RF Genesis: Zero-Shot Generalization of mmWave Sensing through Simulation-Based Data Synthesis and Generative Diffusion Models

Xingyu Chen
University of California San Diego
xic063@ucsd.edu

Xinyu Zhang
University of California San Diego
xyzhang@ucsd.edu

ABSTRACT

This paper presents RF Genesis (RFGen), a novel and cost-effective method for synthesizing RF sensing data using cross-modal diffusion models, in order to improve the generalization capability of millimeter-wave (mmWave) sensing systems. Traditional machine learning models used in mmWave sensing struggle with limited training datasets. Their performance degrades drastically when confronted with *unseen* users, environments, sensor configurations, test classes, *etc.* RFGen mitigates these challenges by using a cross-modal generative framework to synthesize and expand mmWave sensing data. We specifically propose a custom ray tracing simulator to simulate RF propagation and interaction with objects/environments. We then leverage a set of diffusion models to generate massive 3D scenes, and transform the visual scene representation into the corresponding mmWave sensing data, under the direction of application-specific "prompts". Our proposed approach reconciles the physics-based ray tracing with the black-box diffusion model, leading to accurate, scalable, and explainable vision-to-RF data synthesis. Our extensive real-world experiments highlight RFGen's effectiveness in diverse mmWave sensing applications, enhancing their generalization to unseen test cases without laborious data collection.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Computer systems organization** → *Embedded and cyber-physical systems.*

KEYWORDS

Simulation, Generalization, Generative Diffusion Models, Millimeter Wave Sensing

ACM Reference Format:
Xingyu Chen and Xinyu Zhang. 2023. RF Genesis: Zero-Shot Generalization of mmWave Sensing through Simulation-Based Data Synthesis and Generative Diffusion Models. In *ACM Conference on Embedded Networked Sensor Systems (SenSys '23), November 12–17, 2023, Istanbul, Turkiye.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3625687.3625798

SenSys '23, November 12–17, 2023, Istanbul, Turkiye

ACM ISBN 979-8-4007-0414-7/23/11.
https://doi.org/10.1145/3625687.3625798

1 INTRODUCTION

Wireless sensing is a rapidly emerging technology that is permeating into human life, from research prototypes to standardization and commercialization. Millimeter-wave (mmWave) sensing is gaining particularly strong traction in recent years, owing to the relatively high angular/range resolution from large antenna arrays and wide sampling bandwidth. Beyond the classical distance/speed detection, recent mmWave sensing applications are demonstrating vision-like capabilities, such as gesture/posture tracking, person reidentification, point cloud generation, [4, 12, 26, 39, 43, 44, 59], *etc.*, often driven by deep learning models.

One common challenge for such applications lies in *generalization.* Due to their short wavelengths and coherent combination of multipath reflections, RF signals are highly sensitive to tiny changes in the sensing target. For example, prior work [29] and our own experiments (Sec. 10) reveal that gesture-sensing data can vary dramatically as the subjects vary their orientation. Unfortunately, deep learning models often struggle with unseen data. Collecting massive datasets may mitigate the problem, but it poses unique hurdles. Firstly, it calls for specialized, often costly equipment, which could be prohibitive for everyday applications. Secondly, the data collected is usually specific to the radar configuration under which it was gathered, making the transfer of data between varying applications or scenarios problematic. This complexity amplifies the challenge of generalizing mmWave sensing models. We believe the generalization capability stems from the ability to generate a highly diverse and virtually unlimited dataset, something no existing real-data collection method could feasibly achieve.

Using simulators to synthesize data has proven to be a promising strategy, with extensive validation in the fields of computer vision and graphics [32]. However, this scheme has not been widely used in the RF field. Synthesizing data with RF simulators (*e.g.*, ray tracer or electromagnetic synthesis) requires an abundance of scene information, including the geometry, motion, material properties of objects, and environmental context. These details typically require expensive equipment and laborious efforts. The lack of high-precision scene information also exacerbates the sim-to-real gap.

In this paper, we propose RF Genesis (henceforth referred to as RFGen), a novel mmWave sensing data synthesis method to overcome the generalization challenge. As illustrated in Fig. 2, RFGen builds on two high-level design principles: *(i)* Using a generative framework, a cross-modal diffusion model in particular, to generate massive RF sensing data from vision and graphics datasets. *(ii)* Integrating a white-box physics-based mmWave ray tracing simulator with the black-box generative model to represent realistic RF signal

propagation and interaction with the scene. By utilizing and diffusing existing vision datasets, RFGen significantly curtails the cost of RF sensing data collection, and enables existing mmWave sensing models to be more adaptable and robust in practical scenarios.

To bring the two design principles to fruition, we have to grapple with non-trivial technical challenges.

(i) Massive vision-to-RF data synthesis. The simulation of scene-specific RF sensing signals entails massive labeled 3D vision datasets as inputs. However, existing 3D vision datasets still bear limited diversity and coverage. For instance, no existing human posture dataset [42] simultaneously covers a large number of participants, postures, or scenes, whereas tiny variations across these dimensions may lead to drastically different RF signal patterns. To overcome this challenge, we propose a set of diffusion models to diversify the vision data to a virtually unlimited set of objects and environments. We further use an RF sensing prompt method to steer the diffusion models toward the scenes of most interest to each sensing application.

(ii) Generative modeling of RF signals across massive visual scenes. Whereas it is feasible to use deep learning models to directly render RF signals (similar to computer graphics [37]), the rendering model itself entails massive ground-truth RF data for training. We instead custom-build a physics-based ray tracing simulator, which is explainable and can leverage the sparse nature of the mmWave channel [50] to attain high efficiency. In particular, we observe that, unlike electromagnetic simulation which entails wavelength precision, mmWave sensing applications only need to preserve the general spatiotemporal features of signals.

Unfortunately, the ray tracing based simulation of multipath reflection is still limited due to the nuances in material properties and internal structures. We thus propose a *path diffusion* model to rectify such imperfections. Yet the deep learning-based diffusion does not bode well with the physics-based outputs from ray tracing. We propose a *path-based intermediate representation* of the sensing signals, which enables the generation of structured multipath noise from the diffusion models, based on an *RF path low-rank adaptation* mechanism.

We have implemented the RFGen framework and conducted a comprehensive evaluation using off-the-shelf mmWave radar sensors. Our microbenchmark experiments demonstrate that the mmWave sensing signals generated by RFGen can accurately approximate the ground-truth, in terms of both raw signal structures, point cloud, and multipath noise. Given the flexibility of RFGen, it can be adapted to synthesize training data and augment the generalization capability of a variety of mmWave sensing applications. To verify its effectiveness, we conduct two case studies, *i.e.*, mmWave-based posture recognition and gesture sensing. Our experiments using off-the-shelf radar sensors indicate that RFGen can enable existing sensing models to work reliably across *unseen* postures/gestures, environments, subjects, and their orientations, *etc.* In contrast, the accuracy of state-of-the-art sensing systems degrades to an unusable level (down to around 20%) in such diverse scenarios.

Our contributions can be summarized in the following three points:

- We propose RFGen, a novel and generic solution for transitioning and expanding existing vision/graphics datasets to the wireless sensing domain. This approach is viable, low-cost, and adaptable to a wide array of scenarios.
- We develop a novel mmWave sensing signal synthesis framework that integrates physics-based ray tracing and black-box diffusion models. This combination facilitates the accurate and adaptive simulation of mmWave echo signals.
- We conduct extensive evaluations of our framework across real-world radar platforms and sensing applications. These tests underscore the framework's effectiveness and its potential for widespread use in diverse fields.

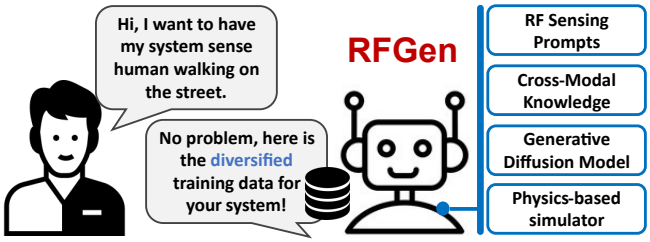

Figure 1: RFGen can generate diverse training data for RF sensing applications to improve generalization, based on the description of the target sensing application and its large cross-modal knowledge.

2 BACKGROUND AND RELATED WORK

Generalization of mobile and RF sensing. In the realm of machine learning, **generalization** refers to a model's capability to accurately perform inference on *new, unseen testing data* [3]. It entails identifying underlying patterns in limited training datasets without succumbing to overfitting. This generalization challenge is more prominent in the context of RF sensing due to the elusive complexities of wireless signals. Take mmWave sensing as an example, the reflection signal is incredibly sensitive to micro-displacements [57], micro-movements [16, 30, 69], diffraction, and multipath reverberation around the subject [58, 69]. So even the same action repeated by the same subject may result in highly variable signal patterns, not to mention cross-subject or cross-environment. Our empirical study (Sec. 9) indicates that state-of-the-art mmWave human posture/gesture recognition systems [29, 61] are unable to correctly recognize the *unseen* postures or gestures. Their performance degrades significantly even for the same posture/gesture performed in different environments, by different users, or at different relative orientations between the user and the radar sensor.

Existing efforts in addressing the generalization problem of sensing applications can be largely categorized into two areas: *(i)* Improving the model performance [60, 62], such as enabling them to discern important patterns within small data samples or providing them with insights based on physical laws for a more accurate and mathematical understanding of the target state. *(ii)* Generating more diversified data for existing algorithms to learn from [9, 54, 65, 68]. Methods of data synthesis or augmentation exist for various modalities such as images, IMUs, ultrasound, and mmWave. However, these techniques often involve converting data from one modality to another or have stringent input requirements, leaving the customization of the required data an open challenge. RFGen

offers a more cost-effective and efficient solution for generating the elusive mmWave sensing data using generative diffusion models which can be easily customized through prompts.

Specific to the wireless sensing area, two approaches to enhancing generalization have been explored recently. *(i) Data augmentation.* Beyond basic transformations such as shifting or rotating signal representations akin to image operations [9], the primary method involves the addition of noise [54]. Whereas the former falls short of representing the channel distortions of wireless signals, the latter struggles in creating structured noise (to accommodate the impacts of environmental multipath). *(ii) Neural style transfer (NST).* Recent works attempt to use NST to transfer clean, simulated signal data into real-world environments [1, 53]. However, conventional NST methods, like GANs, typically only transfer data to previously trained environments and fail in *unseen* environments. Moreover, training such NST models necessitates vast amounts of pre-training data, which is costly and prone to the generalization problem by itself.

