{\partial p}. \tag{21}$$

Every term above is expressed through $\partial a_2/\partial p$, $\partial b_1/\partial p$ and $\partial c_0/\partial p$, which are the only derivatives left to compute. By Eq. (13), a_2, b_1 and c_0 are quadratic in y_0, $\hat{v}$ and $1/s_i$, so their derivatives with respect to s_i follow directly. Since $y_0 = R^\top(a-\mu)$ and $\hat{v} = R^\top\hat{d}$ are linear in μ and R, the derivatives with respect to μ and R follow from the same equation by the chain rule. Under the face parameterization of Eq. (1), every Gaussian member's center μ moves with the translation of the face along its normal n_w, while $s_1 = t_w/2$ is updated along its thickness, so the gradients with respect to the face translation and t_w pass through y_0 and s_1 by the same rule. When the ray misses the Gaussian, both D and the exponentials e^{-z^2} go to zero, so the gradient vanishes smoothly rather than at a hard boundary. The line integral is therefore differentiable with respect to the occupancy and the geometry of the face, and these closed forms are what the CUDA kernel implements.

A.2 Fresnel coefficients and slab response

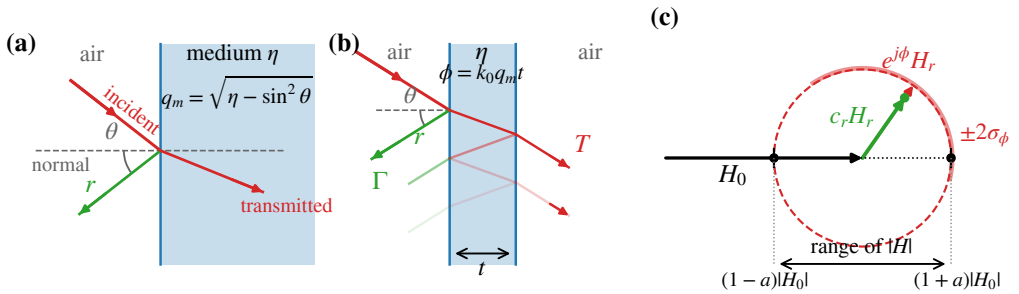

Figure 7: (a) A wave hitting a plane (b) A wave hitting a slab (c) Variation of the received power caused by the phase error of a reflection

Fresnel coefficients. The Fresnel reflection coefficient is modeled with a plane interface (considered as infinite thickness) between air and a medium of complex relative permittivity η with the wave incident from the air side at angle θ (visualized in Fig. 7a). Let $q_m = \sqrt{\eta - \sin^2\theta}$ be the cosine of the angle of refraction scaled by the refractive index $\sqrt{\eta}$, which follows from Snell's law. Thus, $k_0 q_m$ is the component of the wave vector inside the medium along the normal of the interface. We

take the square root of the complex number $\eta - \sin^2\theta$ with positive real part, which models the attenuation of the signal inside the medium. Depending on the direction of the electric field relative to the interface, the reflected signal is split into the TE polarization, whose electric field is parallel to the interface, and the TM polarization, whose electric field has a component normal to the interface. With q_m, the Fresnel reflection coefficients of the two polarizations are

$$r_{\mathrm{TE}} = \frac{\cos\theta - q_m}{\cos\theta + q_m}, \qquad r_{\mathrm{TM}} = \frac{\eta\cos\theta - q_m}{\eta\cos\theta + q_m}. \tag{22}$$

Antennas are usually vertical dipoles, causing a vertical electric field. This makes the reflection on a vertical plane TE polarized and the reflection on a horizontal plane TM polarized. We therefore assign r_{TE} to vertical objects, e.g. walls, and r_{TM} to horizontal objects, e.g. ceilings and floors. Differentiating Eq. (22) with respect to η gives

$$\frac{\partial r_{\mathrm{TE}}}{\partial \eta} = \frac{-\cos\theta}{q_m(\cos\theta + q_m)^2}, \qquad \frac{\partial r_{\mathrm{TM}}}{\partial \eta} = \frac{\cos\theta\,(2q_m^2 - \eta)}{q_m(\eta\cos\theta + q_m)^2} = \frac{\cos\theta\,(\eta - 2\sin^2\theta)}{q_m(\eta\cos\theta + q_m)^2}. \tag{23}$$

