# IMAGING THE RADAR: RECOVERING 4D RADAR TENSORS FROM IMAGES WITH A LATENT DIFFUSION BRIDGE

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Millimeter-wave radar's all-weather robustness and Doppler velocity sensing make it increasingly important for autonomous perception. However, the scarcity of high-quality 4D radar data limits the development of radar perception systems. Images provide a scalable source for radar data expansion: camera sequences are abundant and inexpensive to acquire, preserve scene structure shared with radar, and support scene-level editing. We therefore introduce image-to-4D radar tensor generation and propose the **Ra**dar-aligned **Vi**sual **P**rior Bridge (RaViP-Bridge) to recover complete radar tensors from images. However, direct translation faces a substantial modality gap. Images record scene appearance on a 2D perspective plane, whereas native 4D radar tensors encode spatial and Doppler responses amid noise and clutter. To narrow this gap, we use visual foundation models to extract geometry, semantic, and motion cues from image sequences and organize them into a radar-aligned visual-prior tensor. Given the high dimensionality and noise of native radar tensors, we use dual VAEs to compress the visual-prior and radar tensors into compact latent spaces. We then formulate latent translation as an endpoint-constrained diffusion bridge from the visual-prior latent to the radar latent. On K-Radar, we find that RaViP-Bridge achieves high-fidelity 4D radar generation and improves downstream detection through data augmentation.

## 1 INTRODUCTION

With Doppler velocity sensing and strong environmental robustness, millimeter-wave radar has become an important sensing modality for autonomous perception (Paek et al., 2022; Kwok et al., 2025). Modern 4D radar further measures range, azimuth, elevation, and Doppler velocity, allowing native radar tensors to preserve both spatial structure and radial motion in polar coordinates (Paek et al., 2022). However, the development of radar perception is constrained by the scarcity of high-quality 4D radar data (Paek et al., 2022; Kwok et al., 2025; Borreda et al., 2025).

Physics-based simulation synthesizes radar measurements from modeled scenes, sensor parameters, and propagation assumptions (Magosi et al., 2022; Holder et al., 2019; Bialer & Haitman, 2024; Xiao et al., 2026). Although this route provides explicit control, high-fidelity simulation requires costly scene assets and radar-specific configuration. Data-driven methods further learn radar generation from object layouts or sensor observations with GANs, diffusion models, or related models (Fidelis et al., 2023; Wang et al., 2021; Jung et al., 2026; Kwok et al., 2025; Zhang et al., 2024; Borreda et al., 2025). However, many of them synthesize range-azimuth maps, BEV attributes, sparse points, or other simplified radar representations, which lose part of the native tensor information and limit their downstream utility (Rangaraj et al., 2025; Kwok et al., 2025; Zhang et al., 2026b; Wang et al., 2025; Borreda et al., 2025).

Researchers have used structured LiDAR geometry to synthesize dense radar data (Wang et al., 2021; Jung et al., 2026), but this approach requires an additional 3D sensor and paired acquisition. Images provide a more flexible source: they are abundant, inexpensive, rich in long-tail driving scenes, and potentially editable at the scene level (Borreda et al., 2025). Recent visual foundation models can extract depth, semantic, and motion cues from ordinary image sequences (Lin et al., 2025; Carion et al., 2025; Xu et al., 2023; Borreda et al., 2025). As shown in Figure 1, these cues preserve object-level structure shared with radar measurements and provide a structured source for cross-modal recovery.



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.

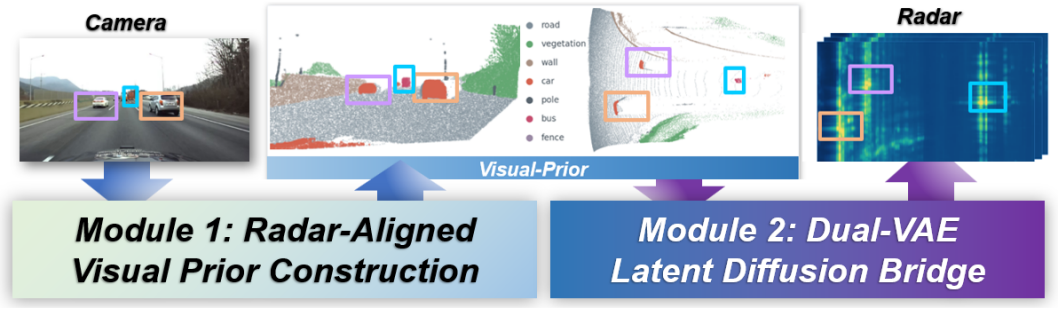



Figure 1: Motivation for image-to-4D radar generation. Although cameras and radar use different sensing mechanisms, they observe the same scene. We use advanced visual foundation models to extract rich scene information from images and precisely guide 4D radar generation.

**We therefore study the generation of complete 4D radar tensors from widely available images as a route to large-scale radar dataset expansion.**

Indeed, image-to-4D radar tensor recovery presents two fundamental challenges. First, the source and target differ in sensing mechanism and representation. A camera records RGB values on a 2D perspective plane, whereas a native radar tensor records echo intensity and Doppler responses over range, azimuth, and elevation. Radar responses further depend on material properties, incidence angle, range, multipath propagation, and other factors that are not directly observed in RGB images. Second, native 4D radar tensors are high-dimensional and noisy, with sparse target returns amid clutter. Their size and low signal-to-noise ratio make direct image-to-radar translation difficult.

To address these challenges, we propose an image-to-4D radar recovery framework called RaViP-Bridge, which develops a latent diffusion bridge from visual priors to radar tensors (Li et al., 2023). We first convert image sequences into a radar-aligned visual-prior tensor with visual foundation models. Second, to handle the high dimensionality, sparsity, and noise of native 4D radar tensors, we use dual VAEs to compress the visual-prior tensor and the real radar tensor into compact latent spaces (Kingma & Welling, 2014). Finally, we formulate image-to-4D radar translation between these latent representations as a diffusion bridge problem with dual boundary conditions (Li et al., 2023).

Our main contributions are as follows:

- We propose an image-to-4D radar tensor generation framework that recovers complete range-azimuth-elevation-Doppler tensors from low-cost visual inputs.
- We design a radar-aligned visual-prior construction module that organizes geometry, permittivity, and motion cues on the radar grid, providing a structured source representation that narrows the image-radar modality gap.
- We develop a dual-VAE latent diffusion bridge module that performs cross-modal translation in compact latent spaces, mitigating the dimensional and noise challenges of native radar tensors and enabling high-fidelity 4D radar recovery.
- Experiments on K-Radar show that the proposed framework achieves high-fidelity 4D radar generation and improves downstream detection through data augmentation.

## 2 Related Work

### 2.1 Radar Simulation and Neural Rendering

Physics-based and hybrid simulators model radar wave interaction through ray tracing, graphics proxies, measured responses, or waveform-parameterized attributes (Magosi et al., 2022; Holder et al., 2019; Bialer & Haitman, 2024; Xiao et al., 2026). These methods provide explicit control, but their fidelity depends on scene construction, radar-specific configuration, and the remaining gap between simulated and real radar measurements.

Neural radar rendering reconstructs scene-specific radar responses with neural fields or Gaussian Splatting and supports novel-view synthesis (Borts et al., 2024; Rafidashti et al., 2025; Zhang et al., 2026a; Kung et al., 2025). Such methods usually require the target scene to be captured before rendering, so they are better suited to reconstructing or re-viewing observed scenes than to scalable creation of training data from camera-only inputs. Our setting instead seeks to recover complete 4D radar tensors from ordinary images, where the input scene can be supplied by low-cost visual data rather than by explicit scene modeling or prior radar capture.

### 2.2 Cross-Modal Radar Generation

Cross-modal radar generation reduces manual scene construction by learning radar outputs from paired sensor data or structured conditions. GAN- and diffusion-based methods have explored radar generation (Fidelis et al., 2023; Rangaraj et al., 2025; Zhang et al., 2024; Kwok et al., 2025). Their outputs often remain partial: range-azimuth maps remove elevation and full Doppler-spectrum structure, while point-based outputs discard the dense radar response field.

For cross-modal radar synthesis, current methods are still largely centered on LiDAR-to-radar translation (Wang et al., 2021; Jung et al., 2026; Shen et al., 2026). LiDAR provides strong geometric structure, but LiDAR-conditioned generation still requires an additional 3D sensor, calibrated sensor configuration, and paired collection. Images offer a broader source because they are abundant, inexpensive, and potentially editable at the scene level. We therefore use images as the low-cost source while targeting complete range-azimuth-elevation-Doppler tensors rather than partial radar representations.

### 2.3 Diffusion Models for Visual-Prior Translation

Standard conditional diffusion starts from Gaussian noise and injects source information into the denoising network (Ho et al., 2020; Rombach et al., 2022). This formulation is flexible, but the condition remains an input to the reverse process rather than an endpoint that defines the translation path.

Diffusion bridge models change this formulation by modeling paired translation as a stochastic process constrained by source and target endpoints. Brownian Bridge Diffusion Models demonstrate this idea for image-to-image translation (Li et al., 2023). The endpoint constraint is well suited to our cross-modal recovery task: the radar-aligned visual-prior latent serves as the source endpoint, the radar latent serves as the target endpoint, and the bridge learns a controlled transition between them.

## 3 Method

### 3.1 Problem Formulation

Let $\mathcal{I}_\tau = \{I_\tau, I_{\tau+\Delta\tau}\}$ denote the front-view image pair used to recover the visual prior for radar frame $\tau$, where $I_\tau$ is aligned with the target radar tensor and $I_{\tau+\Delta\tau}$ is the forward frame used for optical-flow-based motion estimation. In our implementation the frame gap is five camera frames, and $\Delta\tau$ is read from the camera timestamps. Given camera intrinsics, camera-to-radar extrinsics, and $\mathcal{I}_\tau$, we construct a radar-aligned visual-prior tensor

$$S = (S_\rho, S_\varepsilon, S_v) \in \mathbb{R}^{C_s \times R \times A \times E}, \qquad C_s = 3, \tag{1}$$

where $S_\rho$, $S_\varepsilon$, and $S_v$ store projected density, mapped permittivity, and relative radial velocity on the radar range-azimuth-elevation grid. The target is a dense 4D radar tensor

$$R = (R_p, R_d) \in \mathbb{R}^{C_r \times R \times A \times E}, \tag{2}$$

where $R_p$ is the normalized log-power channel and $R_d$ denotes Doppler-spectrum response channels. The task is to learn $G : \mathcal{I}_\tau \mapsto \hat{R}$ through the intermediate tensor $S$, so that the generated tensor $\hat{R}$ matches the complete range-azimuth-elevation-Doppler structure of $R$.