Generative Diffusion Models. Recent advancements in diffusion models have broadened their applications beyond image synthesis [6, 10]. Examples include the generation of sound[64], human actions [52], 3D models [38], and more. They have even been employed in scientific disciplines like protein design [56]. Despite these strides, there is a noticeable gap in research regarding diffusion models' applications in the RF domain. RFGen marks a notable step in identifying the significance of the diffusion model for overcoming the notorious generalization problem in wireless sensing.

Simulation-based cross-modal data synthesis. Simulation-based data synthesis usually takes a physical scene as input and generates desired sensing modalities, such as motion sensor [27], audio [7], Lidar [33], *etc.* Notably, Vid2Doppler [1] proposed using video data of the human body to simulate Doppler signals. Midas [11] built upon this by generating more comprehensive and accurate radar data from videos. However, these methods can only generate scenes from existing video data, and lack understanding and transformation of semantics and context, making application-specific customization difficult. This limitation makes them inexplicable in solving the generalization problem. Furthermore, they usually require the videos to be stable and sufficiently illuminated and are vulnerable to the artifacts from 3D reconstruction algorithms (*i.e.*, algorithms that imperfectly rebuild the target mesh model from videos). Simulating RF propagation requires complex and accurate scene information, including geometry, material properties, movement information, and even the internal structure of objects. Existing cross-modal simulation methods often struggle to obtain such information. In contrast, RFGen leverages a generative diffusion model to proactively generation full-scene information, making the generated data more accurate scalable, customizable, and cost-effective.

3 SYSTEM OVERVIEW

RFGen is a scalable and generic framework that leverages diffusion models learned from existing vision datasets, adapting them to the RF domain, thus generating high-fidelity training data for wireless sensing. The ultimate goal is to enhance the generalization capabilities of wireless sensing models.

The end-to-end workflow is shown in Fig. 2. A user defines the target sensing application through **RF Sensing Prompts**, which outline the potential actions of the target and its environment. RFGen first processes the prompts using an **Object Diffusion** model which, based on its pre-trained knowledge, generates a diverse range of object motions and 3D mesh models. These are subsequently fed into a **Path Tracer**, which simulates RF propagation following physical laws and produces a set of traced paths that interact with the target. Simultaneously, the prompts are also processed by an **Environment Diffusion** model to generate various environmental multipath representations. Such path-traced and diffused multipath data are then fused by the **Path Diffusion** model to further enhance realism. Notably, RFGen operates on a *hybrid model* that merges the white-box physical law models with black-box deep learning models. To enable such reconciliation, RFGen utilizes **RF LoRA** as the communication layer, integrated into the diffusion models. This integration ensures the conversion of the model's output into a newly designed universally compatible format (*i.e., path-based intermediate representation*) between the white-box and black-box models. Finally, the path information is injected into a *signal generator* which computes the resulting radar received signals. These signals can then be used as augmented training data to enhance the generalization of existing mmWave sensing models.

4 PHYSICAL SIMULATION MODELS

The physics-based RF simulator takes the scene information from the **Object Diffusion** and simulates the radar signal's propagation and interaction with the objects, subsequently outputting the echo signal received by the radar. More specifically, the RF simulator employs a **Path Tracer** to emit a set of rays from the radar's location, and simulates the rays' interaction with objects in the scene (*e.g.*, through reflection), taking into account the material and geometrical properties of the objects. The resulting path information is mixed with the environmental path generated by the diffusion model. This collected path information is eventually integrated and processed by the **Signal Generator** to compute the final raw signal. We now elaborate on the two building blocks of the RF simulator.

4.1 Path Tracer

The **Path Tracer** is essentially a ray tracing algorithm. Since ray tracing involves emitting a massive number of rays in each frame to calculate the intersection and interaction of electromagnetic waves with all geometric bodies in the scene, it can easily become a computational bottleneck. Consequently, an efficient and accurate path tracer is of paramount importance.

Existing ray tracers for RF simulation attempted to curtail the computation cost by merely tracing key nodes on specific objects [21]. However, this method not only compromises accuracy but also poses challenges when adapting to other objects or adding new objects to the scene. Other RF ray tracers, often based on optical ray tracing [47], support arbitrary objects but emit rays uniformly in space with a limited total number, akin to pixel sampling in images. This method led to the excessive sampling of the background environment, resulting in non-contributing samples and the wastage of most rays. Furthermore, it causes an insufficient sampling of the objects, and hence inaccurate representation of the RF echo signal

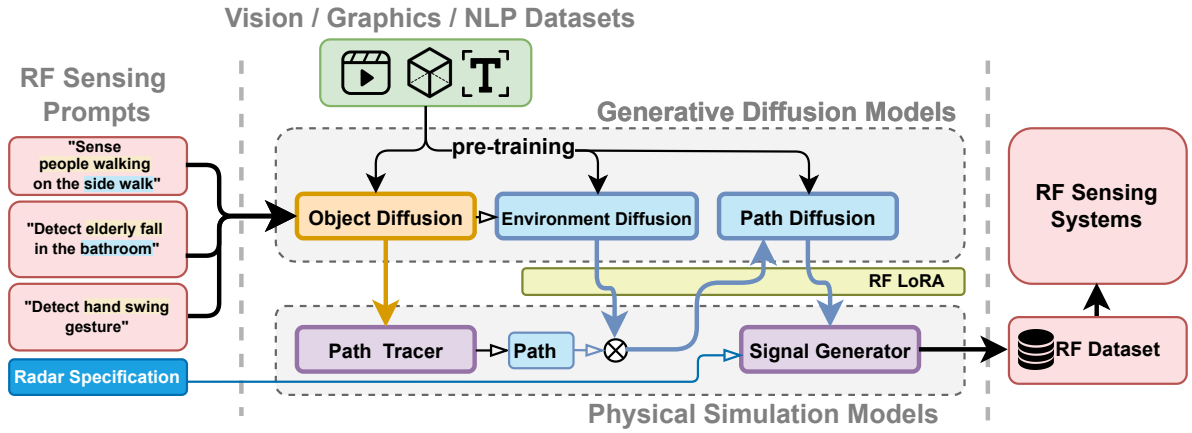

Figure 2: The system overview for RFGen contains Reconstructor and Simulator modules. RFGen explores viable and generic mmWave radar synthesis from existing video datasets.

response.

Material Properties: We incorporate the Fresnel coefficients as a robust mathematical model to simulate the behavior of radio frequency (RF) propagation in different materials. These coefficients are instrumental in calculating the proportion of electromagnetic radiation that is reflected and transmitted when an RF signal encounters an interface between two media. The significance of these coefficients is particularly apparent in RF propagation simulations, where the properties of materials heavily influence signal strength and path. The Fresnel equations, which stem from Maxwell's equations, are crucial in determining how an RF signal reacts at the boundary between two different media. These equations calculate the reflection and transmission coefficients by considering the incident angle, the RF signal's polarization state, and the relative permittivity and permeability of the two media.

In the simulation, we assign specific relative permittivity and permeability values to each material present in the scene. These parameters are vital in determining the Fresnel coefficients, thus enabling us to model a wide array of materials with reasonable accuracy. This approach includes non-conductive materials like wood and glass and conductive ones like metals. It is important to note that this is a simplification, but it has proven to be an effective way to capture the behavior of different materials. The permittivity can be calculated as [17]:

$$\epsilon = \epsilon_r \epsilon_0 - \frac{j\sigma}{\omega}, \qquad \eta = \sqrt{\frac{\mu_0}{\epsilon}}, \tag{1}$$

where ϵ is the complex permittivity. η is the wave impedance. ϵ_r is the relative permittivity of the material. ϵ_0 is the permittivity of free space. j is the imaginary unit. σ is the conductivity of the material. ω is the angular frequency. μ_0 is the permeability of free space.

As the RF signal, or ray, in our simulator, encounters a boundary, we utilize the Fresnel equations to ascertain the proportion of the ray that is reflected and that which is transmitted into the new medium. This process allows us to adjust the intensity of the reflected and transmitted rays, which in turn influences the signal strength detected by the receiver in our simulation. The Fresnel-based material approximation model within our ray tracing simulator offers a balance between computational efficiency and accuracy. It successfully handles the fundamental phenomena of reflection, transmission, and absorption that occur in real-world RF propagation scenarios, particularly in indoor environments where the RF signal often interacts with multiple different materials.

$$\begin{aligned} \cos\theta_i &= -\mathbf{i}\cdot\mathbf{n}, & \sin\theta_i &= \sqrt{1-\cos^2\theta_i}, \\ \sin\theta_t &= \sqrt{\epsilon_r}\sin\theta_i, & \cos\theta_t &= \sqrt{1-\sin^2\theta_t}, \end{aligned} \tag{2}$$

where $\mathbf{i}$ is the incident direction vector, $\mathbf{n}$ is the normal vector, θ_i is the incident angle, θ_t is the transmission angle. After calculating the incident direction of the ray, we can calculate the reflection coefficient of the surface:

$$r_p = \frac{\eta\cos\theta_i - \eta\cos\theta_t}{\eta\cos\theta_i + \eta\cos\theta_t}, \tag{3}$$

$$r_s = \frac{\cos\theta_i - \eta\cos\theta_t}{\cos\theta_i + \eta\cos\theta_t}, \tag{4}$$

$$r = \frac{r_p + r_s}{2}, \tag{5}$$

where r_p is the reflection coefficient for parallel polarization, and r_s is the reflection coefficient for perpendicular polarization.

RF Adaptive Sampling: Traditional ray tracing frequently encounters a primary computational challenge: the requirement to evaluate an immense number of rays. This stems from ray tracing's reliance on Monte Carlo sampling[49], where increased sampling typically leads to heightened accuracy and precision [49]. However, a surprising number of these rays do not make a meaningful contribution to the final result.

To address this trade-off between accuracy and computational cost inherent in traditional ray tracing, we introduce the *RF adaptive sampling* technique, In our approach, instead of uniformly emitting rays into the scene, we prioritize emitting additional rays from the *edges* of objects. This process begins by uniformly sending an initial batch of "guiding rays" into space. Edge detection is then performed based on the distance information returned by these guiding rays. We define "edges" as regions where any two adjacent rays exhibit a

propagation distance difference greater than 5%. This distinction is crucial because when sampling at edges and wedges, adjacent rays are more likely to intersect different faces and locations.

