CVPR 2026 · DENVER

CVPR
JUNE 3-7, 2026
DENVER
COLORADO

# Generalizable
## *Radio-Frequency Radiance Fields*
## for Spatial Spectrum Synthesis

**Kang Yang** · **Yuning Chen** · **Wan Du**

*University of California, Merced*

`github.com/kangyang73/GRaF`

1 THE PROBLEM

# Wireless systems run on data we cannot afford to collect

### 5G, 6G, WiFi — everywhere

Networks anchor smart cities, healthcare, and sensing pipelines.

### Sensing needs dense RF data

DNN-based localization and channel models are data-hungry.

### Site surveys are the bottleneck

Measuring spectra at thousands of locations is slow, costly, manual.

1 WHAT WE SYNTHESIZE

# Spatial spectrum: power across all directions at a receiver

From the receiver's antenna array, signal arrives from many directions at once.

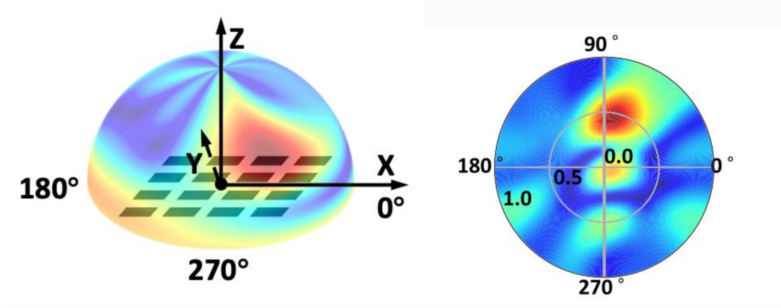
Power per direction (azimuth, elevation) forms a 2D heatmap on the unit sphere.

### Why downstream tasks care

**Angle of Arrival estimation**

Localization, beam steering, channel modeling

*A faithful spectrum drives the rest of the wireless stack.*



*3D sphere view (a) and 2D projection (b) of one spatial spectrum.*

2 WHY THIS IS HARD

# Today's RF NeRFs train one model per scene

🚫 RAY TRACING

### Needs CAD models

Accurate geometry is rarely available and the cost is prohibitive.

🚫 NeRF² / NeWRF

### Scene-specific MLP

Overfits the training scene; new layout means retraining from scratch.