The learnable material parameters are η' and σ. We now calculate the derivatives of η with respect to them,

$$\frac{\partial \eta}{\partial \eta'} = 1, \qquad \frac{\partial \eta}{\partial \sigma} = -\frac{\mathrm{j}\,(f/f_0)^\gamma}{2\pi f \varepsilon_0}. \tag{24}$$

While the training loss $\mathcal{L}$ of Eq. (11) is real, the rendered CSI H of Eq. (3) is complex. Thus, the gradient with respect to η' is $\partial\mathcal{L}/\partial\eta' = 2\,\mathrm{Re}\big[(\partial\mathcal{L}/\partial H)^*\,\partial H/\partial\eta\big]$. The gradient with respect to σ has the same form with $\partial\eta/\partial\sigma$ in place of $\partial\eta/\partial\eta'$. The incidence angle θ changes along with the optimization of the face geometry, and the reflection gradient therefore reaches the face geometry through $\partial r/\partial\theta$. Differentiating Eq. (22) with respect to θ, with $\partial q_m/\partial\theta = -\sin\theta\cos\theta/q_m$, we obtain

$$\frac{\partial r_{\mathrm{TE}}}{\partial \theta} = \frac{2\,(1-\eta)\sin\theta}{q_m(\cos\theta + q_m)^2}, \qquad \frac{\partial r_{\mathrm{TM}}}{\partial \theta} = \frac{2\,\eta\,(1-\eta)\sin\theta}{q_m(\eta\cos\theta + q_m)^2}. \tag{25}$$

The derivations above show that the reflection coefficients are differentiable.

Slab transmission and reflection. The Fresnel coefficients above assume a medium of infinite thickness, where no wave returns from the inside of the medium. However, a real object has a finite thickness t (two interfaces to air), where a wave reflects or penetrates at each interface as visualized in Fig. 7b. The bounces between the two interfaces add coherently, causing interference, and the ITU slab model (ITU-R, 2021) models the overall transmission T and reflection Γ as

$$T = \frac{(1-r^2)\,\mathrm{e}^{-\mathrm{j}\phi}}{1 - r^2\mathrm{e}^{-2\mathrm{j}\phi}}, \qquad \Gamma = \frac{r\,(1-\mathrm{e}^{-2\mathrm{j}\phi})}{1 - r^2\mathrm{e}^{-2\mathrm{j}\phi}}, \tag{26}$$

where r is the Fresnel reflection coefficient of Eq. (22) according the polarization of the face, and $\phi = k_0 q_m t$ is the phase the wave accumulates when signal penetrates the slab. The derivatives of T and Γ with respect to r and ϕ are

$$\frac{\partial T}{\partial r} = \frac{-2r\,\mathrm{e}^{-\mathrm{j}\phi}\big(1-\mathrm{e}^{-2\mathrm{j}\phi}\big)}{\big(1 - r^2\mathrm{e}^{-2\mathrm{j}\phi}\big)^2}, \qquad \frac{\partial T}{\partial \phi} = -\mathrm{j}\,T\,\frac{1 + r^2\mathrm{e}^{-2\mathrm{j}\phi}}{1 - r^2\mathrm{e}^{-2\mathrm{j}\phi}}, \tag{27}$$

$$\frac{\partial \Gamma}{\partial r} = \frac{\big(1-\mathrm{e}^{-2\mathrm{j}\phi}\big)\big(1 + r^2\mathrm{e}^{-2\mathrm{j}\phi}\big)}{\big(1 - r^2\mathrm{e}^{-2\mathrm{j}\phi}\big)^2}, \qquad \frac{\partial \Gamma}{\partial \phi} = \frac{2\mathrm{j}\,r\,\mathrm{e}^{-2\mathrm{j}\phi}\,(1-r^2)}{\big(1 - r^2\mathrm{e}^{-2\mathrm{j}\phi}\big)^2}. \tag{28}$$

The phase depends on the material through $\partial\phi/\partial\eta = k_0 t/(2q_m)$. The material gradient of the slab therefore goes through r and ϕ,

$$\frac{\partial T}{\partial \eta} = \frac{\partial T}{\partial r}\frac{\partial r}{\partial \eta} + \frac{\partial T}{\partial \phi}\frac{\partial \phi}{\partial \eta}, \qquad \frac{\partial \Gamma}{\partial \eta} = \frac{\partial \Gamma}{\partial r}\frac{\partial r}{\partial \eta} + \frac{\partial \Gamma}{\partial \phi}\frac{\partial \phi}{\partial \eta}. \tag{29}$$