### 3.2 Framework Overview

To recover complete 4D radar tensors from images, we propose RaViP-Bridge, as shown in Figure 2. The framework consists of two modules: radar-aligned visual-prior construction and a dual-VAE latent

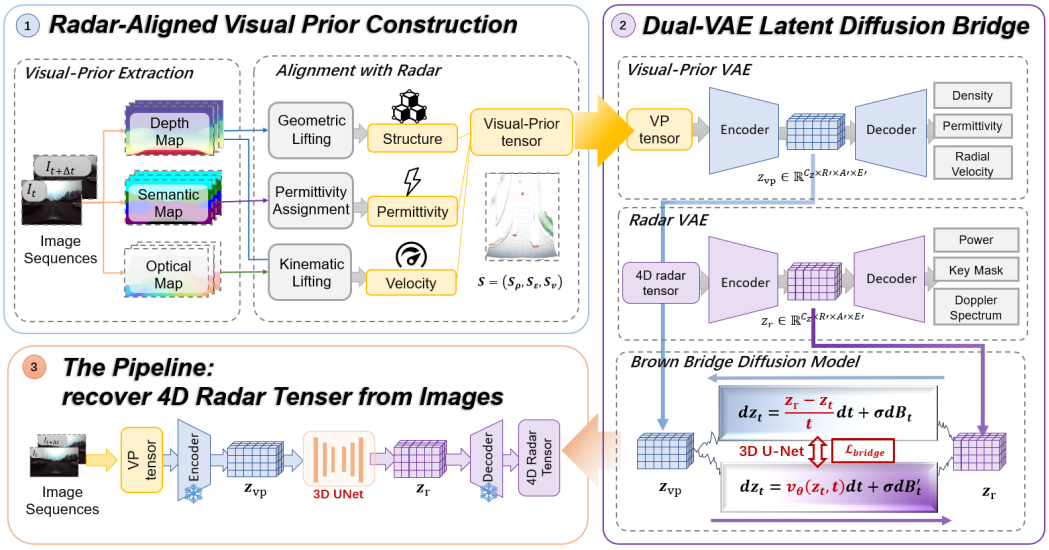



Figure 2: Overview of RaViP-Bridge. The visual-prior construction module extracts visual cues from image sequences and organizes them into a radar-aligned visual-prior tensor. The dual-VAE latent diffusion bridge module encodes visual-prior and radar tensors into latent grids with identical dimensions and models latent translation as a Brownian bridge. During inference, the visual-prior latent initializes bridge sampling, and the radar decoder produces a complete 4D radar tensor.

diffusion bridge. We first use visual foundation models to extract geometry, semantics, and motion from image sequences and organize these cues on the radar grid, reducing the camera-radar modality gap. We then train two VAEs to encode the visual-prior tensor and the high-dimensional, noisy radar tensor into compact latent grids with identical dimensions. Building on BBDM and L2RLDB (Li et al., 2023; Shen et al., 2026), we model visual-prior-to-radar latent translation as a Brownian bridge. We place the visual-prior latent at the source endpoint of the stochastic trajectory to preserve source structure during translation to radar.

### 3.3 Radar-Aligned Visual-Prior Construction

To tackle the modality gap caused by the different imaging geometries and sensing physics of cameras and radar, we extract depth, semantic, and motion cues and organize them into a radar-aligned visual-prior tensor. We first rectify image frames using camera calibration, then use Depth Anything 3 for depth estimation, SAM 3 for semantic masks, and UniMatch for optical flow (Lin et al., 2025; Carion et al., 2025; Xu et al., 2023). Before projection, object-trail depth filtering suppresses stretched foreground depth artifacts, mask-boundary depth correction reduces depth-mask mismatch near semantic edges, and connected-component velocity smoothing suppresses isolated flow outliers in dynamic regions.

**Geometry lifting.** For each valid pixel $u_i \in \mathbb{R}^2$, let $\tilde{u}_i \in \mathbb{R}^3$ be its homogeneous coordinate, $d_i \in \mathbb{R}$ its estimated depth, and $K \in \mathbb{R}^{3\times3}$ the camera intrinsic matrix. The 3D point in the camera frame is

$$p_i^c = d_i K^{-1} \tilde{u}_i, \qquad p_i^c \in \mathbb{R}^3. \tag{3}$$

Using the calibrated rotation $R_{r\leftarrow c} \in \mathbb{R}^{3\times3}$ and translation $t_{r\leftarrow c} \in \mathbb{R}^3$, the point is transformed into the radar frame by

$$p_i^r = R_{r\leftarrow c} p_i^c + t_{r\leftarrow c}. \tag{4}$$

This lifting step assigns each valid image cue a position in the radar coordinate system.

**Semantic permittivity assignment.** Geometry alone does not indicate how strongly a region may respond to radar. We therefore convert semantic categories into representative dielectric-property priors through a simulation-inspired class-material-permittivity lookup table. For a valid projected element, the semantic label first determines a material class, and the lookup table assigns an approximate relative permittivity value $\varepsilon_i$.

**Kinematic lifting and radial projection.** We estimate motion cues from the image pair $\{I_\tau, I_{\tau+\Delta\tau}\}$. Optical flow matches a pixel at time $\tau$ to its position at $\tau + \Delta\tau$, and depth gives the corresponding 3D points $p_i^{r,\tau}$ and $p_i^{r,\tau+\Delta\tau}$ in the radar frame. We project the relative motion onto the radar line of sight:

$$v_i = \frac{(p_i^{r,\tau+\Delta\tau} - p_i^{r,\tau})^\top p_i^{r,\tau}}{\Delta\tau \, \|p_i^{r,\tau}\|_2}. \tag{5}$$

The projected scene elements are therefore

$$P_\tau = \{(x_i, y_i, z_i, \varepsilon_i, v_i)\}_{i=1}^{N_\tau}, \tag{6}$$

where $N_\tau$ is the number of valid elements at time $\tau$.

**Radar-grid voxelization.** Voxelization aggregates $P_\tau$ on the same range-azimuth-elevation grid as the radar tensor. For each polar voxel, $S_\rho$ stores the projected-element count, $S_\varepsilon$ stores the dominant permittivity prior after semantic voting, and $S_v$ stores the mean valid radial velocity. The voxelized priors form $S = (S_\rho, S_\varepsilon, S_v) \in \mathbb{R}^{3\times R\times A\times E}$. We define a visual-prior occupancy mask $M_{\text{vp}} \in \{0,1\}^{R\times A\times E}$ from projected occupied voxels and apply velocity supervision only on occupied voxels, preventing empty bins from dominating the motion channel.

### 3.4 Dual-VAE Latent Diffusion Bridge

Given the large tensor dimensions, we train separate VAEs to obtain compact source and target endpoints for bridge translation (Kingma & Welling, 2014). We first learn the radar latent space from real measurements, then train the visual-prior VAE to produce a latent grid with identical dimensions.

**Radar re-representation and VAE.** Native radar tensors couple a dense spatial power field with sparse Doppler spectra, so direct reconstruction is easily dominated by empty or noisy bins. We therefore represent each sample by a dense RAE power channel $R_p$, a valid-hit mask $M_r \in \{0,1\}^{R\times A\times E}$, and Doppler spectra $R_d$ retained at valid hits. We apply a logarithmic transform to Doppler power, normalize valid responses, and leave non-hit Doppler bins at zero. The radar VAE $(E_r, D_r)$ encodes $(R_p, R_d)$ into $z_r$ and reconstructs $\hat{R}_p$, $\hat{R}_d$, and $\hat{M}_r$ using separate heads.

To preserve both spatial power and sparse velocity responses, we train the radar VAE with

$$\begin{aligned}\mathcal{L}_{\text{rec}}^r = \lambda_p \|R_p - \hat{R}_p\|_2^2 + \lambda_d \|M_r \odot (R_d - \hat{R}_d)\|_2^2 \\ + \lambda_g \|R_d - \hat{R}_d\|_2^2 + \lambda_a \mathcal{D}_{\text{abs}}(R, \hat{R}),\end{aligned} \tag{7}$$

$$\mathcal{L}_r = \mathcal{L}_{\text{rec}}^r + \beta_r \mathcal{L}_{\text{KL}}^r + \lambda_m^r \mathcal{L}_{\text{mask}}^r. \tag{8}$$

Here $\mathcal{L}_{\text{mask}}^r$ is balanced binary cross-entropy on $M_r$, and $\mathcal{D}_{\text{abs}}$ measures absolute error after denormalization. The masked term preserves valid Doppler spectra, while the global term suppresses spurious responses outside radar hits. With these losses, we train $E_r$ to retain spatial and Doppler structure and $D_r$ to reconstruct the radar tensor.

**Visual-prior VAE.** We then fix the radar VAE and train $(E_{\text{vp}}, D_{\text{vp}})$ to reconstruct density, quantized permittivity, radial velocity, and the visual-prior occupancy mask $M_{\text{vp}}$. We use the same latent shape for $z_{\text{vp}} = E_{\text{vp}}(S)$ and $z_r = E_r(R)$, preserving location-wise correspondence on the downsampled radar grid. Let $Q(S_\varepsilon)$ denote the quantized permittivity class. We use channel-specific supervision:

$$\begin{aligned}\mathcal{L}_{\text{rec}}^{\text{vp}} = \lambda_\rho \|S_\rho - \hat{S}_\rho\|_2^2 + \lambda_\varepsilon \text{CE}(\hat{S}_\varepsilon, Q(S_\varepsilon)) \\ + \lambda_v \|M_{\text{vp}} \odot (S_v - \hat{S}_v)\|_2^2,\end{aligned} \tag{9}$$

where CE is cross-entropy over permittivity classes. We supervise velocity only within $M_{\text{vp}}$ so that empty voxels do not dominate the motion channel.

The visual-prior VAE objective is

$$\mathcal{L}_{\text{vp}} = \mathcal{L}_{\text{rec}}^{\text{vp}} + \beta_{\text{vp}} \mathcal{L}_{\text{KL}}^{\text{vp}} + \lambda_m^{\text{vp}} \mathcal{L}_{\text{mask}}^{\text{vp}}. \tag{10}$$

**Latent bridge formulation.** The visual-prior latent $z_{\text{vp}}$ already carries scene layout, permittivity cues, and radial-motion cues on the radar grid. To use these cues as the starting point of generation, we formulate a Brownian bridge between the visual-prior and radar latents.