As a result, we emit more rays toward these detected edge and wedge areas. This technique not only enhances the utilization of rays but also boosts efficiency, mitigating the traditional computational bottlenecks. Furthermore, by leveraging GPU parallel processing and acceleration structures such as the Bounding Volume Hierarchy (BVH) [13, 18]. The outcome is an impressively swift simulation speed at the millisecond level.

4.2 Signal Generator

After obtaining path information from the Path Tracer and Diffusion Models, we use the **Signal Generator** to calculate the final signal received by the radar. The path information includes all the interaction information (*e.g.*, incident angle, normal vector) with objects' surfaces, and we need this to calculate the impact on electromagnetic waves. Calculating the exact scattered field would involve solving Maxwell's equations, which is impractical to compute for large or complex objects. Therefore, we employ the Physical Optics Integral (POI) method [20] to simplify the problem to an integral over the object's surface, which is much more computationally efficient. The Physical Optics approximation provides a reasonably accurate representation of the scattered field for many practical scenarios, especially when the object is much larger than the wavelength of the incident wave. We model the material's reflection coefficient using Fresnel coefficients [23, 63]. To specify material properties (*e.g.*, permittivity and conductivity of the human body), we obtain the information from publicly available material databases [34, 41] and directly input it into the target object's initial base mesh model. As the rays encounter an object's surface, we use the classical Fresnel equations to ascertain the proportion of the ray that is reflected versus transmitted into the new medium.

Finally, these accumulations are incorporated into the calculation of the signal received by the radar. For the widely used FMCW radar [24], the received signal is modeled as:

$$S_{IF_s}(t) = \sum_{i=0}^{N} A(\alpha, \gamma) exp(2\pi j(\mu t\tau + f_c\tau)), \tag{6}$$

where N is the number of rays. $A(\cdot)$ is the antenna gain pattern, parameterized by the spherical angles α (azimuth) and γ (elevation). f_c is the carrier frequency. and μ is the frequency slope calculated by $\mu = \frac{B}{T}$ where B is the signal bandwidth and T the chirp duration. The signal delay $\tau = \frac{d}{c}$, where d is the ray path length and c the light speed.

5 GENERATIVE DIFFUSION MODELS

RFGen uses diffusion models for the dual purpose of generating 3D scenes and improving the quality of simulated RF signal data. The diffusion models belong to a class of latent variable models trained using variational inference. These models strive to understand the latent structure of a dataset by mimicking the diffusion of data points through the latent space. This innovative approach provides a fresh perspective on the image generation process. Rather than relying on conventional methods, diffusion models deconstruct the image generation process into numerous smaller "denoising" stages. This iterative method enables the model to incrementally refine its output, establishing a self-correcting mechanism that culminates in high-quality samples.

The fundamental principle of diffusion models is simple yet effective. They commence with an input image, which is progressively infused with Gaussian noise in a sequence of steps, known as the forward process.

$$q(x_t^{1:N}|x_{t-1}^{1:N}) = \mathcal{N}(\sqrt{\alpha_t x_{t-1}^{1:N}}, (1-\alpha)I), \tag{7}$$

Whereby, N is the dimensionality of the state vectors. The conditional probability distribution of the states at time t given the states at the previous time step $t-1$ is represented by $q(x_t^{1:N}|x_{t-1}^{1:N})$. The mean of the Gaussian distribution, which drives the model, is given by $\sqrt{\alpha_t x_{t-1}^{1:N}}$. Here, α_t denotes the influence of past states on the current state. The covariance matrix of the distribution is $(1-\alpha)I$, where I is the identity matrix implying the independence of each state. The α_t controls the variance of the Gaussian distribution and in turn the influence of past states on future states.

Subsequently, a neural network is trained to recover the original image by reversing this noising process - a step referred to as the reverse diffusion process. The ability to simulate this reverse process empowers the model to generate new data.

Diffusion models boast versatility, finding utility in various tasks such as image denoising, inpainting, super-resolution, and image generation. Moreover, it can also be applied to non-image data generation, such as physics simulation and human motion generation [52], showing huge potential for cross-modality capability. However, despite their strong capacity for knowledge transfer across modalities, diffusion models currently struggle to apply or transition into the RF domain. This challenge arises from the nature of RF signals - a type of wireless signal which is nonhuman sensory sensing and usually lacks pixel-like structural information, making it highly dimensional and complex. Hence, transferring diffusion models into the RF domain is a significant challenge.

To address this, RFGen integrates the physics-based RF simulator with a diffusion framework. Initially, we use the diffusion model to generate object models and their corresponding motions that are suitable for the simulator (*i.e.*, **Object Diffusion**). This allows us to accurately simulate the RF echo signal emanating from the objects of interest. Subsequently, we adapt the image diffusion model to generate environmental noise (*i.e.*, **Environment Diffusion**), which is ultimately used to enhance the overall outcomes from the simulator (*i.e.*, **Path Diffusion**). We accomplish this by devising a new **Path-based Representation** along with **RF path LoRA**, which are innovative mechanisms capable of adapting image diffusion models to the RF domain. Importantly, these mechanisms do not undermine the comprehensive knowledge extracted from large models.

5.1 RF Sensing Prompts

Prompts serve as vital inputs in the context of large language models like ChatGPT [40] and Stable Diffusion, effectively setting the tone for the model's subsequent response. Whether it be a singular word, a sentence, or an elaborate paragraph, a prompt instructs the model on the required generation. The model, in response, crafts a continuation that attempts to mirror the prompt's style, tone,

and subject matter. In essence, the prompt plays a crucial role in determining the quality and relevance of the model's output, where a well-formulated and specific prompt is likely to yield superior results over a vague or ambiguous one.

In RFGen, the text input used to describe the sensing target and its context is referred to as **RF Sensing Prompts**. Such RF Sensing Prompts describe the sensing target type, the target's movements or activities, the ambient environment, as well as supplementary information. RF Sensing Prompts can take the form of a full sentence or a series of keywords separated by commas. Unlike prompts targeted for a conversation or image generation task, the RF Sensing Prompts are fed into multiple diffusion models designed for various tasks with different internal designs. As a result, for RFGen to effectively generate the accurate data, it's essential that users offer a thorough and clear description of the specific RF sensing application they're targeting.

The text-based RF Sensing Prompts is then converted into a vector representation, referred to as an "embedding", to capture the semantic information contained in the texts. At each step of the diffusion process, the model makes a prediction for the next step based on both the current state of the subject image/motion and the text prompt's embedding. This way, the text prompt's semantic information guides the generative process, causing the final output to be a natural and logical visualization/interpretation of the text prompts.

5.2 Object Diffusion

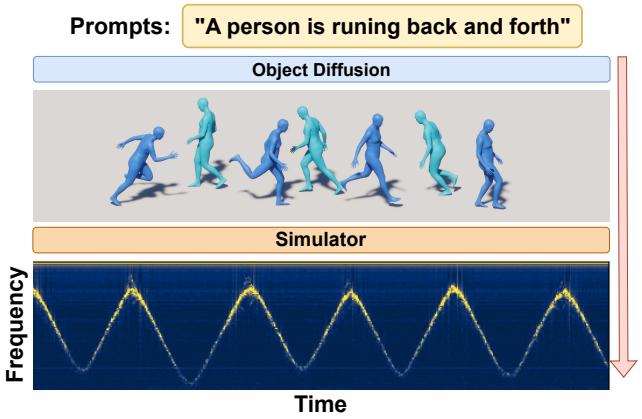

Figure 3: The object diffusion model takes the prompts as input and outputs the target object with the corresponding motion that can be directly used by the simulator.

The **Object Diffusion** takes RF sensing prompts as input and outputs a target object with corresponding shapes and movements as shown in Fig. 3. Typically, the target object is represented explicitly, often using mesh models equipped with corresponding skeletal joints. The object diffusion computes the mesh and movements to meet the requirements specified in the prompts. Subsequently, this output, the computed mesh model, and movements are fed into the aforementioned **Path Tracer** to calculate the signal paths.

5.3 Physics-Diffusion Models: Communication and Integration

RFGen's physical simulation and deep learning modules are closely interconnected at every stage. This marks a clear contrast to existing works that use pure physical simulation [31] or pure deep learning [48], or that employ physical simulation first, followed by refinement using deep learning [1]. This necessitates stricter requirements for module design, mainly because physical simulation is a "white-box" process with explicit information expression, while machine learning modules are "black-box" processes with generally implicit intermediate information.

We break the tension by creating a *physics-diffusion communication layer*, comprised of a **Path-based Intermediate Representation (PIR)** and **RF LoRA**. As shown in Fig. 4, PIR is a bi-directionally compatible image-format representation of RF signals, used to express the information required in the process of simulating RF propagation. Meanwhile, LoRA is a plugin that can transform the existing pre-trained diffusion models to adapt to PIR.

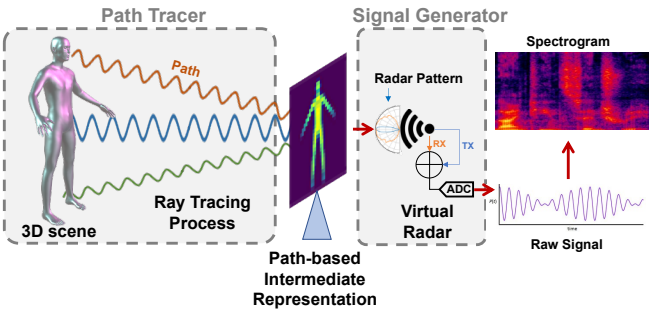

Figure 4: The Path-based Intermediate Representation is a hardware-independent representation of the scene and can be directly used by the signal generator to calculate the final RF signal.

5.3.1 Path-based Intermediate Representation. The RF simulator's ability to simulate multipath reflections is still limited by the difficulties in obtaining fine-grained scene materials and internal structure information. The problem can be circumvented by limiting the simulation to line-of-sight first-order reflections, because other multipath signals often become too faint to discern, morphing into a form of noise. Nevertheless, this multipath noise can impact the generalization of RF sensing systems across different environments.

In RFGen, we aspire to use the diffusion models to generate structured noise corresponding to a variety of environments to improve generalization. While RF signals can be presented in formats (*e.g.*, Spectrogram) similar to images, the meaning of a "pixel" within a matrix differs drastically between RF and visual signals. Thus, it is challenging to use large diffusion models pre-trained on images to directly generate RF signals. Furthermore, the lack of structured spatial information in the spectral image makes signal enhancement more difficult.

To address this gap, we propose an RF signal representation method called **Path-based Intermediate Representation (PIR)**. This representation is an image-like format, *i.e.*, a 2D array of elements, where each element (*i.e.*, pixel) represents an area in space.