The thickness t enters T and Γ only through the phase ϕ. In the renderer, the thickness that enters T is the effective thickness $\ell\cos\theta$ that the ray actually crosses, with ℓ the line integral of Sec. A.1, while Γ uses t directly (Eq. (8)), so the thickness gradients are

$$\frac{\partial T}{\partial t} = \frac{\partial T}{\partial \phi}\,k_0 q_m \cos\theta\,\frac{\partial \ell}{\partial t}, \qquad \frac{\partial \Gamma}{\partial t} = \frac{\partial \Gamma}{\partial \phi}\,k_0 q_m, \tag{30}$$

where $\partial\ell/\partial t = \frac{1}{2}\,\partial\ell/\partial s_1$ follows from Eq. (17) with $s_1 = t/2$. For a solid slab, where $\ell = t/\cos\theta$, the gradient of T reduces to $(\partial T/\partial\phi)\,k_0 q_m$. The geometry stage of Sec. 4.2 updates the thickness with this gradient. The derivations above show that the slab transmission and reflection coefficients are differentiable with respect to the material and the thickness.

Passivity. The sum of the transmit and reflect power must not exceed the power a slab receives, i.e. $|T|^2 + |\Gamma|^2 \leq 1$. We therefore parameterize the learnable materials with two unconstrained variables $\tilde{\eta}_b$ and $\tilde{\sigma}_b$ per material group as

$$\eta'_b = 1 + \log\left(1 + \mathrm{e}^{\tilde{\eta}_b}\right), \qquad \sigma_b = \log\left(1 + \mathrm{e}^{\tilde{\sigma}_b}\right), \tag{31}$$

so that $\eta'_b \geq 1$ and $\sigma_b \geq 0$, and every material the model can learn satisfies this constraint, with equality when $\sigma_b = 0$.

A.3 Expected power under phase uncertainty

Size of the interference. We first show the impact of the phase error introduced by the path length variation. Consider a two-path scenario, where the CSI of the direct path is H_0 and the CSI of the reflected path is H_r with a phase error ϕ. The received CSI is $H = H_0 + \mathrm{e}^{\mathrm{j}\phi} H_r$ as shown in Fig. 7c, and the received power is therefore

$$|H|^2 = |H_0|^2 + |H_r|^2 + 2\,\mathrm{Re}\big(H_0^* H_r\, \mathrm{e}^{\mathrm{j}\phi}\big). \tag{32}$$

With $a = |H_r|/|H_0|$, the interference term $2\,\mathrm{Re}\big(H_0^* H_r\, \mathrm{e}^{\mathrm{j}\phi}\big)$ ranges over $\pm 2a|H_0|^2$, making the power lie between $(1-a)^2|H_0|^2$ and $(1+a)^2|H_0|^2$. For $a = 1/3$, the power is 2.5 dB above the direct path at the constructive extreme and 3.5 dB below it at the destructive extreme. This causes a power deviation of about 3 dB within a path length variation of half a wavelength (6 cm at 2.4 GHz).

Expected power. After the LoS stage, the Gaussians still carry position errors of a few centimeters, causing the phase uncertainty of reflection paths. We model this uncertainty with the learnable coherence parameter s_w of face w, the standard deviation of its position error along the normal, and compute the expected received power

$$\mathbb{E}|H|^2 = |H_0|^2 + |H_r|^2 + 2\,\mathrm{Re}\big(H_0^* H_r\, \mathbb{E}[\mathrm{e}^{\mathrm{j}\phi}]\big). \tag{33}$$

A face whose position is off by s_w along its normal displaces the reflected path by $2s_w \cos\theta_w$, where θ_w is the incidence angle on the face. The phase of a path is then k_0 times its length, indicating that phase error is $\phi = 2k_0 s_w \cos\theta_w$. Since the position error is random, we take it as a Gaussian random variable with standard deviation s_w, so ϕ is also a Gaussian random variable with standard deviation $\sigma_\phi = 2k_0 s_w \cos\theta_w$. We therefore design the expectation of the phase factor $\mathrm{e}^{\mathrm{j}\phi}$ as the coherence factor c_r:

$$c_r \equiv \mathbb{E}[\mathrm{e}^{\mathrm{j}\phi}] = \mathrm{e}^{-\sigma_\phi^2/2} = \exp\big(-\tfrac{1}{2}(2k_0 s_w \cos\theta_w)^2\big). \tag{34}$$