🚫 3DGS-BASED

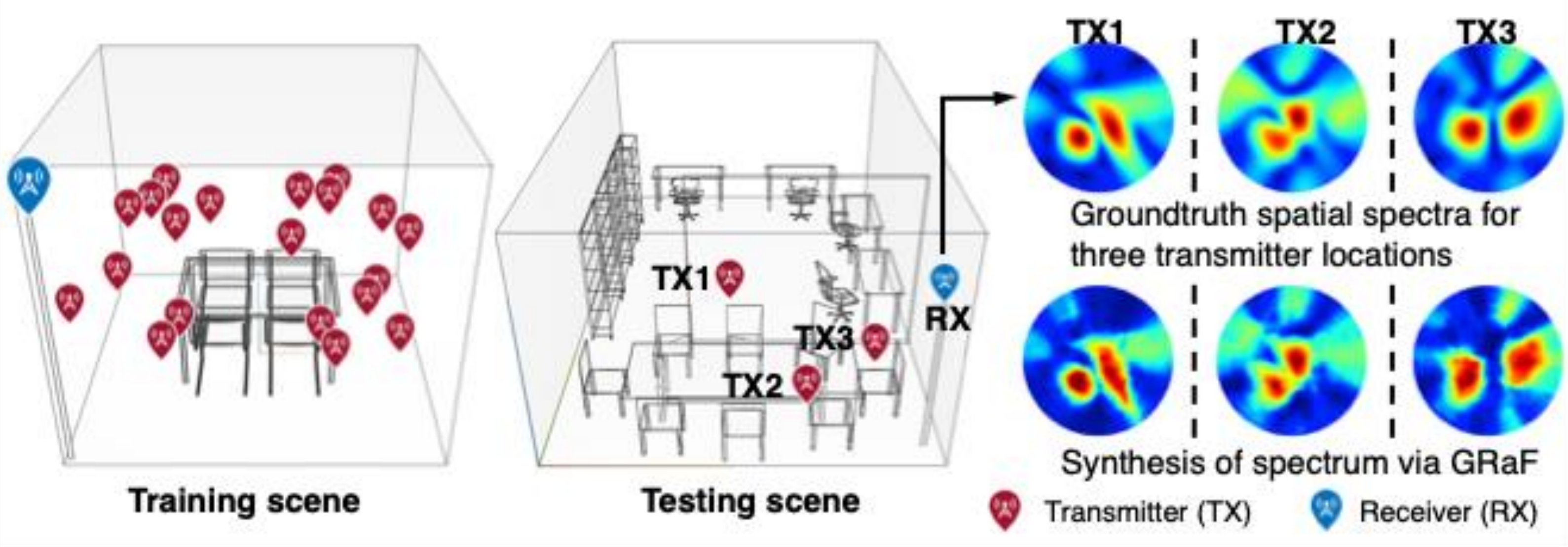
### Faster, still single-scene

Improves speed within a scene, no cross-scene generalization.

*Every new room, every new layout — retrain. The bottleneck just moved.*

3 WHAT WE WANT

# Synthesize spectra at any transmitter location, in any scene



> *Train once on a scene — synthesize spectra for new transmitter positions, even when the room itself changes.*

# Spectra of nearby transmitters interpolate the target

***Theorem 1 (Spectrum Interpolation)***

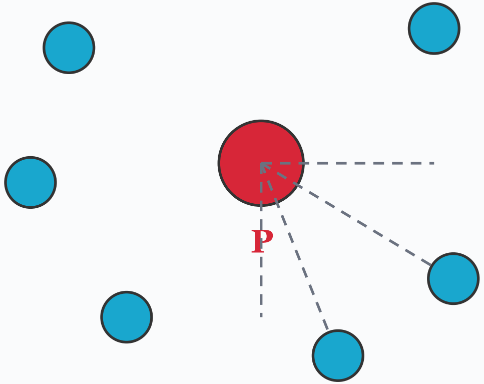
For a target transmitter at **P** with *L* nearest neighbors:

$$SS(P) \approx \Sigma\, w_i\, SS_i$$

**Error bound:** $\varepsilon \leq K \cdot \delta^2$

*δ = neighborhood radius · K = environment curvature*

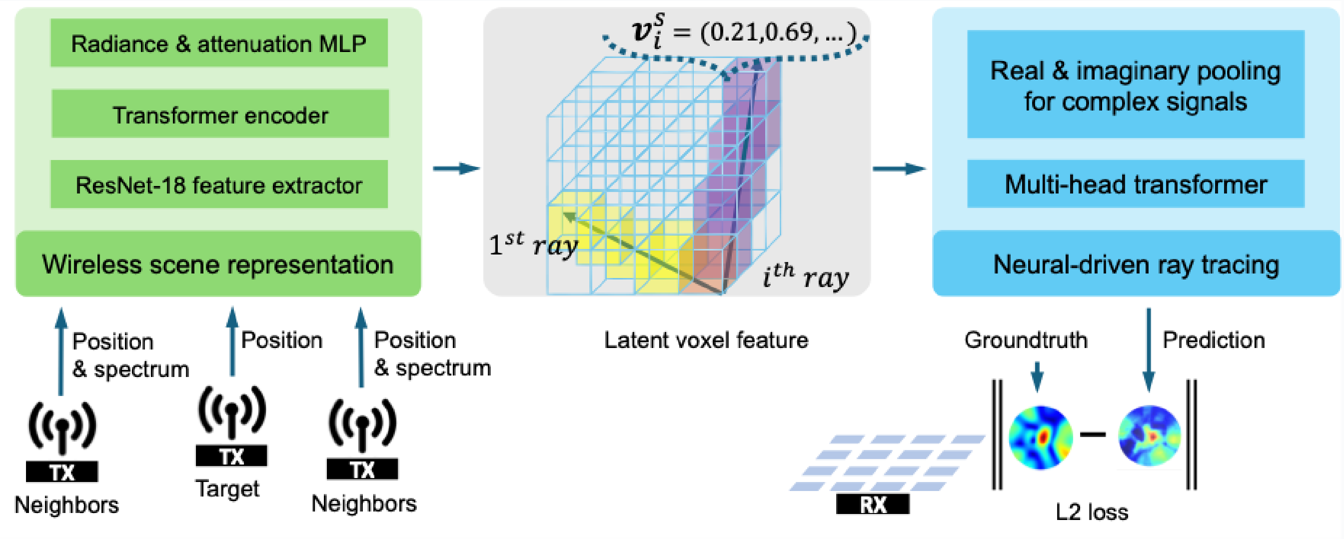
Smaller neighborhoods → tighter approximation.