However, unlike traditional images that express visible light intensity through RGB channels, we *use three channels to denote energy, time of arrival (ToA), and velocity of intersected objects in RF signal paths.* The energy channel signifies the power of the path starting from the transmitter (Tx), undergoing attenuation, absorption, and reflection, and finally received by the receiver (Rx). The ToA channel represents the total duration the path took from being emitted by the Tx to being received by the Rx. It is noteworthy that while ToA images are similar to depth images, they also account for multipath reflections and sum up the total path duration. The velocity channel represents the linear velocity of the object in contact with the path at the ray angle, crucial for simulating the Doppler effect.

Each pixel within the PIR essentially embodies the path's propagation information in space, consistent with the output of the path tracing part in a ray tracing simulator. Such representation can easily compute the final signal through a signal generator, yet it remains independent of radar device parameters. Moreover, it contains structured spatial information, allowing image-based diffusion models to transfer their knowledge effectively.

5.3.2 RF Path Low-Rank Adaptation (LoRA). So, how can existing image diffusion models generate PIR? Our solution adapts the Low-Rank Adaptation (LoRA) [22], a method originally developed for fine-tuning large language models. By freezing the pre-trained model weights and injecting trainable layers (*i.e.*, rank-decomposition matrices) into each transformer block, LoRA avoids recomputing the gradients for most model weights, thus significantly reducing the trainable parameters and GPU memory requirements. When applied to diffusion models [19], it acts on the cross-attention layers that connect the image representations with the textual prompts.

In RFGen, we train a LoRA for diffusion that fine-tunes to the PIR format, namely **RF Path LoRA**. By utilizing the RF simulator's path tracer on several open-source 3D models, we render a set of images as few-shot inputs to train the model so it learns the characteristics of PIR. This aids in the efficient learning and adaptation of the existing image-based diffusion models for generating PIR, thereby enabling us to generate structured ambient environment signals for enhancing generalization.

5.4 Environment Diffusion

The **Environment Diffusion** operates as a Text-to-Image diffusion model. Pre-trained on extensive indoor and outdoor scene photographic datasets, this diffusion model is further fine-tuned to the PIR signal representation using RF LoRA that is embedded with transformer layers. With the RF LoRA, the environment diffusion learns how to output PIR format as a "painting style", as illustrated in Fig. 5.

A key advantage of environment diffusion is its capability to generate diverse environmental information based on the given prompt. Despite being generated from textual prompts, the resulting PIR maintains reliable spatial information. Moreover, this diffusion model can accurately specify the Field of View (FOV) and orientation details, allowing for a harmonious match with real mmWave radar configurations.

The spatial information embedded in the output PIR is later amalgamated with the paths computed by the Path Tracer. The

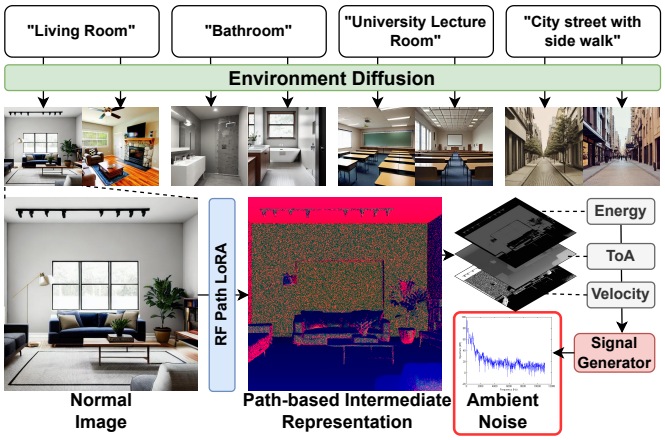

Figure 5: The environment diffusion with RF Path LoRA can generate various of environments from given prompts to calculate the ambient noise.

combined output is then fed to the **Path Diffusion** model for further enhancement. By delivering accurate spatial information alongside adaptability to varied environments, the Environment Diffusion significantly contributes to bridging the simulation-to-reality gap in RF signal simulation.

5.5 Path Diffusion

Path Diffusion, the final diffusion model in RFGen aims to rectify any flaws in the mixed product of the simulator-calculated paths and the output PIR from Environment Diffusion. Similar to Environment Diffusion, it is also an image generation model fine-tuned through RF LoRA. However, the path diffusion model uses not only the Prompt as input but also the mixed PIR. It further refines the PIR output. In other words, it acts like an image-to-image diffusion model that can rectify an imperfect sketch into a photorealistic image. To enable Text-to-Image diffusion models to support the mixed PIR as input, we employ the ControlNet [66].

ControlNet provides a higher degree of control over the generation process in diffusion models such as Stable Diffusion. It permits users to condition the generation process with a variety of spatial contexts, such as depth/segmentation maps and key points. ControlNet leverages a dual structure with both "trainable" and "locked" copies of pre-trained parameters. This structure preserves the learned semantics while also allowing for new, task-specific diffusion modeling.

In essence, path diffusion can further optimize the combination of the traced path and PIR from Environment Diffusion for enhanced signal realism. This represents a further step in mitigating the Sim-to-real gap in the RF simulator, providing a more robust and accurate representation of the RF environments.

6 IMPLEMENTATION AND INTEGRATION

Radar hardware configurations. RFGen's diffusion models are device-agnostic, but its RF simulator should be configured according to the specific sensing device, in particular the antenna layout and carrier frequencies. We have configured RFGen to simulate

diverse mmWave radar platforms including *(i)* TI AWR1832: 77 GHz radar with a 3T×4Rx antenna array. *(ii)* TI MMWCAS-RF-EVM (i.e., the Cascade radar): 77 GHz with a 12Tx×16Rx antenna array. *(iii)* Infineon Position2Go: 24 GHz frequency with a simpler 1Tx×2Rx array. The radar output power, antenna elements' relative positions and gain patterns are available in the respective hardware specs, and accordingly used to configure RFGen's RF simulator. Specifically, each antenna element acts as one signal source emitting a set of rays (Sec. 4), and each ray's strength is scaled by the antenna gain along the corresponding direction. Different antennas' rays interact with the objects/environments and with each other. The resulting entangled rays are reflected back to each antenna element, forming a multi-channel baseband sample just like the actual radar hardware.

Sensing models. RFGen can simulate and output the ADC sampled data from each receiving antenna, producing a data format identical to that of actual radar hardware. This allows it to directly augment existing RF sensing models, which commonly adopt the same input format. Since mmWave signals are non-human sensory signals, one cannot visually assess the quality of the generated results in the same way as other AIGC systems. To address this, we conduct end-to-end system experiments as case studies. If the generated signal enhances the system's performance with real data, then the synthesized data can be deemed effective and usable. For our case studies, we use RFGen to enhance two state-of-the-art mmWave sensing systems. (i) ***mmMesh*** [61] performs mmWave-based human pose reconstruction. It takes sparse point cloud data from MIMO radar as input, and outputs the motion and shape parameters of a Standard Multi-Person Linear (SMPL) model. (ii) ***DI-Gestures*** [29] is a mmWave-based gesture recognition system. It accepts Dynamic Range Angle Image (DRAI) as input and outputs the class of recognized gesture. We have reproduced both systems and processed the simulated signals in the same manner as the original papers.

Pre-trained diffusion models. Recall that diffusion models in RFGen are used simultaneously for generating target object geometry and motion and for enhancing simulator outcomes. To accommodate these varied tasks, we implement the diffusion framework in RFGen using different pre-trained models and network architectures.

For Object Diffusion, our current implementation primarily focuses on human motion, corresponding to the two case studies. Hence, we utilize the human motion diffusion model in [52] as the pre-trained model. However, it is worth noting that Object Diffusion can be replaced with diffusion models for other entities, such as animals or vehicles. RFGen provides a flexible framework to incorporate various object diffusion models, demonstrating its adaptability and potential for wide-ranging applications.

For the Environment Diffusion and Path Diffusion, we adopt the ProtoGenX53Photorealism [51] model since it is one of the most popular community-maintained models, designed for generating photorealistic photos, and is capable of Granular Adaptive Learning. This pre-trained model is capable of generating authentic indoor and outdoor scenes. By employing these distinct pre-trained models, RFGen can adaptively cater to the complex and diverse requirements of RF signal simulation, offering increased realism and precision in the generated data. The pre-trained model is fine-tuned by RF LoRA via DreamBooth [46]. We utilize the path information traced on diverse public 3D models [35] as our training data and generate a total of 800 frames in the PIR image format. The Object Diffusion model comprises 23M parameters, while the Environment Diffusion and Path Diffusion models comprise 1.066B parameters. The training step is set to 100 and the learning rate is set to 0.0001. Training the LoRA takes about one hour on an RTX3070 graphics card.

7 EVALUATION SETUP

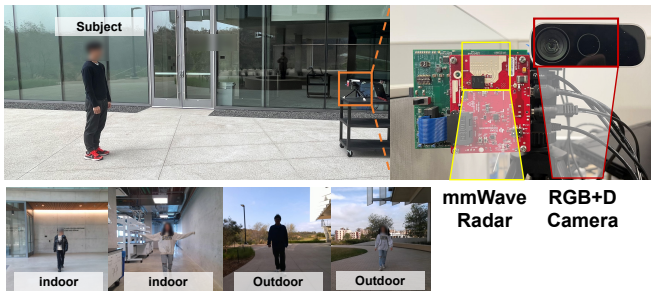

Figure 6: Data collection hardware and environments.

Subjects: A total of 10 participants are recruited, with 5 female and 5 male, spanning 22 to 32 years in age, and 160cm to 185cm in height. All procedures are approved by IRB.

Environment: As shown in Fig. 6, we evaluate RFGen in 10 different environments, including both indoor multipath-rich lab environment and outdoor open space, at different times of day (9am-9pm) and different weather conditions (*e.g.*, rainy, foggy, sunny).

Evaluation metrics: We elaborate on the different evaluation metrics for assessing the performance of RFGen. It is important to note that these matrices are the same as those used in baselines proven effective in prior works.[8, 36, 61]

- *Average Vertex Error* ***(V)*** evaluates the average difference between the vertices of the actual 3D meshes and those reconstructed from the mmWave signals. A lower V value signifies a more accurate reconstruction.
- *Average Joint Localization Error* ***(S)*** measures the error in localizing joints in the reconstructed human pose compared to the actual one. Lower is better.
- *Average Joint Rotation Error* ***(Q)*** quantifies the error in estimating the rotation of joints in the reconstructed pose.
- *Mesh Localization Error* ***(T)*** assesses the difference between the actual and estimated location of the entire mesh in 3D space. A lower T value indicates a more accurate positioning of the reconstructed mesh in the 3D scene.
- *Gender Prediction Accuracy* ***(G)*** gauges the model's accuracy in predicting the gender of subjects based on the reconstructed 3D mesh. A higher G value implies the model's capability to more accurately extract subtle body shape features.