Then we rewrite Eq. (33) as

$$\mathbb{E}|H|^2 = |H_0|^2 + |H_r|^2 + 2c_r\,\mathrm{Re}\big(H_0^* H_r\big). \tag{35}$$

Our renderer computes the power as a coherent sum of the paths (Eq. (3)), so we express Eq. (35) in the same form. Scaling the reflection by c_r inside the coherent sum, the received power is

$$|H_0 + c_r H_r|^2 = |H_0|^2 + c_r^2|H_r|^2 + 2c_r\,\mathrm{Re}(H_0^* H_r). \tag{36}$$

Substituting Eq. (36) into Eq. (35), we obtain

$$\mathbb{E}|H|^2 = |H_0 + c_r H_r|^2 + (1 - c_r^2)\,|H_r|^2. \tag{37}$$

The first term of Eq. (37) is the interference that remains under the uncertainty, and the second term adds back the power of the reflection that c_r removed, so the total power is preserved for every $c_r \in [0, 1]$. When several reflections are present, we assume their phase errors are independent, so the interference between reflections r and r' is scaled by $c_r c_{r'}$. Following the calculation in Eq. (35) and Eq. (36), we obtain Eq. (6). c_x in Eq. (5) takes the same form with the path length error σ_x per crossing, but multiplies H_r directly, so that the power of a reflection that crosses objects is not added back. In the implementation, we only consider reflections up to 2 bounces, the coherence factor is applied to the single-bounce paths, and the two-bounce paths are always added coherently.

A.4 PHASE ERROR OF A CROSSING

The phase of the transmission coefficient T in Eq. (8) is $\phi = k_0 q_m \ell \cos\theta$, and for a full crossing $\ell \cos\theta$ is the thickness t of the object. An error δt in the thickness therefore shifts the phase by

$$\delta\phi = k_0 \,\mathrm{Re}(q_m)\, \delta t, \tag{38}$$

which is $\mathrm{Re}(q_m) \approx \sqrt{\eta'}$ times the free-space phase of the same length. For $\delta t = 1\,\mathrm{cm}$, Eq. (38) gives 48° at 2.4 GHz and 114° at 5.8 GHz for plasterboard ($\eta' = 2.73$), 41° and 97° for wood ($\eta' = 1.99$), and 67° and 158° for concrete ($\eta' = 5.24$). A reflection that crosses one object with a centimeter of thickness error therefore adds to the coherent sum with a phase that is wrong by a large fraction of a cycle at 5 GHz and above, which is why Eq. (5) attenuates it.

B SCENE INITIALIZATION

Initialization from vision data. FresGRT initializes the geometry of the Gaussians from vision data. The vision data can come from a robot running SLAM in the space, or from a panorama captured at every measurement position. In our experiments, we use depth panoramas of 8 directions per position. We back-project the depth images with their camera poses into a point cloud of the space, from which we extract the planar faces with RANSAC (Fischler & Bolles, 1981) and place the Gaussians on them. The materials of every group are initialized to $\eta' = 3$ and $\sigma = 0.02\,\mathrm{S/m}$.

Fresnel-zone gate. By the Huygens–Fresnel principle, the reflection from a plane comes mainly from the first Fresnel zone around the specular point, so a surface acts as a mirror only if it covers that zone (Rappaport, 2002). The first Fresnel zone of a transmitter–receiver pair at distance L has a diameter of $D_\mathrm{F} = 2\sqrt{\lambda L/4}$ at the midpoint between them. We take L as the median transmitter–receiver distance from the measurements in the dataset, and allow a face to reflect only if both its width and its height exceed D_F. In our implementation, faces smaller than D_F still transmit but do not reflect. For example, at 2.4 GHz and $L = 5\,\mathrm{m}$, $D_\mathrm{F} = 0.8\,\mathrm{m}$.

C TRAINING DETAILS

Table 7: Training details of each stage.