For bridge training, both VAEs are frozen so that paired latents $(z_{\text{vp}}, z_r)$ define stable source and target endpoints. Let $t \in [0, 1]$, with the radar latent $z_r$ at $t = 0$ and the visual-prior latent $z_{\text{vp}}$ at $t = 1$. We sample bridge states:

$$z_t = (1 - t)z_r + tz_{\text{vp}} + \sigma_t \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I), \tag{11}$$

where $z_t, \epsilon \in \mathbb{R}^{C_z \times R' \times A' \times E'}$, $I$ is the identity covariance, and $\sigma_t = \sigma\sqrt{t(1-t)}$ controls the intermediate variance. This gives a Brownian bridge (Li et al., 2023). The schematic expresses its reverse drift using the remaining bridge time $t$. For the continuous-time dynamics, we use increasing reverse time $u = 1 - t$ and $\tilde{z}_u = z_{1-u}$:

$$d\tilde{z}_u = \frac{z_r - \tilde{z}_u}{1 - u}\, du + \sigma\, dB_u, \qquad u \in [0, 1), \tag{12}$$

where $B_u$ denotes standard Brownian motion and $(z_r - z_t)/t$ is the reverse-time drift. We initialize sampling at the visual-prior latent and evolve it toward the target radar latent.

**Drift prediction with a 3D U-Net.** Since the target radar latent is unavailable at inference, we learn to predict the bridge drift from the current latent state. Following the Markovian projection used in Brownian bridge diffusion (Li et al., 2023), we train a 3D U-Net $v_\theta$ to regress the bridge drift from the noisy latent state and time embedding for $0 < t < 1$:

$$\mathcal{L}_{\text{bridge}} = \mathbb{E}_{t,\epsilon,z_{\text{vp}},z_r}\left[\left\|\frac{z_r - z_t}{t} - v_\theta(z_t, t)\right\|_2^2\right]. \tag{13}$$

The network follows a 3D encoder-decoder architecture with skip connections and timestep-conditioned residual blocks, allowing it to model spatial radar structure at different bridge times. The learned reverse-time drift defines the SDE

$$d\tilde{z}_u = v_\theta(\tilde{z}_u, 1 - u)\, du + \sigma\, dB'_u \tag{14}$$

from $u = 0$ to $u = 1$, with $\tilde{z}_0 = z_{\text{vp}}$, where $B'_u$ denotes Brownian motion in the learned reverse process. For inference, we obtain $z_0 = \hat{z}_r$ with the deterministic sampler below and decode it with the radar VAE.

**Accelerated diffusion bridge inference.** For efficient inference, we select a decreasing subsequence $1 = t_S > \cdots > t_1 = 0$ from the training trajectory and use the non-Markovian bridge sampler of Li et al. (2023). At step $s$, the 3D U-Net predicts the drift $\hat{v}_s = v_\theta(z_{t_s}, t_s)$, from which we estimate the radar endpoint and intermediate noise:

$$\hat{z}_r = z_{t_s} + t_s \hat{v}_s, \qquad \hat{\epsilon}_s = \frac{z_{t_s} - (1 - t_s)\hat{z}_r - t_s z_{\text{vp}}}{\sigma_{t_s}}. \tag{15}$$

We then update the bridge state by

$$z_{t_{s-1}} = (1 - t_{s-1})\hat{z}_r + t_{s-1} z_{\text{vp}} + \sqrt{\sigma^2_{t_{s-1}} - \delta_s^2}\, \hat{\epsilon}_s + \delta_s \epsilon_s, \tag{16}$$

where $\epsilon_s \sim \mathcal{N}(0, I)$ and $\delta_s$ controls sampling stochasticity. At the initial endpoint $t_S = 1$, we set $\hat{\epsilon}_S = 0$. We use 30 deterministic sampling steps ($\delta_s = 0$) in all experiments. Additional training and inference details are provided in the supplementary material.

# 4 Experiments

## 4.1 Evaluation Protocol

**Dataset.** K-Radar is a multimodal autonomous-driving dataset with cameras, LiDAR, dense 4D radar tensors, calibration, and 3D box annotations across diverse weather conditions (Paek et al., 2022). We select 9,008 valid camera-radar pairs and split them 4:2:1 for training, expansion/checkpoint selection, and final testing. A sample is retained when the aligned image, calibration, visual-prior tensor, and radar target can all be constructed.

**Evaluation Scope.** We assess whether generated outputs match complete 4D radar tensors and whether they support detector learning when used alone or as data augmentation.

**Baselines.** Since image-to-4D radar tensor recovery has no established direct baseline, we construct three controlled image-conditioned variants:

- **Pix2PixHD RAE** (3D RAE, no Doppler). We adapt Pix2PixHD (Wang et al., 2018) to polar voxel grids and synthesize RAE radar tensors without Doppler channels.
- **Latent Pix2PixHD** (4D RAE+Doppler). We apply Pix2PixHD (Wang et al., 2018) to adversarial translation between visual-prior and radar latent representations.
- **Latent Diffusion** (4D RAE+Doppler). We denoise radar latents from Gaussian noise under visual-prior conditioning (Rombach et al., 2022).

We also report the published L2RDaS result where available as a LiDAR-conditioned reference (Jung et al., 2026).

**Implementation Details.** Visual-prior and radar tensors share the polar grid $(R, A, E) = (192, 96, 32)$. The visual-prior tensor contains density, permittivity, and radial velocity, while the radar tensor contains normalized log-power and Doppler-spectrum responses with 64 channels. Both VAEs use independent 3D convolutional weights and a downsampling factor of 4, producing latents of shape $4 \times 48 \times 24 \times 8$. The minimum spatial scale of the bridge 3D U-Net is 8. We use 1,000 bridge steps with a linear schedule and $\sigma = 1.0$ during training, and the 30-step deterministic sampler described above during inference. We train the models on one NVIDIA RTX 4090 GPU with 48 GB memory using AdamW, a learning rate of $1 \times 10^{-4}$, a batch size of 8, mixed precision, and cached latent pairs. All latent-space baselines use the same VAEs, latent cache, normalization, and data split. We use a fixed random seed of 42 throughout training and evaluation.

### 4.2 4D Radar Tensor Synthesis

Figure 3 compares generated and real radar responses in range-azimuth and range-Doppler views. The generated tensors recover localized spatial responses around the visible vehicles and reproduce their radial-motion patterns. We use these complementary views to assess spatial localization and Doppler structure within the same generated tensor.

**Metrics.** We evaluate direct synthesis with MAE, PSNR, SSIM, LPIPS, FID, and Doppler PSNR (Zhang et al., 2018; Heusel et al., 2017). MAE is computed after converting the polar radar tensor to a Cartesian ZYX volume. PSNR and SSIM measure normalized tensor fidelity and structural consistency. LPIPS and FID are computed on log-scaled 2D ZYX projections, and Doppler PSNR is computed on valid Doppler regions.

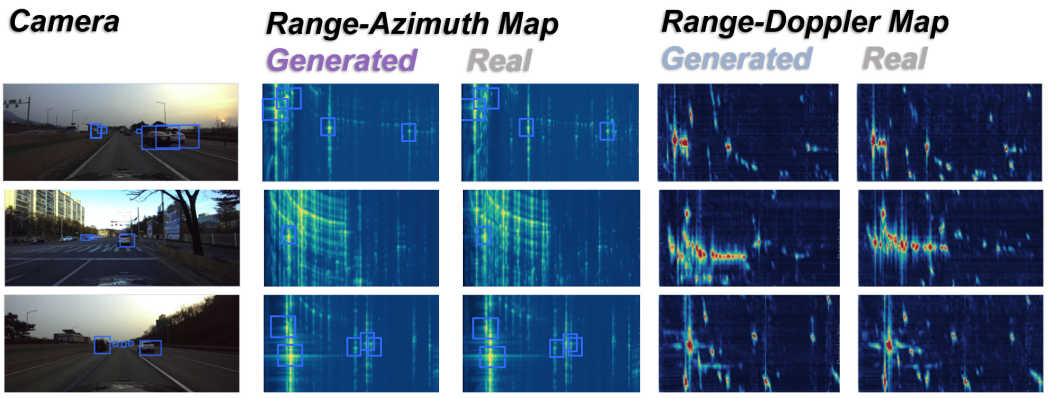
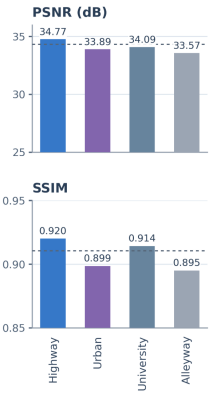



Figure 3: Visual examples of generated 4D radar tensors on K-Radar. Range runs horizontally, with azimuth and Doppler shown vertically in the respective maps. The generated maps retain prominent object echoes and rich background detail without obvious blurring or averaging, closely matching the real measurements.

Table 1: Comparison of image-to-4D radar tensor generation methods on K-Radar. The right panel shows scene-wise PSNR and SSIM for our method.

| Method | MAE↓ | PSNR↑ | SSIM↑ | LPIPS↓ | FID↓ | D-PSNR↑ |
|---|---|---|---|---|---|---|
| Baselines | | | | | | |
| L2RDaS | – | 31.01 | 0.897 | – | – | – |
| Pix2PixHD RAE | 0.097 | 30.23 | 0.878 | 0.038 | 21.49 | – |
| Latent Pix2PixHD | 0.085 | 29.87 | 0.847 | 0.031 | 24.34 | 22.45 |
| Latent Diffusion | 0.083 | 30.03 | 0.871 | 0.036 | 22.72 | 22.53 |
| Ablations | | | | | | |
| Ours w/o $v$ | 0.074 | 33.17 | 0.898 | 0.030 | 18.73 | 22.13 |
| Ours w/o $e$ | 0.084 | 32.60 | 0.874 | 0.035 | 18.98 | 22.94 |
| **Ours** | **0.056** | **34.31** | **0.910** | **0.028** | **12.66** | **24.23** |




**Results.** Table 1 shows that our bridge performs best on all reported metrics. Against Latent Diffusion, PSNR rises from 30.03 to 34.31, FID falls from 22.72 to 12.66, and Doppler PSNR rises from 22.53 to 24.23. Since all latent methods share the visual-prior tensor, radar VAE, split, and normalization, the gains over Latent Pix2PixHD and Latent Diffusion demonstrate the advantage of translating from the visual-prior endpoint for tensor fidelity, distributional realism, and Doppler recovery.