For microbenchmark evaluation of the RF simulator's accuracy, we compare the simulated signal with a real-world radar signal following two metrics:

- *Structural Similarity Index* ***(SSIM)*** measures the similarity between the simulated and real-world signals. A higher SSIM indicates a closer match, suggesting a more accurate simulation.

- *Peak Signal to Noise Ratio* ***(PSNR)***: quantifies the ratio between the maximum possible power of a signal and the power of corrupting noise. A higher PSNR indicates a lower level of noise in the simulated signal, and hence a more precise simulation.

We also employ ***Hausdorff Distance*** [36] for comparing the simulated and actual radar point clouds, as it is popular in comparing point cloud distribution [25]. This metric quantifies the greatest of all the distances from a point in one set to the nearest point in the other set. A lower Hausdorff distance denotes a higher degree of similarity between the two point sets, indicating a more precise alignment or correspondence. The Hausdorff distance is particularly sensitive to the worst-case scenario, *i.e.*, even a single pair of distant points can result in a large Hausdorff distance. Thus, it is a robust measure for maximum discrepancies between two point clouds.

8 SYSTEM EVALUATION

In this section, we evaluate the accuracy of RFGen's RF simulator.

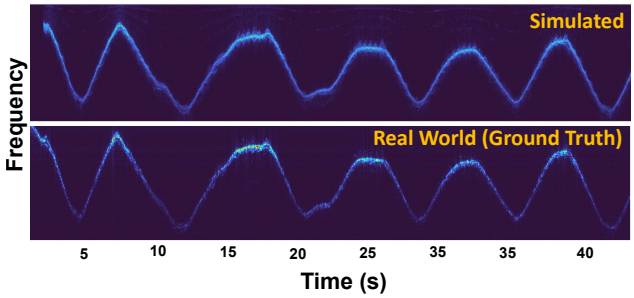

Figure 7: RFGen Simulated signal in Spectrogram format.

8.1 Evaluation of the RF Simulator

RFGen can generate the raw radar signals encapsulates information about the Doppler effect and phase, which can be used for different signal processing, akin to real-world data. Therefore, we evaluate the RF Simulator on two levels: *signal-level* and *point-cloud level*. The former focuses on fundamental information, such as range, phrase, and speed. The latter emphasizes the accuracy of 3D scenes, including attributes like the shapes and sizes of objects. In the experiments, we use actual radar hardware (i.e., TI AWR1843 Radar with 3Tx and 4Rx) to capture the ground-truth raw signals and point cloud, while simultaneously using an RGBD camera to obtain the subjects' 3D mesh which is used as input to the RF simulator. We then compare the simulator's outputs with the ground truth. We collected 8 samples per subject per environment according to the setup described in Section 7. The subject is at 1.5 to 5 meters in front of the radar, facing random directions. A total of 800 samples are collected for this evaluation.

Signal-Level Evaluation. The input for simulation consists of human meshes reconstructed from the RGBD camera, as shown in Figure 6, with a participant walking back and forth. We convert the simulator's output raw time-domain signal into Spectrogram and Dynamic Range Angle Image (DRAI). An example comparison is shown in Fig. 7, which displays a Time-Range Spectrogram processed over a 10-second duration. The pixel intensity represents the received signal strength. The average SSIM and PSNR are calculated for Spectrogram (***0.971, and 32.45*** respectively) and DRAI (***0.924, and 31.75*** respectively). Given the typical thresholds of above 0.85 for SSIM and above 30 for PSNR, these results suggest a high degree of structural similarity between the compared images [28, 55]. Thus, these results indicate that RFGen's simulator performs satisfactorily at the signal level, accurately modeling the time-domain raw signal.

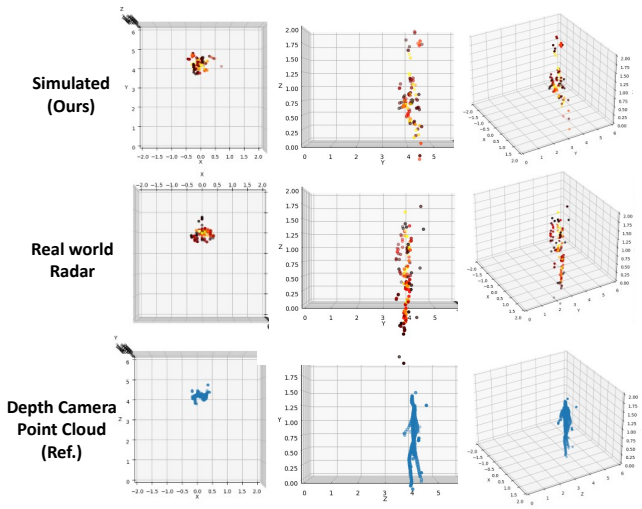

Figure 8: RFGen simulated signal in point cloud formats.

Point cloud-Level Evaluation. We calculate the point clouds from the measured and simulated signals following the same point cloud generation algorithm in mmMesh [61]. The sample results are shown in Fig. 8.

The average Hausdorff distance across all collected samples is 0.37 meters. Considering the human scale of an average height of 1.75m and the complexity of the body, a Hausdorff distance of 0.37m may be considered satisfactory [15, 67]. This value represents only a minor fraction of an average human's height. The intricacy of the human form highlights the significance of this level of accuracy, especially when we consider that the human mesh isn't modeled with ultra-precision. Additionally, real-world sensing uncertainties might also influence this distance. The low average Hausdorff distance suggests that the RF simulator can generate point clouds that closely mirror those obtained from real radar signals. This demonstrates the simulator's capability to replicate the essential features and characteristics of the MIMO signal at the point cloud level.

8.2 Evaluation on different Radar Platforms

RFGen supports and can simulate various radar platforms. We use the TI AWR1843 and TI Cascade radar as evaluation platforms. These two radars have different antenna counts (3x4 and 12x16 respectively) - leading to differing spatial resolutions. Consequently, we compare RFGen's simulation results on these distinct platforms.

We co-located the AWR1843 and Cascade Radar side by side, collecting data simultaneously as ground truth. To avoid interference between the two radars. The results are represented in Fig. 9. Since the Cascade Radar has a higher number of virtual radars,

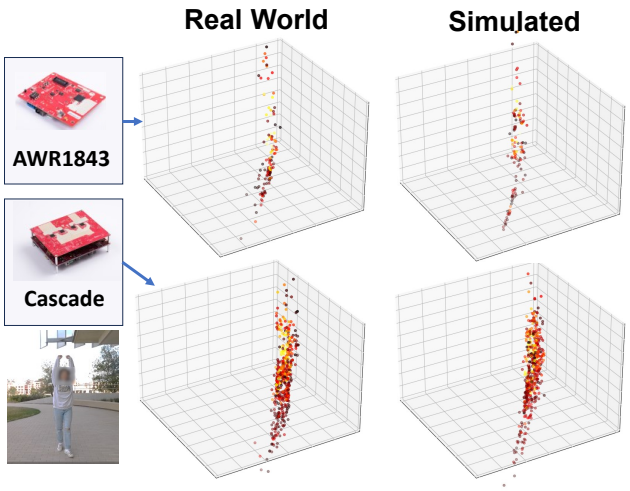

Figure 9: . RFGen simulated signal for two different radar platforms.

it provides a higher resolution for point cloud data. RFGen can accurately simulate this phenomenon while maintaining a lower average error in terms of the Hausdorff Distance (13.7cm). These results demonstrate that RFGen can be compatible with a variety of radar models and configurations, thus proving its scalability and adaptability.

8.3 Noise Model Evaluation

We assess the effectiveness of RFGen in generating structured noise to represent the effects of environmental multipath. The noise model incorporates the environment diffusion and path diffusion. We use state-of-the-art methods such as AWGN (Additive White Gaussian Noise) noise augmentation [54] and NST (Neural Style Transfer) [1, 53] as benchmarks. For NST, we trained a CycleGAN[70] to map the simulated signals' DRAI under the same scenario onto real DRAI. The training dataset comprises both indoor and outdoor scenes. In the case of AWGN, we directly manipulate the simulated DRAI using the OpenCV library.

To establish a baseline mmWave sensing model for end-to-end performance evaluation, we design a DRAI-based posture classification system that includes four postures: standing, raising the left hand, raising the right hand, and raising both hands above the head. The test subjects are positioned normally in front of the sensor at a distance of one meter. We perform the same actions under various environments, including scenes with heavy multipath interference. It is worth noting that we do not implement any noise filtering and thresholding component in the baseline, as these are usually application dependent.

The recognition accuracy in seen environments (*i.e.*, in the NST training data) for *baseline baseline+AGWN*, *baseline+NST*, and *baseline+RFGen* are ***93.79%, 88.30%, 99.13%, and 97.83%***, respectively. For *unseen* environments (*i.e.*, NOT in the NST training data), the recognition accuracy is ***91.47%, 84.71%, 92.84%, and 95.82%***, respectively. The results indicate that the AGWN data augmentation may adversely reduce recognition accuracy since its noise may not accurately reflect the actual scenario. NST performs well in the trained scenes but exhibits a decline in performance when faced with scenes that it has not previously encountered. The performance degradation is noticeable but not dramatic, primarily because the first-order line-of-sight reflections on the subject still dominate. On the other hand, while the RFGen noise model may slightly lag behind NST in specifically trained scenarios, the flexibility and diversity of RFGen ensure a more stable accuracy in other tested scenarios, especially those unseen ones.

8.4 Impact of Quality of Prompts

The content, length, and quality of the prompt impact the quality of the results generated by the diffusion model. Although prompt engineering remains an active area of research, we conduct a preliminary evaluation of the impacts specific to the RF sensing prompts in RFGen.

Table 1: Descriptions of the four RF sensing prompts used in .

Prompt	Content
P1	A person is jumping.
P2	A person is doing jumping jacks in place.
P3	A person is doing jumping jacks in an outdoor playground.
P4	**Positive:** A person is doing jumping jacks in place, ultra quality, best quality, realistic, natural, facing forward, realistic, masterpiece, high detail. **Negative:** idle, immobile, falling, unrealistic, physics-defying, trembling.