Stage	Trainable	Frozen	Paths	Loss	Iter.
Occupancy	ρ_g, κ	materials, geometry	LoS	$\mathcal{L}_\mathrm{level}$	150
Material	η'_b, σ_b, κ	ρ_g, geometry	LoS	$\mathcal{L}_\mathrm{level}$	300
Position	μ_g, t_w	materials, κ, ρ_g	LoS, 1 bounce	$\mathcal{L}_\mathrm{shape}$	500
Position (2nd pass)	μ_g, t_w	materials, κ, ρ_g	LoS, $\leq$2 bounces	$\mathcal{L}_\mathrm{shape}$	300
Fine-tuning	η'_b, σ_b, κ, s_w	geometry, ρ_g	LoS, $\leq$2 bounces	$\mathcal{L}_\mathrm{level}+\mathcal{L}_\mathrm{depth}+\mathcal{L}_\mathrm{anchor}$	400

Hyperparameters. We use Adam to train all stages described in Sec. 4 with batch size 96. The δ of the Huber loss in Eq. (9) is set to 6 dB. While the learning rate of materials, occupancy, and gain is set to 0.03 with cosine decay to $2\,\%$ within each stage of Tab. 7, the learning rate of face positions and thicknesses is 2×10^{-3} without decay.

Estimating σ_x. We set aside ten percent of the training receivers and train the LoS stage on the rest. For every set-aside pair whose direct path crosses an object, we rotate the phase of the rendered path through the object until the rendered frequency response best matches the measured one, which gives the phase error of that crossing. The standard deviation of these phase errors divided by k_0 is σ_x, which is 1.3 to 11.5 cm on the ray-tracing datasets. The set-aside receivers then return to the training set for the remaining stages. Double-bounce reflections whose path leaves the room of the transmitter and the receiver are not rendered.

D DATASETS

Our datasets include the real site datasets and the ray-tracing datasets, which we detail in Tab. 8.

Table 8: Details of our real site datasets and ray-tracing datasets.

Site	Area (m^2)	Positions	Anchors	Bands	Survey time (h)
Site 1	114	147	3	3	4.5
Site 2	95	141	3	3	4.3
Site 3	108	174	3	3	5.3
RT 1	81	719	17	4	–
RT 2	39	335	9	4	–
RT 3	32	274	8	4	–
RT 4	43	353	12	4	–
RT 5	56	490	18	4	–
RT 6	118	1053	26	4	–
RT 7	22	196	4	4	–
RT 8	42	371	11	4	–

Real site datasets. We build a platform to collect CSI measurements. The platform includes a robot that carries a Raspberry Pi and moves in the space to take the measurements. We place three anchors at random positions in the space and measure the CSI with the Nexmon CSI tool (Gringoli et al., 2019) on 40 MHz channels at 2.4 GHz, 5 GHz, and 5.8 GHz. Each band takes 30 s to measure, so one position takes 90 s. The robot also carries an RGB-D camera, which runs SLAM to localize the robot and captures a depth panorama of 8 directions at every position for the point cloud that initializes the geometry of the Gaussians, which takes around 20 s per position. Therefore, the survey time of a position is 110 s in our testbed.

Ray-tracing datasets. We take eight scenes from ZInD (Cruz et al., 2021) and generate the datasets by ray tracing with Sionna (Aoudia et al., 2025). The walls of a scene are divided into exterior walls and interior walls. We assign common ITU materials, i.e. concrete, plasterboard, wood, and glass, to these walls. For each scene, we generate three datasets whose interior walls are concrete, plasterboard, and wood, which gives 24 datasets in total. We set the maximum number of reflections to 5, enable transmission, and generate the data at 1 GHz, 2.4 GHz, 5 GHz, and 5.8 GHz. For the initial geometry of the Gaussians, we collect a panorama at every measurement position and initialize the Gaussians as described in Sec. B.

E Additional results

Overall performance. Tabs. 9 and 10 report the in-band and the cross-band errors of FresGRT and of every baseline following the testing of Tabs. 1 and 2, trained at 5.8 GHz on the real sites and at 1 GHz and 5.8 GHz on the ray-tracing datasets. For the in-band scenario, all models perform similarly. On the real sites, the largest error is 4.56 dB (EchoNeRF) when trained at 5.8 GHz, while in simulation the maximum errors are 4.46 dB (GSpaRC) when trained at 1 GHz and 5.22 dB (RFScape) when trained at 5.8 GHz. In the cross-band scenario, FresGRT again outperforms every baseline. On the real sites, FresGRT outperforms the best baseline by 39% and 35% at 2.4 GHz and 5 GHz when trained at 5.8 GHz. On the ray-tracing datasets, FresGRT outperforms the best baseline by 64%, 51% and 49% at 2.4 GHz, 5 GHz and 5.8 GHz when trained at 1 GHz, and by 69%, 64% and 51% at 1 GHz, 2.4 GHz and 5 GHz when trained at 5.8 GHz. Averaged over all cross-band testing, FresGRT reaches 4.3 dB on the real sites and 4.6 dB on the ray-tracing datasets, 37% and 58% below the best baseline.