*Target P (red) is interpolated from its L nearest neighbor transmitters (teal).*

5 **GRaF PIPELINE**

# Two components: a latent RF radiance field + neural ray tracing



**① Latent RF radiance field**
Geometry-aware Transformer over neighbor spectra

**② Neural ray tracing**
Complex-valued aggregation per direction

# Encode neighbor spectra + geometry into a latent Z

**ResNet-18 over each neighbor spectrum**

Captures directional power patterns per neighbor.

**Positional embedding of ($P_i - P$)**

Preserves geometry between target and neighbors.

**Geometry-aware Transformer + cross-attention**

Dynamically weights neighbors — non-linear interpolation.

**Two MLPs → latent $Z \in \mathbb{R}^d$**

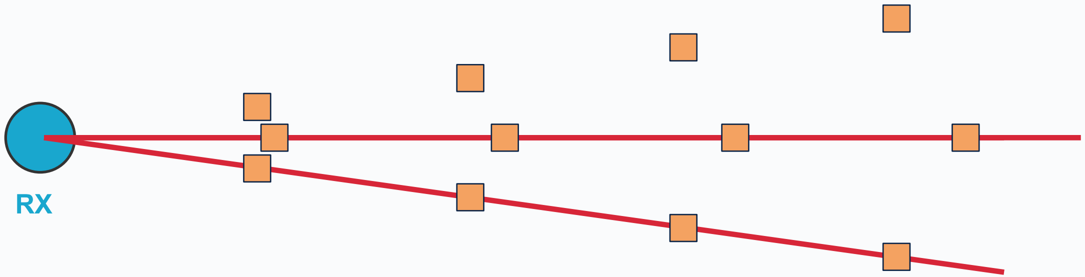
Encodes path loss, multipath, reflection, diffraction.

WHAT Z ENCODES

- Path loss
- Shadowing
- Reflection
- Diffraction
- Scattering
- Multipath

5 NEURAL RAY TRACING

# Trace rays from the receiver, aggregate complex contributions



**PER VOXEL**

Complex radiated signal
+ complex attenuation

$MLP_s(\nu_s)$, $MLP_a(\nu_s)$

**Per ray r — aggregate complex signal across S samples**

$$y_r = \Sigma_s ( \Pi_j a_j ) \cdot s_s \cdot (\lambda / 4\pi d_s) \cdot e^{-j2\pi f d_s/c}$$

→ $SS\Theta(r) = |y_r|^2$

*complex signal*
*+ amplitude/phase attenuation*
*+ free-space path loss and phase shift*

**PER RAY**

Multi-head transformer treats voxels as tokens — attention = cumulative attenuation.

# Datasets, baselines, and four image-style metrics

## DATASETS

**RFID**

915 MHz, 4×4 array, 6,123 TX locations (from NeRF²)

**MATLAB**

Conference + Office layouts × 3 versions each — chairs/tables vary

**Extras**

Bedroom, outdoor, multi-band (928 MHz, 2.4 GHz, 5.8 GHz)

## BASELINES

**KNN**

Average neighbor spectra

**KNN-DL**

Per-pixel learnable weights

**NeRF²**

Scene-specific RF NeRF (= NeWRF without DoA)