As a microbenchmark, we chose the action of *jumping* due to the significant variations in people's understanding and execution of this action. We asked test users to perform jumping jacks in outdoor scenes to gather the test data. The prompts selected for this experiment can be found in Table 1. Prompts P1 to P3 become increasingly precise in guiding the desired action. It's important to note that prompt P4 is complex, containing both "positive" prompts that encourage constructive or favorable outcomes and "negative" prompts that elicit critical or unfavorable outcomes. Upon observing the motion animations and PIRs generated by diffusion models based on prompts P1-P4, we found that P1 results in the greatest variation in generated motions in jumps, including running jumps, standing long jumps, and jump roping. The actions generated by P2 and P3 are essentially consistent, but P3 incorporates noise data from the playground environment. P4's action data is the best among the selected prompts, showing the body facing the sensor without any strange or irrational movements, this is primarily due to our detailed descriptions of Positive Prompts (i.e., to make a concept more pronounced in the output) and Negative Prompts (i.e., to de-emphasize certain elements in the output or to remove unwanted items).

We evaluate the same baseline model as mentioned above using data from prompts P1-P4, and the resulting recognition accuracy is ***94.3%, 94.9%, 95.02%, and 98.2%***, respectively. These results show that *carefully crafted and accurate prompts generally yield higher data quality*. Additionally, the inclusion of some universal positive words also improve the quality of the generated results. However, excessively specific prompts may reduce or overlook certain variant states of the target action, thus limiting generalization. Nonetheless, we believe that the content of the prompt should be based on the

target application, using comprehensive language that strikes a balance between specificity and generalization in describing the sensing application.

8.5 Computation Overhead

The Computation Overhead of RFGen can be broken into two parts:
Physical Simulation Models: The ray tracing modules is highly optimized and parralled on GPU as mentioned in Section 4. The main overhead of ray tracing is in launching a large number of rays and calculating their intersections with the geometric mesh. However, with the use of BVH, the complexity of this process can be optimized to $O(logN)$, where N is the number of triangles. Adaptive sampling can significantly reduce the number of rays required. This means that ray tracing a model with the complexity of a human body mesh takes approximately 10-50 milliseconds per frame and roughtly 3 to 15 seconds per action.
Generative Diffusion Models: Diffusion Models requires relatively more computation overhead due to its diffusion sampling process. Key sources include sampling multiple times per image, attention mechanisms, large model size of over 1 billion parameters, and high resolution generation up to 1024x1024. Computation scales linearly with the number of sampling steps and image resolution. A higher sampling count produces better quality but requires more overhead. On a typical home desktop with a mid-range GPU, generating a 512x512 image with 50 sampling steps takes around 90 seconds. Reducing sampling steps or resolution can significantly decrease generation time. Overall, Stable Diffusion offers high quality diffusion-based image generation, but at the cost of considerable computation overhead that depends largely on model structure parameters like size and sampling count.

For RFGen, the **Object Diffusion** executes once for an entire action and typically takes around 3 seconds. The **Environment Diffusion**, given the same environment and observation angle, needs to run only once and takes between 3 to 5 seconds. In contrast, the **Path Diffusion** might execute once for each frame, essentially "repainting" it. However, there are several avenues to further enhance the diffusion model's utilization, such as kernel predicting [2] and interpolating, and also to improve the diffusion's efficiency, like structural pruning[14].

In practical terms, for RFGen to generate a fresh training dataset comprising 1000 unique action data points for TI AWR1843 radar, it can take anywhere from 5 hours up to a day on a RTX3070 graphics card, with individual samples ranging from 20 seconds to 2 minutes each. Although generating a dataset via RFGen still requires a substantial amount of time and computational resources, it's worth noting that RFGen's data synthesis method is still more cost-effective and time-saving than manually collecting video/radar data, and it provides better diversity and scalability.

9 CASE STUDY I: HUMAN POSTURE ESTIMATION

In this section, we evaluate RFGen's performance in enhancing mmWave-based human posture estimation system [61]. This evaluation is conducted on the TI-AWR1843 radar.

9.1 Posture Design

We define a set of *basic poses* in line with the state-of-the-art mmMesh system [61]. This set includes: (B1) Both hands raised horizontally (T Pose), (B2) Both hands raised high (Y Pose), (B3) Only the left hand raised horizontally, and (B4) Only the right hand raised horizontally. Also included are leg-raising movements: (B5) Left leg raised and (B6) Right leg raised. The basic poses are partially illustrated in Fig.10(a) to (e).

Furthermore, we define a set of *complex poses*. Instead of specifying precise poses for these actions, we provide the test subjects with instructions and allow them to perform the actions based on their own interpretation. The instructions are: (C1) "Walk back and forth", (C2) "Jump", and (C3) "Pick up an item from the ground." It is worth noting that different test subjects interpreted the same instruction in different ways due to individual habits. For example, in response to the "Jump" instruction, some subjects performed a jumping jack (raising both hands above their heads at the highest point), while others crossed their arms in front of their chest. This variation is consistent with real-world scenarios and is part of the generalization challenge.

Table 2: Results for Unseen Postures

	V(cm)	S(cm)	Q(°)	T(cm)
Baseline	54.78	48.52	31.95	23.94
Baseline+RFGen	11.21	7.59	8.14	3.76

9.2 Evaluation on Unseen Posture

We evaluate the baseline system (i.e., mmMesh)'s capacity to recognize *unseen* postures and assess RFGen's ability to enhance it. The evaluation is conducted under constant conditions, with the environment and human subjects remaining unchanged. The system is trained only using the aforementioned basic actions (Sec. 9.1), whereas the testing involves complex actions.

We assess two models: the baseline model, and the enhanced version of RFGen, referred to as *"Baseline+RFGen "*. In the latter, the original baseline model is trained with real radar data for simple postures with additional synthetic data generated via RFGen. Importantly, the prompts used by RFGen for generating training data correspond to the instructions for the complex actions. This approach led RFGen to generate 1000 distinct actions based on these instructions.

The results are presented in Table 2. The baseline model fails to estimate the postures of unseen activities, resulting in significant errors across the V, S, Q, and T metrics. However, when the same model is enhanced by RFGen without any additional data collection, the *posture error is reduced by approximately 60% to 90%*. This impressive improvement is attributed to RFGen's ability to generate extensive and diverse training sets using generative diffusion models. By exposing the classifier to a plethora of plausible scenarios, the generative model introduces the system to the true underlying data distribution, facilitating the learning of more robust feature representations. As a result, the classifier can generalize to complex unseen postures, a feat unachievable with the limited real-world examples in the baseline dataset. *Reinforcing the baseline with RFGen's synthetic data leads to a notable boost in recognizing these previously unseen poses.*

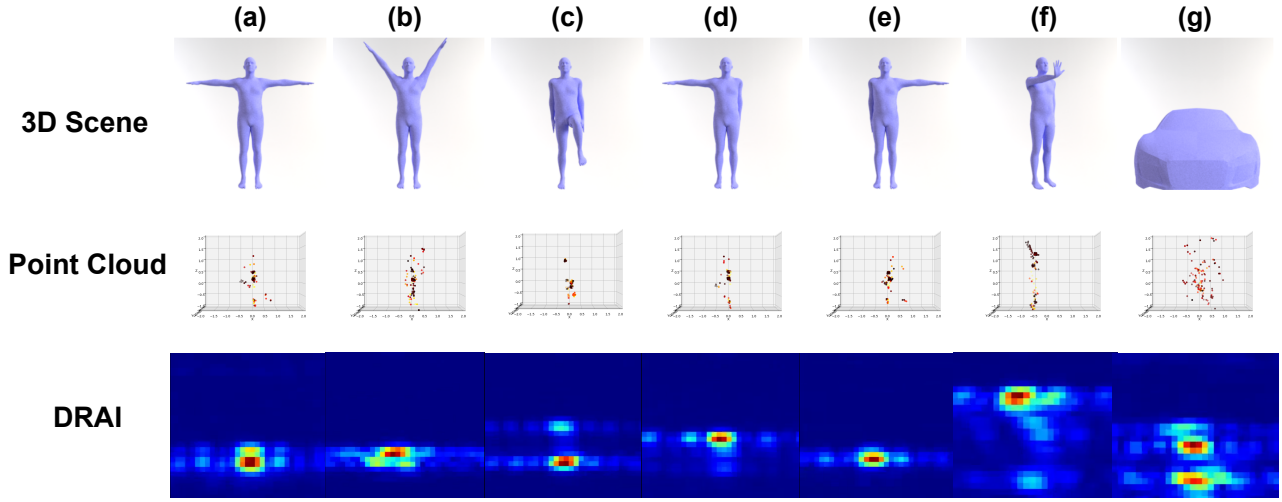

Figure 10: The simulation results for different scene input. (a) to (e) are the different human postures. (f) is an example of human hand gestures at different orientations. (g) is an example of a car, showing RFGen's simulator is capable of various types of 3D scene input.

9.3 Evaluation on Unseen Environments

Table 3: Results for Unseen Environments

	V(cm)	S(cm)	Q(°)	T(cm)
Baseline	8.95	15.79	9.89	13.93
Baseline+RFGen	6.96	10.64	6.12	6.74

Similarly, we examine the ability of the baseline system to recognize basic poses within complex environments. For the baseline model, training is conducted in half of the environments (three indoor and three outdoor) using basic actions, and the remaining environments (three indoor and three outdoor) are used for testing. Importantly, the test environments encompassed both indoor and outdoor settings. The results, depicted in Table 3, reveal that the baseline model maintains a degree of robustness in unseen scenarios, particularly in cases where environmental objects are located far from the human subject.

However, the baseline model performance degrades significantly when objects like tables, chairs, and sofas are in close proximity to the human subject, causing significant multipath noise. By contrast, the baseline model supplemented with RFGen, namely "baseline+RFGen", continued to demonstrate high accuracy even under such conditions.

9.4 Evaluation on Unseen Users

We evaluate the baseline model's performance on unseen users with and without RFGen enhancement. In this experiment, for the baseline model, we use all the data from two male and three female participants for training, and the remaining individuals as our test data set. The scenario and actions performed remained consistent throughout. For the "baseline+RFGen " model, we generate data for 20 different body shapes and genders performing identical actions as a supplement. The results are shown in Table 4. We observe that the "baseline+RFGen" model still demonstrates remarkable improvements across all metrics. Notably, *errors related to body shape (i.e., the S and T metrics) are decreased by more than 50%.* Both models maintained a very high accuracy rate in predicting gender. The results indicate that using RFGen can improve generalization capabilities across users to a certain extent.