Table 9: MAE of the band-mean power (dB) averaged on the real sites, trained at 5.8 GHz.

MAE (dB) ↓	Trained at 5.8 GHz		
Band (GHz)	5.8	2.4	5
RFScape	3.82	7.37	6.17
$NeRF^2$	**3.45**	11.25	7.32
EchoNeRF	4.56	10.17	6.48
GSpaRC	4.08	10.92	7.75
WRF-GS+	3.78	10.98	6.85
FresGRT	4.21	**4.53**	**3.99**

Table 10: MAE of the band-mean power (dB) averaged on the ray-tracing datasets, trained at 1 GHz and at 5.8 GHz.

MAE (dB) ↓	Trained at 1 GHz				Trained at 5.8 GHz			
Band (GHz)	1.0	2.4	5.0	5.8	5.8	1.0	2.4	5.0
RFScape	3.23	10.44	11.92	12.63	5.22	10.60	11.61	9.83
NeRF^2	2.85	10.12	19.11	21.13	**3.62**	21.05	11.05	9.54
EchoNeRF	4.30	13.32	22.32	24.33	4.13	13.14	10.38	8.82
GSpaRC	4.46	10.73	19.32	21.30	4.95	21.12	11.78	10.72
WRF-GS+	4.25	10.84	19.31	21.26	5.17	21.10	11.78	10.08
FresGRT	**2.37**	**3.62**	**5.79**	**6.47**	4.50	**3.29**	**3.78**	**4.35**

Learned scene representation. Following Sec. 5.2, we show the scene representation learned at 2.4 GHz on all the remaining scenes in Fig. 8, i.e. Site 3 and the 20 ray-tracing datasets that Fig. 5 leaves out. RT X-P, RT X-C, and RT X-W denote the datasets of floor plan X whose interior walls are plasterboard, concrete, and wood. FresGRT learns a frequency-transferable scene representation on every scene.