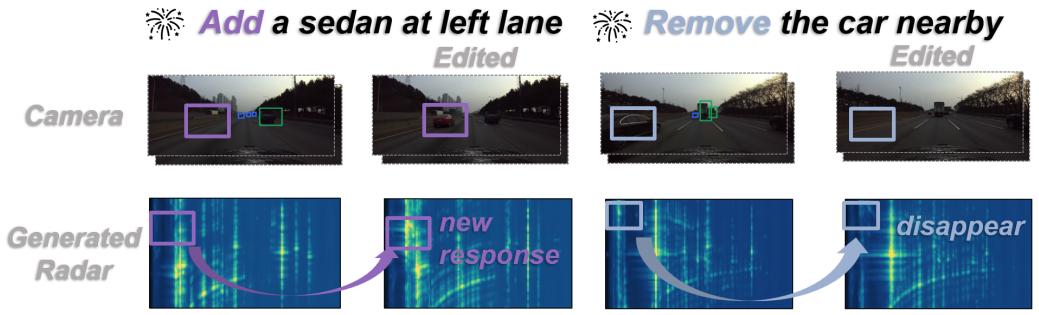



Figure 4: Illustrative image-editing cases. Adding or removing a vehicle in the camera image changes its corresponding radar response.

**Illustrative editing cases.** We examine how image edits affect the recovered radar responses in Figure 4. Adding a sedan in the left lane produces an additional local response, while removing the nearby vehicle suppresses its radar response. These cases show object-level control over the generated radar tensor. The appendix shows further edits that add a pedestrian, truck, or bicycle, or replace a sedan with a truck. Such control offers a route to reducing object-class imbalance and expanding coverage of long-tail scenes in radar datasets.

### 4.3 Downstream Detection Evaluation

**Benchmarks.** We evaluate detector utility under two settings. In the synthetic setting, detectors are trained on generated radar from the validation split and tested on real test data; the real-data reference uses radar from the same validation split. In the augmentation setting, generated validation-split radar is added to real training data, and the reference uses real training data alone. We use RTNH, the official K-Radar 4D radar tensor detector (Paek et al., 2022), as the main downstream benchmark, and SGDet3D, a representative camera-radar multimodal detector (Bai et al., 2025), as a second perception stack.

Table 2: Detection results on RTNH and SGDet3D. Achievement is relative to the matched real-data baseline.

| | | Synthetic Setting | | | | Augmentation Setting | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | IoU 0.3 | | IoU 0.5 | | IoU 0.3 | | IoU 0.5 | |
| Detector | Training data | $AP_{\text{BEV}}$ | $AP_{\text{3D}}$ | $AP_{\text{BEV}}$ | $AP_{\text{3D}}$ | $AP_{\text{BEV}}$ | $AP_{\text{3D}}$ | $AP_{\text{BEV}}$ | $AP_{\text{3D}}$ |
| *RTNH* | Real data ref. | 67.91 | 64.84 | 58.15 | 42.77 | 68.53 | 65.66 | 60.85 | 44.79 |
| | Pix2PixHD RAE | 18.09 | 17.45 | 14.36 | 11.96 | 69.22 | 66.01 | 61.25 | 43.38 |
| | Latent Pix2PixHD | 16.08 | 14.06 | 12.65 | 11.04 | 68.61 | 65.59 | 60.05 | 43.92 |
| | Latent Diffusion | 12.54 | 10.51 | 7.49 | 5.48 | 68.56 | 65.87 | 60.77 | 44.90 |
| | **Ours** | **57.65** | **54.41** | **45.65** | **30.46** | **71.35** | **68.18** | **61.79** | **46.28** |
| | **Achievement** | **84.9%** | **83.9%** | **78.5%** | **71.2%** | **104%** | **104%** | **102%** | **103%** |
| *SGDet3D* | Real data ref. | 69.527 | 65.711 | 54.701 | 39.612 | 73.297 | 70.869 | 60.493 | 47.579 |
| | **Ours** | **59.345** | **55.675** | **42.624** | **30.606** | **76.828** | **75.156** | **64.054** | **51.020** |
| | **Achievement** | **85.3%** | **84.7%** | **77.9%** | **77.3%** | **105%** | **106%** | **106%** | **107%** |

**Metrics.** Following RTNH and SGDet3D evaluation protocols, we report Average Precision (AP) for BEV and 3D boxes at IoU thresholds 0.3 and 0.5. All generated and real radar tensors are exported with the same sparse-radar preprocessing before detector training.

**Results.** Table 2 shows that our generated tensors provide stronger standalone supervision than the competing generated data. When added to real training data, they improve both RTNH and SGDet3D over their real-data references, including at the stricter IoU threshold. The gains across radar-only and camera-radar detectors indicate that the generated tensors retain structure useful to different perception stacks.

### 4.4 Component Analysis

**Visual-prior channels.** Table 1 shows different effects for the two ablations. Removing permittivity causes the larger loss in MAE, PSNR, and SSIM, whereas removing velocity causes the larger loss in Doppler PSNR. Density remains in both ablations, so these differences identify information beyond projected occupancy: permittivity informs radar return strength, while radial velocity informs the Doppler response. Both removals also worsen FID. The channels therefore make complementary contributions to the generated tensor rather than affecting separate outputs in isolation.

**Bridge formulation.** Under fixed VAEs and data exposure, the bridge outperforms adversarial latent mapping and conditional latent diffusion in both tensor synthesis and downstream detection. Conditional latent diffusion uses the visual prior as an external condition, whereas the bridge starts from its latent representation and models a trajectory toward the paired radar latent. Latent Pix2PixHD also maps between the paired representations, but uses direct adversarial translation. The shared VAE setup makes this a comparison of translation formulations, supporting the bridge's source-to-target trajectory for the present task.

## 5 Conclusion

We introduce image-to-4D radar tensor recovery for complete radar data generation from low-cost visual inputs. We propose a latent diffusion bridge framework that organizes geometry, permittivity, and motion cues into a radar-aligned visual-prior tensor and translates it into complete range-azimuth-elevation-Doppler radar tensors. Dual-VAE compression keeps the iterative translation in compact latent grids, and accelerated sampling recovers the radar latent in 30 steps before full-resolution decoding. Experiments on K-Radar show improved tensor synthesis over controlled image-conditioned baselines. The generated tensors support standalone detector training and improve detection through data augmentation on both RTNH and SGDet3D.

### AI USE STATEMENT

We used generative AI tools to polish selected sentences for clarity and readability. We reviewed the AI-assisted wording and take responsibility for the final content of the submission.

## REFERENCES


Xiaokai Bai, Zhu Yu, Lianqing Zheng, Xiaohan Zhang, Zili Zhou, Xue Zhang, Fang Wang, Jie Bai, and Hui-Liang Shen. SGDet3D: Semantics and geometry fusion for 3D object detection using 4D radar and camera. *IEEE Robotics and Automation Letters*, 10(1):828–835, 2025. doi: 10.1109/LRA.2024.3513041.

Oded Bialer and Yuval Haitman. RadSimReal: Bridging the gap between synthetic and real data in radar object detection with simulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15407–15416, 2024. doi: 10.1109/CVPR52733.2024.01459.

Tomer Borreda, Fangqiang Ding, Sanja Fidler, Shengyu Huang, and Or Litany. RadarGen: Automotive radar point cloud generation from cameras, 2025. arXiv preprint arXiv:2512.17897.

David Borts, Erich Liang, Tim Broedermann, Andrea Ramazzina, Stefanie Walz, Edoardo Palladin, Jipeng Sun, David Brueggemann, Christos Sakaridis, Luc Van Gool, Mario Bijelic, and Felix Heide. Radar fields: Frequency-space neural scene representations for FMCW radar. In *ACM SIGGRAPH 2024 Conference Papers*, New York, NY, USA, 2024. Association for Computing Machinery. doi: 10.1145/3641519.3657510.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. SAM 3: Segment anything with concepts, 2025. arXiv preprint arXiv:2511.16719.

Eduardo C. Fidelis, Fabio Reway, Herick Y. S. Ribeiro, Pietro L. Campos, Werner Huber, Christian Icking, Lester A. Faria, and Torsten Schoen. Generation of realistic synthetic raw radar data for automated driving applications using generative adversarial networks, 2023. arXiv preprint arXiv:2308.02632.

Rossiza Gourova, Oleg Krasnov, and Alexander Yarovoy. Analysis of rain clutter detections in commercial 77 GHz automotive radar. In *Proceedings of the 14th European Radar Conference*, pp. 25–28, 2017. doi: 10.23919/EURAD.2017.8249138.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Martin Holder, Clemens Linnhoff, Philipp Rosenberger, and Hermann Winner. The Fourier tracing approach for modeling automotive radar sensors. In *Proceedings of the 20th International Radar Symposium*, pp. 1–8, 2019. doi: 10.23919/IRS.2019.8768113.

Woo-Jin Jung, Dong-Hee Paek, and Seung-Hyun Kong. LiDAR-to-4D radar synthesis for building large-scale tensor datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Findings*, pp. 889–899, 2026.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

Pou-Chun Kung, Skanda Harisha, Ram Vasudevan, Aline Eid, and Katherine A. Skinner. RadarSplat: Radar Gaussian splatting for high-fidelity data synthesis and 3D reconstruction of autonomous driving scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27596–27606, 2025.

Jimmie Kwok, Holger Caesar, and Andras Palffy. 4D-RaDiff: Latent diffusion for 4D radar point cloud generation, 2025. arXiv preprint arXiv:2512.14235.

Bo Li, Kaitao Xue, Bin Liu, and Yu-Kun Lai. BBDM: Image-to-image translation with Brownian bridge diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1952–1961, 2023. doi: 10.1109/CVPR52729.2023.00194.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views, 2025. arXiv preprint arXiv:2511.10647.

Zoltan Ferenc Magosi, Hexuan Li, Philipp Rosenberger, Li Wan, and Arno Eichberger. A survey on modelling of automotive radar sensors for virtual test and validation of automated driving. *Sensors*, 22(15):5693, 2022. doi: 10.3390/s22155693.

Dong-Hee Paek, Seung-Hyun Kong, and Kevin Tirta Wijaya. K-Radar: 4D radar object detection for autonomous driving in various weather conditions. In *Advances in Neural Information Processing Systems*, volume 35, pp. 3819–3829, 2022. doi: 10.52202/068431-0276.

Mahan Rafidashti, Ji Lan, Maryam Fatemi, Junsheng Fu, Lars Hammarstrand, and Lennart Svensson. NeuRadar: Neural radiance fields for automotive radar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2025. doi: 10.1109/CVPRW67362.2025.00234.

Pavan Aakash Rangaraj, Tunc Alkanat, and Ashish Pandharipande. RAIDS: Radar range-azimuth map estimation from image, depth, and semantic descriptions. *IEEE Sensors Journal*, 25(7):12381–12388, 2025. doi: 10.1109/JSEN.2025.3540660.