Table 4: Results for Unseen User

	V(cm)	S(cm)	Q(°)	T(cm)	G (%)
Baseline	7.27	9.57	4.20	14.42	87.24
Baseline+RFGen	6.41	4.17	5.60	7.86	92.51

10 CASE STUDY II: HAND GESTURE RECOGNITION

We proceed to evaluate RFGen's performance when enhancing the state-of-the-art mmWave-based hand gesture recognition system [29]. This evaluation is conducted on the TI-AWR1843 radar and Infineon position2go radars.

10.1 Hand Gesture Design

Following the approach described in the DI-Gesture paper [29], the generation set primary covers single-hand actions, with 6 types of movements in total: (G1) swipe left, (G2) swipe right, (G3) push forward, (G4) pull backward, (G5) draw a clockwise circle, and (G6) draw a counterclockwise circle. Note that DI-Gesture is performed assuming the individual is facing the mmWave radar directly. As a result, besides the palm, other parts of the human body will also be exposed to the mmWave radiation. Therefore, we simulate the entire human body whenever RFGen is used to synthesize new training data.

10.2 Evaluation on Unseen Orientations

mmWave-based gesture recognition systems often struggle with generalization when hand gestures are presented at varying orientations, which cripples their usability in practice. In this section, we evaluate RFGen's ability to mitigate these issues.

We select 0, 30, 60, and 90 degrees as orientation measures. The prompts used for data synthesis are P5 shown in Table 5. For the

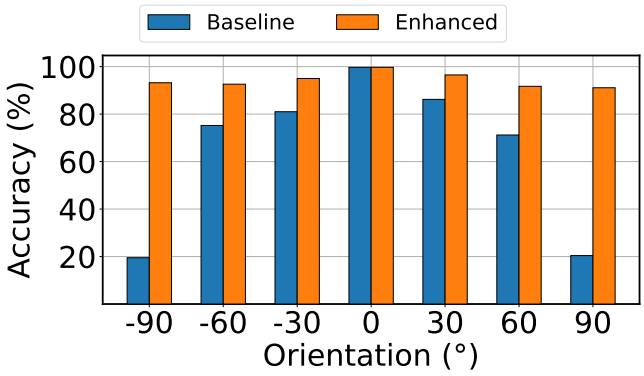

Figure 11: Gestures Recognition Baseline performance at different orientations with and without RFGen enhancement.

baseline system, the training data is only collected when the subject is at 0° relative to the radar while testing data involves diverse orientation angles. The results are depicted in Fig. 11. The *baseline system struggles in identifying the target hand gestures, with accuracy dropping to 61% at 60 degrees and around 20% at 90 degrees!* However, when supplemented with the 200 randomly generated synthetic data samples from RFGen *(i.e., Baseline+RFGen), a high recognition accuracy of above 90% is maintained across all orientations.* These results highlight RFGen's potential to enhance the robustness of hand gesture recognition systems.

10.3 Evaluation on Unseen Gestures

Table 5: Descriptions of the RF sensing prompts used in the Case Study.

Prompt	Content
P5	A person is doing gesture *{Gesture Description G1-G6 in Section 10.1}* in a repetitive manner. The body is at a random orientation range from 0-90 degrees, the hand is in front of the view. Best Quality, natural, realistic, masterpiece, high detail.
P6	A person is tapping the thumb and index finger together in a repetitive manner. The body is facing forward and stationary, and the hand is in front of the view. Best Quality, natural, realistic, masterpiece, high detail.
P7	A person is swinging the index finger left and right in a repetitive manner. The body is facing forward and stationary, and the hand is in front of the view. Best Quality, natural, realistic, masterpiece, high detail.

Adding a new gesture to an existing gesture recognition system often requires the collection of a large amount of new data through dedicated equipment, which can be labor-intensive and costly. In this section, we train the gesture recognition system using solely RFGen-generated data and evaluate it on real-world hand gesture test sets collected from the radar device.

We employ "Click" and "Finger swing" as new gestures. The prompts used for data synthesis are shown in Table 5 P6 and P7, respectively. A total of 200 random samples are generated and used for training. The experiment results show the trained model is able to *recognize these unseen gestures at 85.3% and 93.9% accuracy, respectively.* This indicates the baseline model trained exclusively on the synthetic RFGen data is capable of recognizing real-world hand gestures. In summary, RFGen can serve as a valuable tool for training and customizing mmWave-based sensing systems in a cost-effective and efficient manner.

11 DISCUSSION

Prompt engineering. Prompt engineering has emerged as a pivotal component in the deployment and utilization of Diffusion models [45] and Large Language Models (LLMs)[40]. The design and selection of input prompts are fundamental in steering the responses of such generative models and consequently influence the quality of outputs.

In this work, we have carried out a preliminary evaluation of the influence of prompt quality on output signals in Section 8.4. However, more in-depth research is certainly warranted. In particular, the investigation of cross-modal prompts may represent a promising direction for future work. As we move forward, we aim to address these nuances in order to advance our understanding and improve the effectiveness of diffusion models in RF sensing systems.

Dynamic sensors. There are certain mmWave sensing applications that require data from sensors during motion [59]. RFGen's simulator natively supports dynamic sensor motion and is capable of simulating the Doppler effects. However, due to limitations inherent to the diffusion models, both the Environment Diffusion and Path Diffusion modules currently struggle with dynamic scene locations and may face some jitting and inconsistency between multiple frames.

Nevertheless, with the rapid advancement in diffusion technology, there are already related works underway to address these issues [5]. In this paper, we merely propose a framework for transferring existing diffusion models into the RF domain, thereby paving the way for further improvements and application of this technology.

12 CONCLUSION

We have designed and validated RFGen, a mmWave sensing data synthesis framework that integrates a high-precision ray-tracing simulator with a cross-modal generative diffusion model. RFGen can generate diverse visual scenes based on given prompts from the target application and transform the visual data into mmWave sensing records. It garners high accuracy in the cross-modal transformation, using a combination of physics-based ray tracing and generative models to rectify the signal patterns. RFGen already demonstrates a remarkable enhancement in sensing and generalization capabilities across two distinct sensing tasks and data processing methods. Yet it can be easily adapted to augment other deep learning based mmWave sensing applications.

ACKNOWLEDGMENTS

We appreciate the insightful comments and feedback from the anonymous reviewers and shepherd. This work is partially supported by the NSF under Grants CNS-1901048, CNS-1925767, CNS-2128588, and CNS-2312715.

REFERENCES

[1] Karan Ahuja, Yue Jiang, Mayank Goel, and Chris Harrison. 2021. Vid2Doppler: Synthesizing Doppler radar data from videos for training privacy-preserving activity recognition. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–10.

[2] Steve Bako, Thijs Vogels, Brian McWilliams, Mark Meyer, Jan Novák, Alex Harvill, Pradeep Sen, Tony Derose, and Fabrice Rousselle. 2017. Kernel-predicting convolutional networks for denoising Monte Carlo renderings. *ACM Trans. Graph.* 36, 4 (2017), 97–1.

[3] Pietro Barbiero, Giovanni Squillero, and Alberto Tonda. 2020. Modeling generalization in machine learning: A methodological and computational study. *arXiv preprint arXiv:2006.15680* (2020).

[4] Dennis Barrett. 2017. Smarter robotics through mmwave radar sensing | electronic design. https://www.electronicdesign.com/markets/automotive/article/21805921/smarter-robotics-through-mmwave-radar-sensing

[5] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. 2023. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 22563–22575.

[6] Hanqun Cao, Cheng Tan, Zhangyang Gao, Guangyong Chen, Pheng-Ann Heng, and Stan Z Li. 2022. A survey on generative diffusion model. *arXiv preprint arXiv:2209.02646* (2022).

[7] Changan Chen, Carl Schissler, Sanchit Garg, Philip Kobernik, Alexander Clegg, Paul Calamia, Dhruv Batra, Philip Robinson, and Kristen Grauman. 2022. Soundspaces 2.0: A simulation platform for visual-acoustic learning. *Advances in Neural Information Processing Systems* (2022).

[8] Xingyu Chen, Zhengxiong Li, Biacheng Chen, Yi Zhu, Chris Xiaoxuan Lu, Zhengyu Peng, Feng Lin, Wenyao Xu, Kui Ren, and Chunming Qiao. 2022. MetaWave: Attacking mmWave Sensing with Meta-material-enhanced Tags. In *The 30th Network and Distributed System Security (NDSS) Symposium 2023*. The Internet Society.

[9] William H Clark IV, Steven Hauser, William C Headley, and Alan J Michaels. 2021. Training data augmentation for deep learning radio frequency systems. *The Journal of Defense Modeling and Simulation* 18, 3 (2021), 217–237.

[10] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. 2023. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).

[11] Kaikai Deng, Dong Zhao, Qiaoyue Han, Zihan Zhang, Shuyue Wang, Anfu Zhou, and Huadong Ma. 2023. Midas: Generating mmWave Radar Data from Videos for Training Pervasive and Privacy-preserving Human Sensing Tasks. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 7, 1 (2023), 1–26.

[12] Yudi Dong and Yu-Dong Yao. 2020. Secure mmWave-radar-based speaker verification for IoT smart home. *IEEE Internet of Things Journal* 8, 5 (2020), 3500–3511.

[13] Christer Ericson. 2004. *Real-time collision detection*. Crc Press.

[14] Gongfan Fang, Xinyin Ma, and Xinchao Wang. 2023. Structural Pruning for Diffusion Models. *arXiv preprint arXiv:2305.10924* (2023).

[15] Nahuel E Garcia-D'Urso, Jorge Azorin-Lopez, and Andres Fuster-Guillo. 2023. Accurate Estimation of Parametric Models of the Human Body from 3D Point Clouds. In *International Conference on Soft Computing Models in Industrial and Environmental Applications*. Springer, 236–245.

[16] Jian Gong, Xinyu Zhang, Kaixin Lin, Ju Ren, Yaoxue Zhang, and Wenxun Qiu. 2021. RF vital sign sensing under free body movement. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (2021).

[17] David J Griffiths. 2005. Introduction to electrodynamics.

[18] Yan Gu, Yong He, Kayvon Fatahalian, and Guy Blelloch. 2013. Efficient BVH construction via approximate agglomerative clustering. In *Proceedings of the 5th High-Performance Graphics Conference*. 81–88.