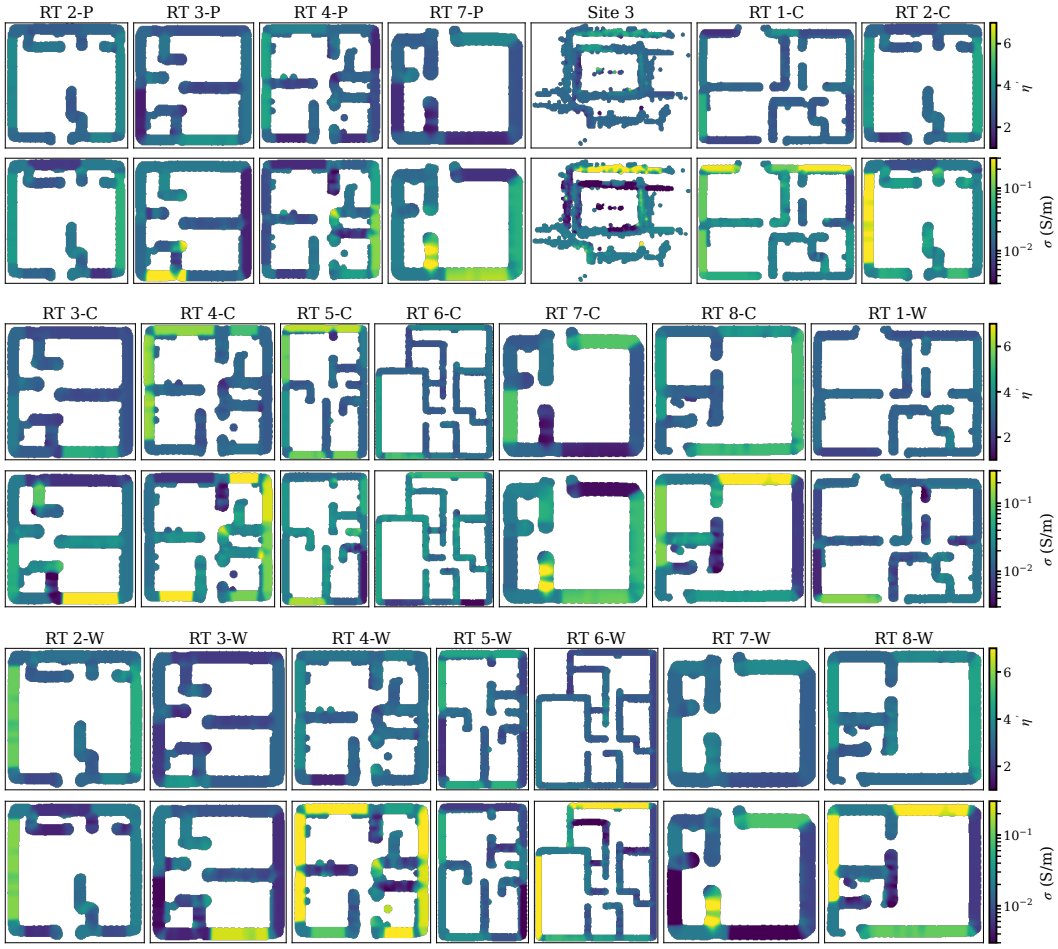

Figure 8: FresGRT's learned scene representation of Site 3 and the 20 remaining ray-tracing datasets. Top row of each block: the relative permittivity η'. Bottom row of each block: the conductivity σ.

F Rendering Time

Rendering time on different scenes. Extending the experiment of Tab. 5, we list the rendering time of every scene in Tab. 11. We test FresGRT on one NVIDIA Quadro RTX 4000. The rendering

time of one position is about 100 ms on every scene, which meets the computation time of a real-time system in the literature (Xie et al., 2019).

Rendering time breakdown. We further analyze the computation time break down in Tab. 12. We categorize the components of rendering into three parts:

- Direct path: the computation time of rendering direct path signal is from computing the line integral of Eq. (7) from the transmitter to the receiver. It is one segment per position and takes 13 to 27 ms.
- Reflections: it measures the two line integrals of every reflecting face, from the transmitter to the reflection point and from the reflection point to the receiver. It is $2W$ segments per position for W reflecting faces, taking 38 to 69 ms.
- Other: besides direct path and reflections, FresGRT needs to compute the mirror geometry, the reflection coefficients, and the coherent sum of Eq. (6). It takes 1 to 12 ms.

The reflections take 61% to 76% of the time, because every segment costs about the same closed-form line integral and the reflections need $2W$ segments per position while the direct path only needs one.

Table 11: Rendering time of FresGRT on every scene for one new receiver position and for every transmitter–receiver pair of the dataset (count in parentheses), with all reflections.

Scene	Gaussians	Faces	Subcarriers	One position	All pairs
RT 1	4,092	23	51	106 ms	3.43 s (12,223)
RT 2	2,635	17	51	82 ms	490 ms (3,015)
RT 3	2,277	21	51	72 ms	500 ms (2,192)
RT 4	2,176	22	51	89 ms	1.27 s (4,236)
RT 5	3,188	15	51	93 ms	1.58 s (8,820)
RT 6	4,494	22	51	113 ms	9.08 s (27,378)
RT 7	1,637	14	51	64 ms	100 ms (784)
RT 8	2,662	18	51	82 ms	500 ms (4,081)
Site 1	4,226	15	112	76 ms	200 ms (441)
Site 2	3,346	8	112	66 ms	120 ms (423)
Site 3	4,196	19	112	97 ms	350 ms (522)

Table 12: Computation time break down of each component

Scene	Direct path	Reflections	Other	Total
RT 1	27 ms (27%)	65 ms (65%)	9 ms (9%)	100 ms
RT 2	23 ms (29%)	49 ms (62%)	7 ms (9%)	79 ms
RT 3	22 ms (27%)	51 ms (63%)	8 ms (10%)	81 ms
RT 4	27 ms (26%)	69 ms (66%)	7 ms (7%)	104 ms
RT 5	20 ms (24%)	54 ms (64%)	11 ms (13%)	84 ms
RT 6	26 ms (25%)	69 ms (65%)	12 ms (11%)	106 ms
RT 7	19 ms (29%)	40 ms (61%)	6 ms (9%)	66 ms
RT 8	21 ms (26%)	52 ms (65%)	7 ms (9%)	80 ms
Site 1	13 ms (22%)	46 ms (77%)	1 ms (2%)	60 ms
Site 2	19 ms (33%)	38 ms (66%)	1 ms (2%)	58 ms
Site 3	24 ms (28%)	61 ms (70%)	1 ms (1%)	87 ms