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

Dazhong Shen, Jingjing Gu, Qiang Zhou, Meng Zhao, and Ying Sun. LiDAR-to-4DRadar diffusion bridge via cross-modal alignment and translation in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17110–17120, 2026.

Jessica Smith, Richard Dudley, David Jones, David Cheadle, Imran Mohamed, Fengping Li, Daniel Bownds, Hsun Yang, Joel Rapley, Andre Burgess, Mira Naftaly, Mayokun Aikomo, Jeremy Price, Nawal Husnoo, Stephan Havemann, Matthew Fry, Martin Osborne, James McGregor, and Alan Vance. Impact of UK weather on autonomous vehicle radar, LiDAR and camera vision systems. *Sensors*, 26(17):5649, 2026. doi: 10.3390/s26175649.

Leichen Wang, Bastian Goldluecke, and Carsten Anklam. L2R GAN: LiDAR-to-radar translation. In *Proceedings of the Asian Conference on Computer Vision*, volume 12624 of *Lecture Notes in Computer Science*, pp. 704–720. Springer, 2021. doi: 10.1007/978-3-030-69535-4_43.

Shengpeng Wang, Xin Luo, Yulong Xie, and Wei Wang. SDDiff: Boosting radar perception via spatial-Doppler diffusion. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 8805–8813, 2025. doi: 10.24963/ijcai.2025/979.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional GANs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8798–8807, 2018. doi: 10.1109/CVPR.2018.00917.

Weiqing Xiao, Hao Huang, Chonghao Zhong, Yujie Lin, Nan Wang, Xiaoxue Chen, Zhaoxi Chen, Saining Zhang, Shuocheng Yang, Pierre Merriaux, Lei Lei, and Hao Zhao. Controllable radar simulation with waveform parameter embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Findings*, pp. 6424–6434, 2026.

Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, Fisher Yu, Dacheng Tao, and Andreas Geiger. Unifying flow, stereo and depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):13941–13958, 2023. doi: 10.1109/TPAMI.2023.3298645.

Jiarui Zhang, Zhihao Li, Chong Wang, and Bihan Wen. RF4D: Neural radar fields for novel view synthesis in outdoor dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15387–15397, 2026a.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018. doi: 10.1109/CVPR.2018.00068.

Ruibin Zhang, Donglai Xue, Yuhan Wang, Ruixu Geng, and Fei Gao. Towards dense and accurate radar perception via efficient cross-modal diffusion model. *IEEE Robotics and Automation Letters*, 9 (9):7429–7436, 2024. doi: 10.1109/LRA.2024.3426389.

Ruijie Zhang, Bixin Zeng, Shengpeng Wang, Fuhui Zhou, and Wei Wang. RaLD: Generating high-resolution 3D radar point clouds with latent diffusion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18773–18781, 2026b. doi: 10.1609/aaai.v40i22.38946.

# Appendix

## A Dataset and Fixed Partitions

### A.1 K-Radar subset

We use synchronized front-view camera images, camera calibration, dense radar tensors, and 3D annotations from K-Radar (Paek et al., 2022). A sample is retained when the aligned image, calibration, visual-prior tensor, and radar target can all be constructed. This filtering yields 9,008 paired samples. We use one fixed 4:2:1 partition for all methods (Table 3). The training set is used to optimize the two VAEs and the diffusion bridge; the validation set is used for checkpoint and experimental selection; the test set is used only for final tensor-level reporting. No test sample is used to fit normalization statistics or select checkpoints.

Table 3: Fixed K-Radar partitions used by RaViP-Bridge and the controlled baselines.

| Partition | Samples | Fraction | Use |
|---|---|---|---|
| Train | 5,147 | $4/7$ | Parameter optimization |
| Validation | 2,574 | $2/7$ | Model selection and expansion pool |
| Test | 1,287 | $1/7$ | Final tensor evaluation |
| Total | 9,008 | 1 | |

### A.2 Sample identity and pairing

Each record stores a scene identifier, radar frame identifier, partition label, source tensor or source latent, and target tensor or target latent. The pair key is the tuple of scene and radar frame identifiers. This key is preserved when exporting generated radar to downstream detectors, which prevents a prediction from being evaluated against a different camera–radar pair. Metadata order is not treated as sample identity.

## B Radar-Aligned Visual-Prior Construction

### B.1 Frame preparation and calibration

For radar frame $\tau$, we use the aligned front-view image $I_\tau$ and a forward image $I_{\tau+\Delta\tau}$. Camera timestamps determine $\Delta\tau$; the implementation uses a five-camera-frame offset when the corresponding forward frame is available. Images are rectified before visual inference. Camera intrinsics back-project pixels into 3D, and the provided camera-to-radar extrinsics transform the points to the radar coordinate system.

We use Depth Anything 3 for metric scene depth, SAM 3 for semantic regions, and UniMatch for optical flow (Lin et al., 2025; Carion et al., 2025; Xu et al., 2023). The visual models remain frozen. Their outputs are used to construct the source representation; they are not trainable components of the diffusion bridge.

### B.2 Geometry, material, and motion cues

For a valid pixel with homogeneous coordinate $\tilde{u}_i$, depth $d_i$, camera matrix $K$, and camera-to-radar transform $(R_{r\leftarrow c}, t_{r\leftarrow c})$, its radar-frame location is

$$p_i^r = R_{r\leftarrow c}\left(d_i K^{-1}\tilde{u}_i\right) + t_{r\leftarrow c}. \tag{17}$$

We map semantic regions to coarse material categories and then to representative relative-permittivity values. These values are priors, not estimates of instance-specific electromagnetic parameters.

Optical flow links a pixel between $I_\tau$ and $I_{\tau+\Delta\tau}$. After back-projection at both times, we project relative motion onto the radar line of sight:

$$v_i = \frac{(p_i^{r,\tau+\Delta\tau} - p_i^{r,\tau})^\top p_i^{r,\tau}}{\Delta\tau \, \|p_i^{r,\tau}\|_2}. \tag{18}$$

Depth discontinuities around foreground boundaries and isolated flow outliers can create long trails or implausible velocities. We apply object-trail depth filtering, mask-boundary depth correction, and connected-component velocity smoothing before voxelization. Invalid motion estimates are excluded from velocity aggregation.

### B.3 Polar voxelization

We accumulate the projected elements on the radar range–azimuth–elevation (RAE) grid. We use the native K-Radar axes, remove points closer than 3 m, retain the first 192 range bins, and center-crop the angular axes from 107 to 96 azimuth bins and from 37 to 32 elevation bins. Axis orientation is made identical to the radar loader before the crop.

The source tensor is

$$S = (S_\rho, S_\varepsilon, S_v) \in \mathbb{R}^{3\times 192\times 96\times 32}. \tag{19}$$

Within each voxel, $S_\rho$ stores projected-point count, $S_\varepsilon$ stores the material prior selected by semantic voting, and $S_v$ stores mean valid radial velocity. Empty voxels are zero. The source-channel order supplied to the VAE is density, permittivity, and radial velocity. The occupancy mask $M_{\text{vp}}$ marks projected occupied voxels and is used for velocity supervision.

## C Radar Target Preparation

The native radar measurement contains 64 Doppler bins over a $256 \times 107 \times 37$ RAE grid. We construct a 64-channel learning target comprising one dense RAE log-power channel and 63 Doppler-response channels. The RAE channel is retained densely. Doppler spectra are retained at valid radar hits; non-hit locations are zero. We use the same $192 \times 96 \times 32$ spatial crop as the visual prior.

We transform radar power and valid Doppler responses to log scale and normalize them with training-set statistics. The radar power statistics are mean $10.9738$ and standard deviation $0.3563$; the valid Doppler statistics are mean $10.9813$ and standard deviation $0.8873$. Doppler normalization is performed on valid responses before scattering them into a zero background. We retain a binary hit mask $M_r$ for mask-aware reconstruction and Doppler evaluation.

## D Network and Optimization Details

### D.1 Dual 3D VAEs

The visual-prior and radar VAEs have the same spatial structure but do not share weights. Each uses three encoder and decoder stages, two residual layers per stage, channel widths $(64, 128, 256)$, and attention in the lowest-resolution encoder and highest-resolution decoder stage. Both produce a four-channel latent grid of size $48 \times 24 \times 8$ from a $192 \times 96 \times 32$ input grid. The visual-prior VAE accepts three channels; the radar VAE accepts and reconstructs 64 channels. The radar decoder also predicts the hit mask.

We optimize the radar VAE first, using the reconstruction, KL, and mask losses defined in the main paper. We then freeze the radar VAE and optimize the visual-prior VAE with channel-specific reconstruction, KL regularization, and occupancy-mask prediction. Velocity reconstruction is supervised on $M_{\text{vp}}$.

### D.2 Latent diffusion bridge

After both VAEs are frozen, the visual-prior and radar latents define the source and target endpoints of the bridge. We use the main paper's convention: radar latent $z_r$ at bridge time $t = 0$ and visual-prior

latent $z_{\text{vp}}$ at $t = 1$. We form an intermediate state as

$$z_t = (1-t)z_r + tz_{\text{vp}} + \sigma\sqrt{t(1-t)}\,\epsilon, \qquad \epsilon \sim \mathcal{N}(0, I). \tag{20}$$

For $0 < t < 1$, the 3D U-Net is trained to predict the reverse drift $(z_r - z_t)/t$ from $z_t$ and its time embedding. We use 1,000 training timesteps with a linear schedule and $\sigma = 1.0$. The endpoint $t = 0$ is excluded when sampling training targets because the drift denominator vanishes there.

The U-Net uses four resolution stages with widths $(128, 256, 512, 512)$, two residual blocks per stage, attention in the two lowest-resolution stages, SiLU activations, and positional timestep embeddings. We use 30 deterministic steps for inference, initializing the sampler at $z_{\text{vp}}$ and decoding the resulting radar latent with the radar VAE.