[19] Yuchao Gu, Xintao Wang, Jay Zhangjie Wu, Yujun Shi, Yunpeng Chen, Zihan Fan, Wuyou Xiao, Rui Zhao, Shuning Chang, Weijia Wu, Yixiao Ge, Ying Shan, and Mike Zheng Shou. 2023. Mix-of-Show: Decentralized Low-Rank Adaptation for Multi-Concept Customization of Diffusion Models. arXiv:2305.18292

[20] Robert Alfred Herman. 1900. *A treatise on geometrical optics*. University Press.

[21] Rodrigo Hernangómez, Tristan Visentin, Lorenzo Servadei, Hamid Khodabakhshandeh, and Sławomir Stańczak. 2022. Improving Radar Human Activity Classification Using Synthetic Data with Image Transformation. *Sensors* 22, 4 (2022), 1519.

[22] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685

[23] Sajjad Hussain. 2017. *Efficient ray-tracing algorithms for radio wave propagation in urban environments*. Ph. D. Dissertation. Dublin City University.

[24] Cesar Iovescu and Sandeep Rao. 2023. The fundamentals of millimeter wave radar sensor.

[25] Alireza Javaheri, Catarina Brites, Fernando Pereira, and João Ascenso. 2020. A generalized Hausdorff distance based quality metric for point cloud geometry. In *2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 1–6.

[26] Belal Korany, Chitra R. Karanam, Hong Cai, and Yasamin Mostofi. 2019. XModal-ID: Using WiFi for Through-Wall Person Identification from Candidate Video Footage. In *ACM Annual International Conference on Mobile Computing and Networking (MobiCom)*.

[27] Hyeokhyen Kwon, Catherine Tong, Harish Haresamudram, Yan Gao, Gregory D Abowd, Nicholas D Lane, and Thomas Ploetz. 2020. IMUTube: Automatic extraction of virtual on-body accelerometry from video for human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 3 (2020).

[28] Xiangjun Li and Jianfei Cai. 2007. Robust transmission of JPEG2000 encoded images over packet loss channels. In *2007 IEEE International Conference on Multimedia and Expo*. IEEE, 947–950.

[29] Yadong Li, Dongheng Zhang, Jinbo Chen, Jinwei Wan, Dong Zhang, Yang Hu, Qibin Sun, and Yan Chen. 2022. Towards domain-independent and real-time gesture recognition using mmwave signal. *IEEE Transactions on Mobile Computing* (2022).

[30] Zhengxiong Li, Baicheng Chen, Xingyu Chen, Huining Li, Chenhan Xu, Feng Lin, Chris Xiaoxuan Lu, Kui Ren, and Wenyao Xu. 2022. SpiralSpy: Exploring a stealthy and practical covert channel to attack air-gapped computing devices via mmWave sensing. In *Proc. NDSS*. 1–16.

[31] Hao Ling, R-C Chou, and S-W Lee. 1989. Shooting and bouncing rays: Calculating the RCS of an arbitrarily shaped cavity. *IEEE Transactions on Antennas and propagation* 37, 2 (1989), 194–205.

[32] Keith Man and Javaan Chahl. 2022. A Review of Synthetic Image Data and Its Use in Computer Vision. *Journal of Imaging* 8, 11 (2022), 310.

[33] Sivabalan Manivasagam, Shenlong Wang, Kelvin Wong, Wenyuan Zeng, Mikita Sazanovich, Shuhan Tan, Bin Yang, Wei-Chiu Ma, and Raquel Urtasun. 2020. Lidarsim: Realistic lidar simulation by leveraging the real world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[34] Marzuki Marzuki, Dea Kurnia Harysandi, Rini Oktaviani, Lisna Meylani, Mutya Vonnisa, Harmadi Harmadi, Hiroyuki Hashiguchi, Toyoshi Shimomai, L Luini, Sugeng Nugroho, et al. 2020. International Telecommunication Union-Radiocommunication Sector P. 837-6 and P. 837-7 performance to estimate Indonesian rainfall. *TELKOMNIKA (Telecommunication Computing Electronics and Control)* 18, 5 (2020), 2292–2303.

[35] Morgan McGuire. 2017. *Computer Graphics Archive*. https://casual-effects.com/data

[36] Facundo Mémoli and Guillermo Sapiro. 2004. Comparing point clouds. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*. 32–40.

[37] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. arXiv:2003.08934

[38] Norman Müller, Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulo, Peter Kontschieder, and Matthias Nießner. 2023. Diffrf: Rendering-guided 3d radiance field diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4328–4338.

[39] Yong Niu, Yong Li, Depeng Jin, Li Su, and Athanasios V Vasilakos. 2015. A survey of millimeter wave communications (mmWave) for 5G: opportunities and challenges. *Wireless networks* 21 (2015), 2657–2676.

[40] OpenAI. 2023. GPT-4 technical report. *arXiv* (2023), 2303–08774.

[41] Amani Yousef Owda, Neil Salmon, Stuart William Harmer, Sergiy Shylo, Nicholas John Bowring, Nacer Ddine Rezgui, and Mamta Shah. 2017. Millimeter-wave emissivity as a metric for the non-contact diagnosis of human skin conditions. *Bioelectromagnetics* 38, 7 (2017), 559–569.

[42] paperswithcode. 2023. 3D Human Pose Estimation Datasets. https://paperswithcode.com/task/3d-human-pose-estimation

[43] Kun Qian, Zhaoyuan He, and Xinyu Zhang. 2020. 3D point cloud generation with millimeter-wave radar. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* (2020).

[44] Kun Qian, Shilin Zhu, Xinyu Zhang, and Li Erran Li. 2021. Robust multimodal vehicle detection in foggy weather using complementary lidar and radar signals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[45] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10684–10695.

[46] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. 2023. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 22500–22510.

[47] Christian Schöffmann, Barnaba Ubezio, Christoph Böhm, Stephan Mühlbacher-Karrer, and Hubert Zangl. 2021. Virtual radar: Real-time millimeter-wave radar sensor simulation for perception-driven robotics. *IEEE Robotics and Automation Letters* 6, 3 (2021), 4704–4711.

[48] Aristeidis Seretis and Costas D Sarris. 2021. An overview of machine learning techniques for radiowave propagation modeling. *IEEE Transactions on Antennas*

and Propagation 70, 6 (2021), 3970–3985.
[49] Peter Shirley and R Keith Morley. 2008. *Realistic ray tracing*. AK Peters, Ltd.
[50] William Sloane, Camillo Gentile, Mansoor Shafi, Jelena Senic, Philippa A. Martin, and Graeme K. Woodward. 2023. Measurement-Based Analysis of Millimeter-Wave Channel Sparsity. *IEEE Antennas and Wireless Propagation Letters* 22, 4 (2023).
[51] Svengali75. 2023. Svengali75/ProtogenX53Photorealism. https://huggingface.co/Svengali75/ProtogenX53Photorealism Accessed: 2023-06-28.
[52] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. 2022. Human motion diffusion model. *arXiv preprint arXiv:2209.14916* (2022).
[53] Shelly Vishwakarma, Wenda Li, Chong Tang, Karl Woodbridge, Raviraj Adve, and Kevin Chetty. 2021. Neural style transfer enhanced training support for human activity recognition. *arXiv preprint arXiv:2107.12821* (2021).
[54] Peng Wang and Manuel Vindiola. 2019. Data augmentation for blind signal classification. In *MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)*. IEEE, 305–310.
[55] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.
[56] Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. 2022. Broadly applicable and accurate protein design by integrating structure prediction networks and diffusion generative models. *bioRxiv* (2022).
[57] Teng Wei and Xinyu Zhang. 2015. mtrack: High-precision passive tracking using millimeter wave radios. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*.
[58] Teng Wei, Anfu Zhou, and Xinyu Zhang. 2017. Facilitating Robust 60 {GHz} Network Deployment By Sensing Ambient Reflectors. In *USENIX Symposium on Networked Systems Design and Implementation (NSDI)*.
[59] Zhiqing Wei, Fengkai Zhang, Shuo Chang, Yangyang Liu, Huici Wu, and Zhiyong Feng. 2022. Mmwave radar and vision fusion for object detection in autonomous driving: A review. *Sensors* 22, 7 (2022), 2542.
[60] Xuhai Xu, Xin Liu, Han Zhang, Weichen Wang, Subigya Nepal, Yasaman Sefidgar, Woosuk Seo, Kevin S Kuehn, Jeremy F Huckins, Margaret E Morris, et al. 2023. GLOBEM: Cross-Dataset Generalization of Longitudinal Human Behavior Modeling. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 4 (2023).
[61] Hongfei Xue, Yan Ju, Chenglin Miao, Yijiang Wang, Shiyang Wang, Aidong Zhang, and Lu Su. 2021. mmMesh: towards 3D real-time dynamic human mesh construction using millimeter-wave. In *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*. 269–282.
[62] Shuochao Yao, Shaohan Hu, Yiran Zhao, Aston Zhang, and Tarek Abdelzaher. 2017. Deepsense: A unified deep learning framework for time-series mobile sensing data processing. In *Proceedings of the 26th international conference on world wide web*. 351–360.
[63] Zhengqing Yun and Magdy F Iskander. 2015. Ray tracing for radio propagation modeling: Principles and applications. *IEEE access* 3 (2015), 1089–1100.
[64] Chenshuang Zhang, Chaoning Zhang, Sheng Zheng, Mengchun Zhang, Maryam Qamar, Sung-Ho Bae, and In So Kweon. 2023. A survey on audio diffusion models: Text to speech synthesis and enhancement in generative ai. *arXiv preprint arXiv:2303.13336* 2 (2023).
[65] Jie Zhang, Zhanyong Tang, Meng Li, Dingyi Fang, Petteri Nurmi, and Zheng Wang. 2018. CrossSense: Towards Cross-Site and Large-Scale WiFi Sensing. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MobiCom)*.
[66] Lvmin Zhang and Maneesh Agrawala. 2023. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543* (2023).
[67] Longyu Zhang, Bote Han, Haiwei Dong, and Abdulmotaleb El Saddik. 2017. Development of an automatic 3D human head scanning-printing system. *Multimedia Tools and Applications* 76 (2017), 4381–4403.
[68] Xiaotong Zhang, Zhenjiang Li, and Jin Zhang. 2022. Synthesized Millimeter-Waves for Human Motion Sensing. In *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems*. 377–390.
[69] Anfu Zhou, Shaoqing Xu, Song Wang, Jingqi Huang, Shaoyuan Yang, Teng Wei, Xinyu Zhang, and Huadong Ma. 2019. Robot navigation in radio beam space: Leveraging robotic intelligence for seamless mmwave network coverage. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing*.
[70] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.