Table 4: Configuration of the reported bridge model.

| Item | Setting |
|---|---|
| Visual-prior input | $3 \times 192 \times 96 \times 32$ |
| Radar target | $64 \times 192 \times 96 \times 32$ |
| Latent grid | $4 \times 48 \times 24 \times 8$ |
| U-Net widths | 128, 256, 512, 512 |
| Training timesteps | 1,000 |
| Inference | 30 deterministic steps |
| Optimizer | AdamW |
| Learning rate | $1 \times 10^{-4}$ |
| Bridge batch size | 8 |
| Precision | FP16 mixed precision |
| Seed | 42 |
| Selected checkpoint | 450,000 updates |

## E Training and Inference Details

We train the model in three stages following the dependencies among radar representation learning, visual-prior representation learning, and bridge translation. We first train the radar VAE on real radar tensors. We then freeze the radar VAE and train the visual-prior VAE with reconstruction, KL, and mask losses. Finally, we freeze both VAEs, cache paired visual-prior and radar latents, and train the diffusion bridge U-Net on these pairs. The latent cache is shared with the latent-space baselines. We select the bridge checkpoint on the validation partition and evaluate the selected checkpoint on the test partition. For bridge training, we sample times strictly between 0 and 1.

At inference, we use the visual-prior encoder, the bridge network, and the radar decoder. We encode the source once and recover the radar latent through successive drift predictions using the 30-step deterministic sampler described in the main paper. Algorithm 1 gives the complete recovery procedure. The radar encoder and visual-prior decoder are used only during training.

**Algorithm 1** Image-to-4D Radar Recovery via Latent Bridge

**Input:** Image pair and calibration; frozen $E_{\text{vp}}$, $v_\theta$, $D_r$; decreasing sampling times $\mathcal{T}$ from 1 to 0.
**Output:** Recovered radar tensor $\hat{R}$.

Construct radar-aligned $S = (S_\rho, S_\varepsilon, S_v)$.
Encode $z_{\text{vp}} \leftarrow E_{\text{vp}}(S)$; initialize $z_1 \leftarrow z_{\text{vp}}$.
**for** successive times $(t, u)$ in $\mathcal{T}$, with $t > u$ **do**
  Predict drift $\hat{v} \leftarrow v_\theta(z_t, t)$ with the 3D U-Net.
  Estimate radar endpoint $\hat{z}_r \leftarrow z_t + t\hat{v}$.
  **if** $t = 1$ **then**
    $\hat{\epsilon} \leftarrow 0$ {Zero variance at the source}
  **else**
    $\hat{\epsilon} \leftarrow [z_t - (1-t)\hat{z}_r - tz_{\text{vp}}]/\sigma_t$.
  **end if**
  $z_u \leftarrow (1-u)\hat{z}_r + uz_{\text{vp}} + \sigma_u\hat{\epsilon}$.
**end for**
Decode $(\hat{R}_p, \hat{R}_d, \hat{M}_r) \leftarrow D_r(z_0)$.
Denormalize power and Doppler responses; assemble $\hat{R}$ using the predicted valid-hit mask $\hat{M}_r$.
**return** $\hat{R}$.

# F Evaluation Protocol

## F.1 Direct tensor metrics

We evaluate all 1,287 test samples with a fixed sample manifest. We generate and store heavy tensors in chunks of 128, but aggregate scalar metrics over samples. We compute FID once from the complete set of 1,287 generated and target Inception features; per-chunk FID values are not averaged.

We evaluate MAE, PSNR, and SSIM after converting the RAE power representation to a Cartesian ZYX volume. LPIPS and FID use log-scaled 2D projections of the same representation. Doppler PSNR is evaluated on valid target Doppler regions. Table 5 reports the full-test dispersion omitted from the main paper.

Table 5: Full-test metrics for the selected RaViP-Bridge checkpoint. FID is a single global statistic and has no per-sample standard deviation.

| Metric | Mean | Std. |
|---|---|---|
| Cartesian MAE ↓ | 0.0560 | 0.0285 |
| Cartesian PSNR ↑ | 34.312 | 5.801 |
| Cartesian SSIM ↑ | 0.9106 | 0.0834 |
| LPIPS ↓ | 0.0281 | 0.0331 |
| Doppler PSNR ↑ | 24.228 | 1.295 |
| Hit-mask IoU ↑ | 0.4290 | 0.2878 |
| Global FID ↓ | 12.666 | |

## F.2 Controlled baselines

Pix2PixHD RAE predicts only the dense RAE response and does not model Doppler. Latent Pix2PixHD replaces the bridge with adversarial mapping between the same visual-prior and radar latent spaces. Conditional latent diffusion starts from Gaussian noise and conditions denoising on the visual-prior latent. The two latent baselines use the same frozen VAEs, paired latent cache, normalization, and partitions as RaViP-Bridge. Their comparison isolates the latent translation mechanism from the visual front end and radar decoder.

## F.3 Downstream detection

We convert generated and real radar tensors to the same sparse detector input. In the synthetic setting, a detector is trained on generated validation-partition tensors and tested on real test data; its real-

data reference is trained on the corresponding real validation partition. In the augmentation setting, generated validation-partition tensors are added to the real training partition, and the reference uses real training data alone.

For SGDet3D, the RTNH-style evaluator uses a score threshold of 0, excludes test frames with empty ground truth, and evaluates 1,179 of 1,287 test records. We select checkpoints on validation results, not test results. The main paper reports mean BEV and 3D AP over Sedan and Bus-or-Truck. At IoU 0.3, adding generated data improves mean 3D AP from 70.869 to 75.156 and mean BEV AP from 73.297 to 76.828. At IoU 0.5, the corresponding changes are 47.579 to 51.020 and 60.493 to 64.054.

# G Additional Visual Evidence

## G.1 Visual-prior structure under good visibility

Figure 5 shows a camera frame and its projected visual prior. The marked vehicles retain distinct projected regions, while the road and surrounding structures remain identifiable. Figure 6 compares the visual prior with measured LiDAR in three matched scenes. Curved roadside structures and several object-associated clusters occupy corresponding range–azimuth regions, showing that the visual prior retains scene geometry under good visibility.

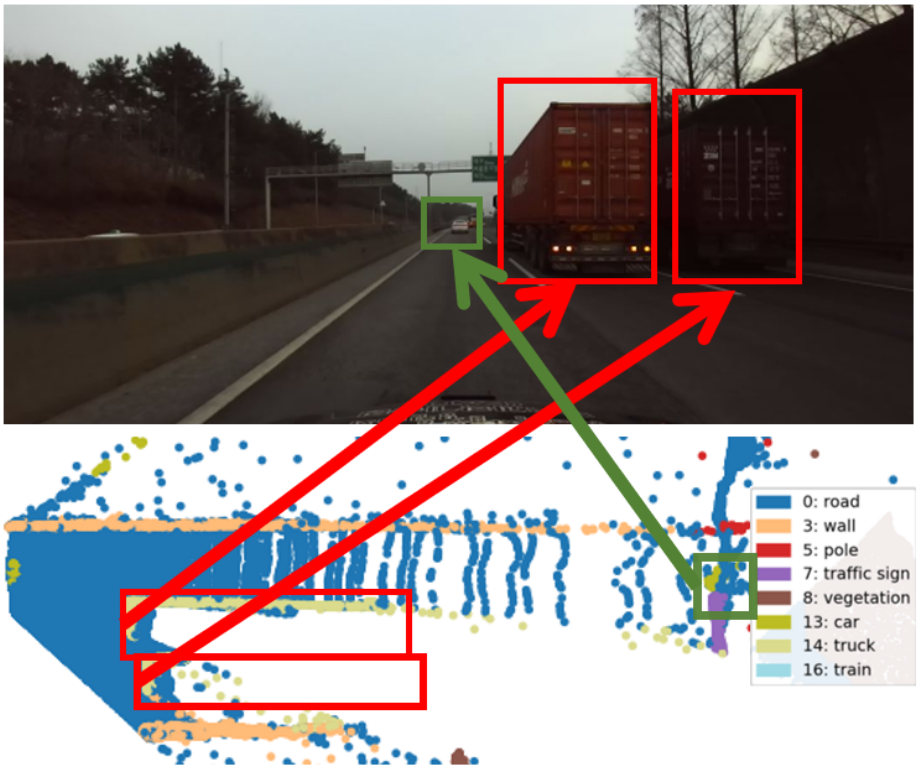



Figure 5: Camera image and projected visual prior from a K-Radar scene. The boxes mark corresponding vehicles; the projection retains their distinct locations together with the surrounding road structure.

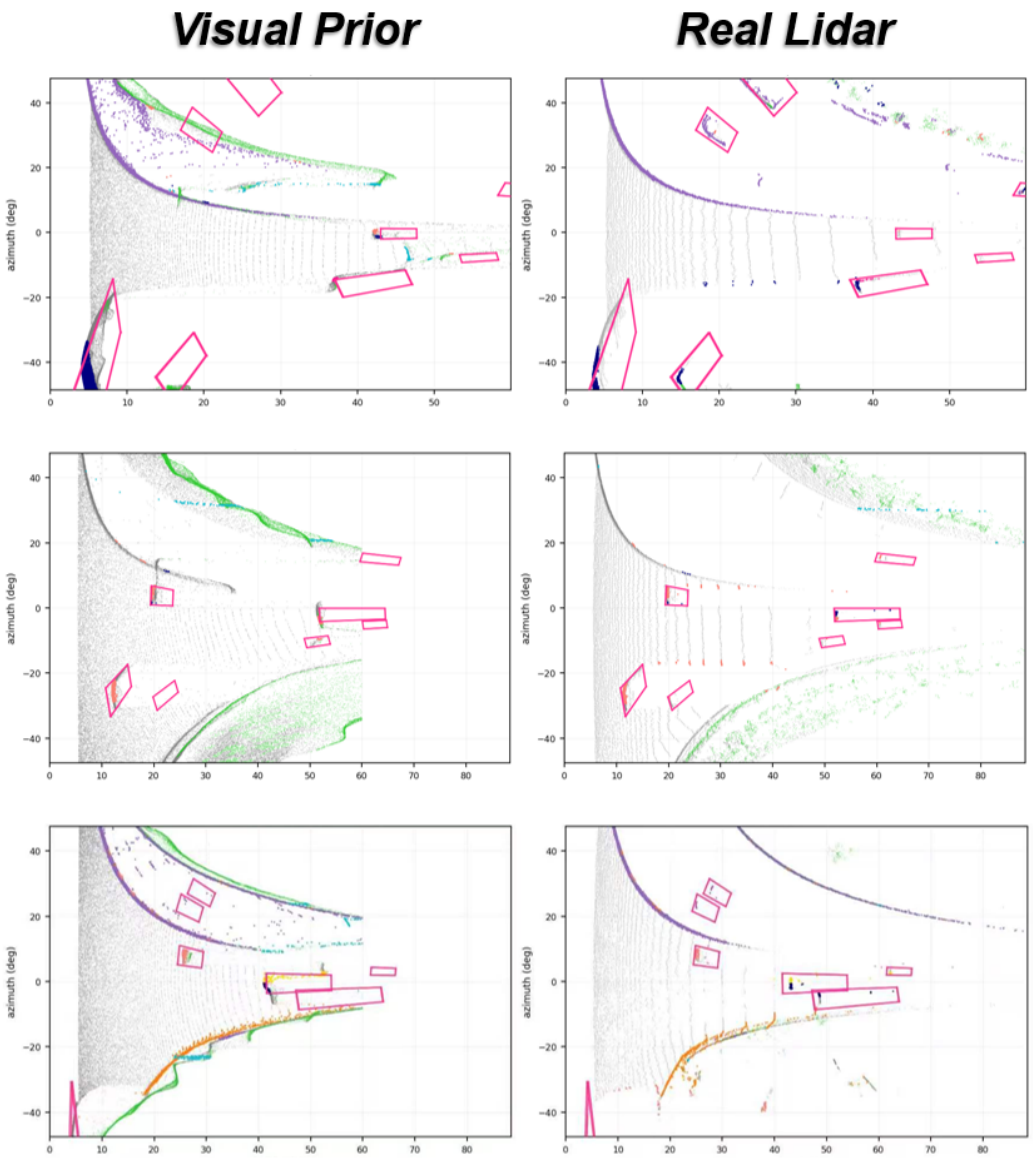



Figure 6: Visual prior (left) and real LiDAR (right) in three matched K-Radar scenes, shown in range–azimuth coordinates. Magenta boxes mark corresponding object locations. Roadside structures and point clusters near several marked objects appear in corresponding regions.

### G.2 Visual-prior extraction on KITTI-360

Figure 7 shows image-derived semantic pseudo-point clouds from KITTI-360. Across four scenes, the top and front views retain the road layout, vegetation, and nearby objects. We construct the point clouds from images and KITTI-360 calibration; both VAEs and the bridge are trained on K-Radar.

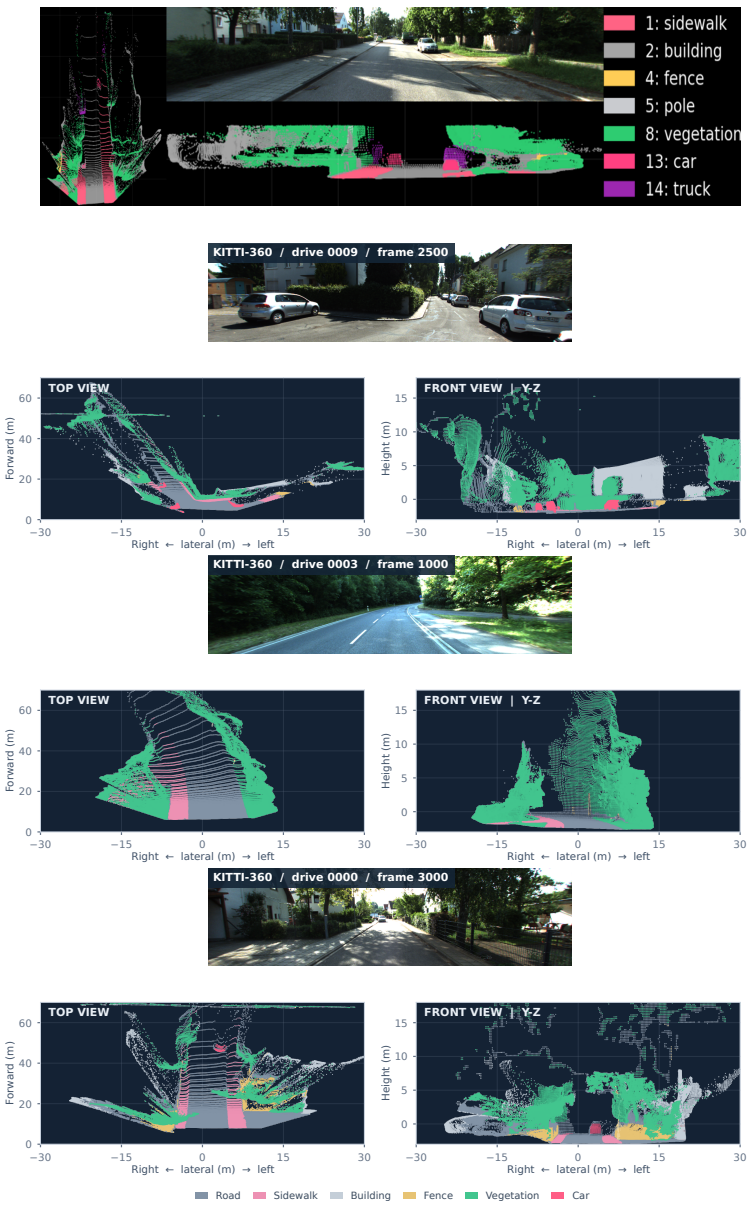



Figure 7: Visual-prior extraction on KITTI-360. The top panel shows a camera frame and two views of its semantic pseudo-point cloud. Below are three additional frames (drives 0009, 0003, and 0000), each with a camera image, top X–Y view, and front Y–Z view. The new point-cloud views share spatial limits and semantic colors.

### G.3 Visual-prior VAE reconstruction

Figure 8 compares the visual-prior density channel with its reconstruction by the frozen source VAE. Across highway, urban, and university scenes, the reconstructions retain the dominant roadside arcs and object-associated clusters. The source VAE thus preserves the spatial layout used to condition the bridge after latent compression.

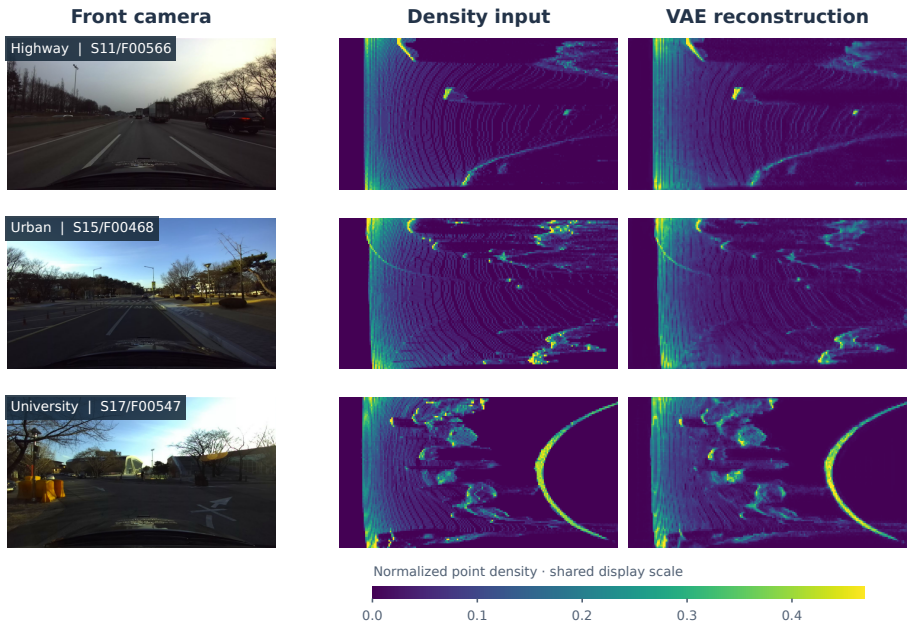



Figure 8: Visual-prior source-VAE density reconstruction on three K-Radar validation frames. Each row contains the synchronized front-view camera image, input density, and VAE reconstruction. The latter two are range–azimuth maps formed by maximum projection over elevation and shown with one shared color scale.

### G.4 Additional qualitative results

Figure 9 shows ten additional validation cases. The generated tensors retain the dominant range–azimuth responses and principal range–Doppler structures across these scenes.

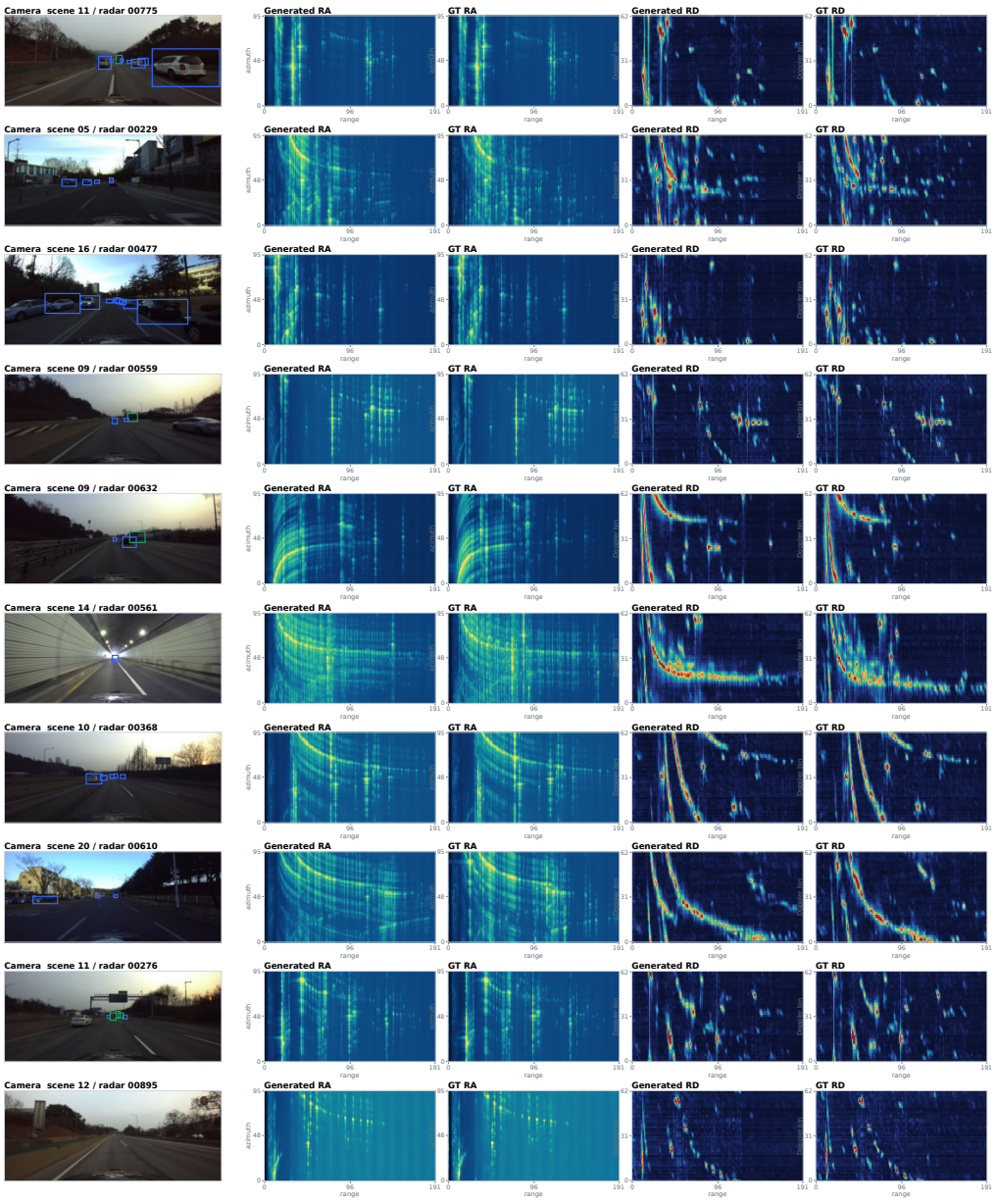



Figure 9: Additional qualitative results on the K-Radar validation partition. Each row shows the camera frame, generated and ground-truth range–azimuth (RA) projections, and generated and ground-truth range–Doppler (RD) projections. RA is the maximum-elevation projection of the decoded power channel. RD is obtained from the decoded Doppler-response channels by maximum projection over azimuth and elevation. All cases use the selected bridge checkpoint and 30 deterministic sampling steps.

### G.5 Additional image-editing cases

Figure 10 extends the image-editing example in the main paper with four edits. Adding a pedestrian strengthens the response at the marked location. Adding a truck or bicycle produces a new local response, while replacing a sedan with a truck changes its response shape. Together, these edits show strong object-level control over the generated radar. This flexibility provides a route to targeted augmentation of underrepresented object configurations within a class.

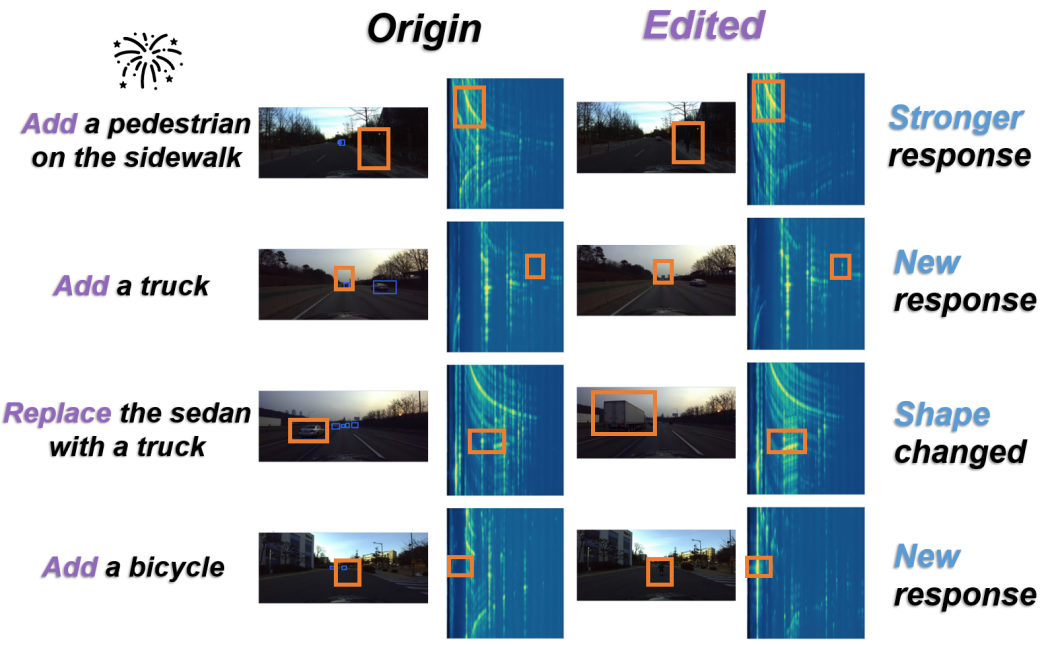



Figure 10: Additional image-editing cases. Each row pairs the original and edited camera images with their generated range–azimuth maps. Orange boxes mark the edited image region and corresponding radar area. From top to bottom, the edits add a pedestrian, add a truck, replace a sedan with a truck, and add a bicycle.

### G.6 Generated radar across baselines

Figure 11 compares range–azimuth maps for one matched case. Our result preserves several localized returns visible in the real radar, whereas the latent diffusion and latent Pix2PixHD baselines lose much of their structure. This visual comparison complements the quantitative results in the main paper.

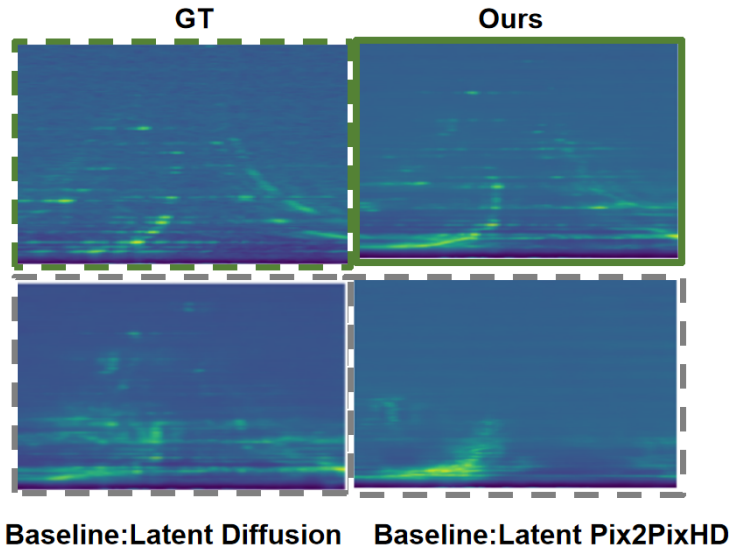



Figure 11: Generated range–azimuth maps for a matched K-Radar case. The top row shows the real radar and our result; the bottom row shows latent diffusion and latent Pix2PixHD. Our map retains more of the target's localized responses, while both baselines appear more diffuse in this example.

## H Error Analysis and Intended Use

### H.1 Range bias from visual depth

Figure 12 isolates a distance error that is less apparent in aggregate metrics. The generated response remains associated with the same visible object, but its peak lies at a slightly greater range than in the real radar. The offset is consistent with a depth error in the visual prior, which places the object farther away before translation.

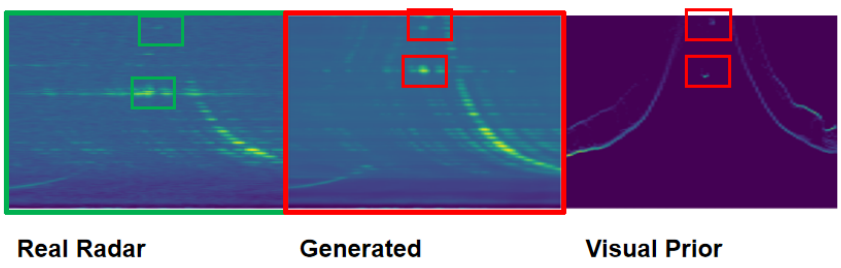



Figure 12: A depth-related range-bias case. The real and generated range–azimuth maps are shown beside the visual prior. The marked object has a generated return at a slightly greater range than its real return, illustrating how an error in projected depth can carry into radar generation.

### H.2 Poor visual evidence

The tunnel scene in Figure 13 shows where the visual route breaks down. The real radar retains returns in the dark tunnel, but the extracted visual prior loses much of its interior structure. The generated radar also departs from the real response in this region. Its change follows the damaged source representation, indicating that the bridge uses projected scene structure when forming radar-like returns. Reliable visual evidence is therefore required for accurate generation.

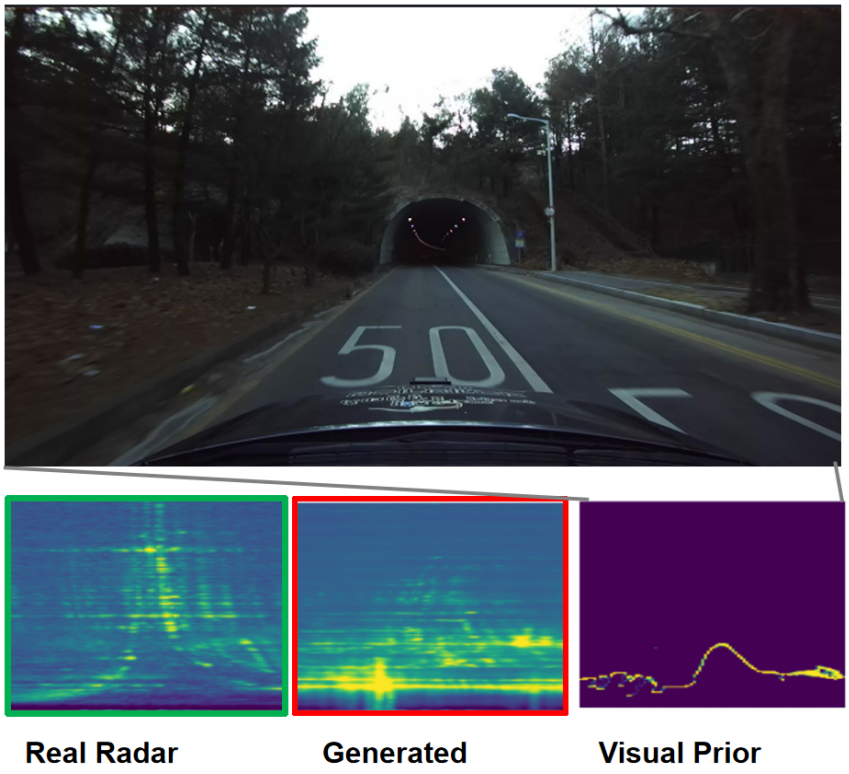



Figure 13: A tunnel case with poor visual evidence. The camera image appears above the real radar, generated radar, and visual prior. The prior loses structure inside the dark tunnel, and the generated response differs from the real radar in the same region, exposing the method's dependence on visible scene geometry.

We use well-visible images to expand radar datasets. Automotive radar can retain target returns when darkness, fog, or rain reduces image visibility (Gourova et al., 2017; Smith et al., 2026). In a controlled test of a 77-GHz automotive radar, fog reduced optical visibility to 17–20 m without a measured change in the radar detections (Gourova et al., 2017). This makes it possible to generate radar from a clear image and pair it with a low-visibility image of the same scene, provided the sensor viewpoint, object locations, and motion are matched. The pairing preserves scene and motion correspondence even when weather changes return strength.