## METRICS

**MSE ↓**

Domain loss across directions

**LPIPS ↓**

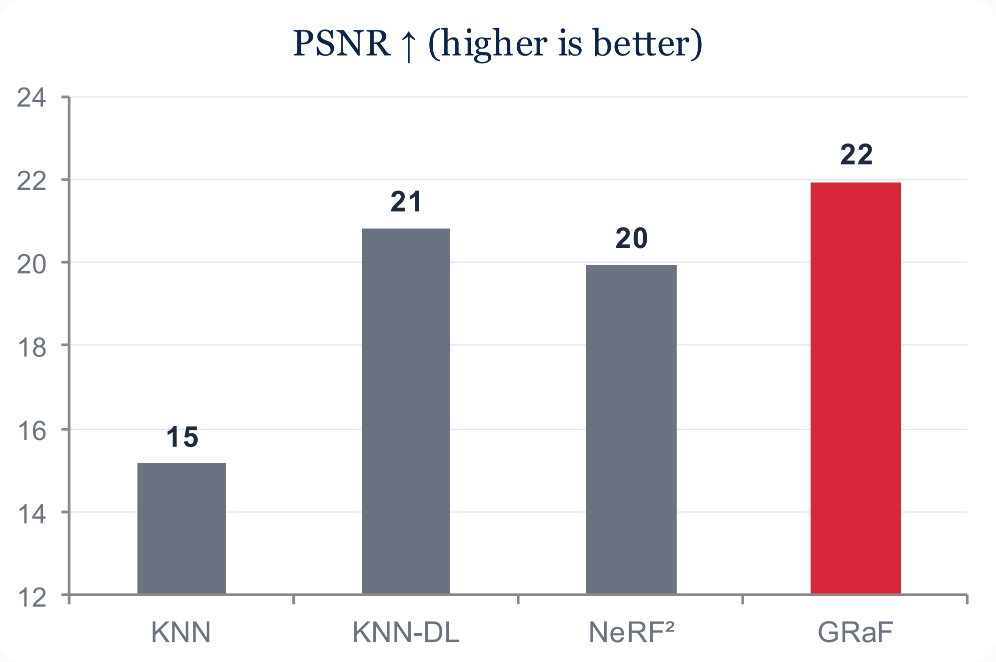
Perceptual similarity

**PSNR ↑ · SSIM ↑**

Pixel + structural quality

6 SINGLE-SCENE PERFORMANCE

# Best across all four metrics in single-scene training



**+44.7%**

PSNR vs KNN

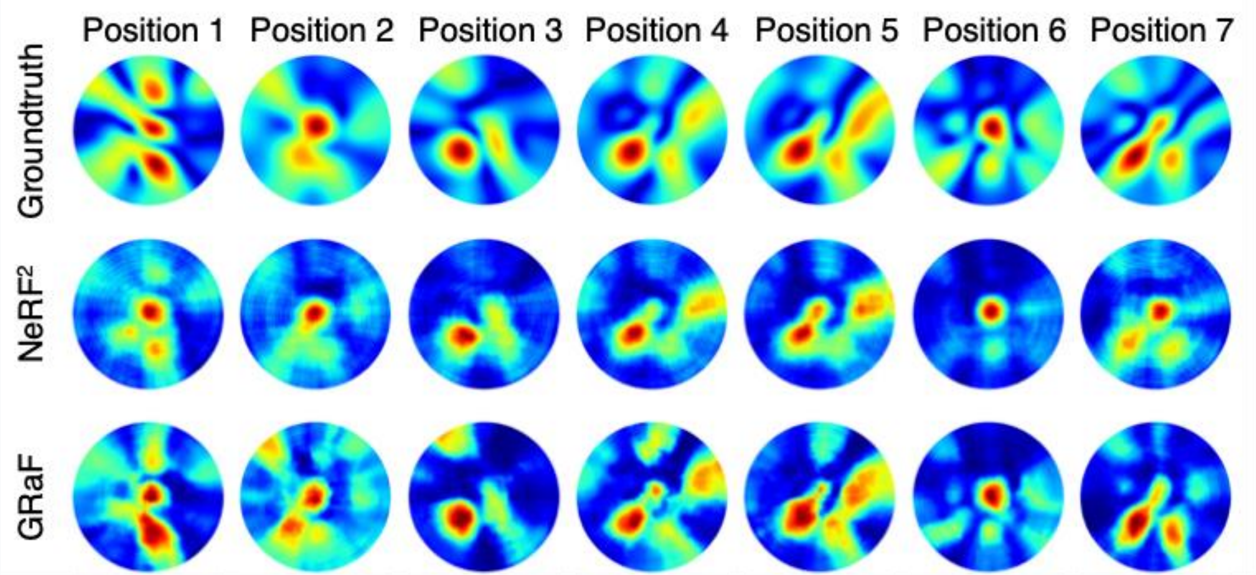
**+10.2%**

PSNR vs NeRF²

**−50.4%**

LPIPS vs NeRF²

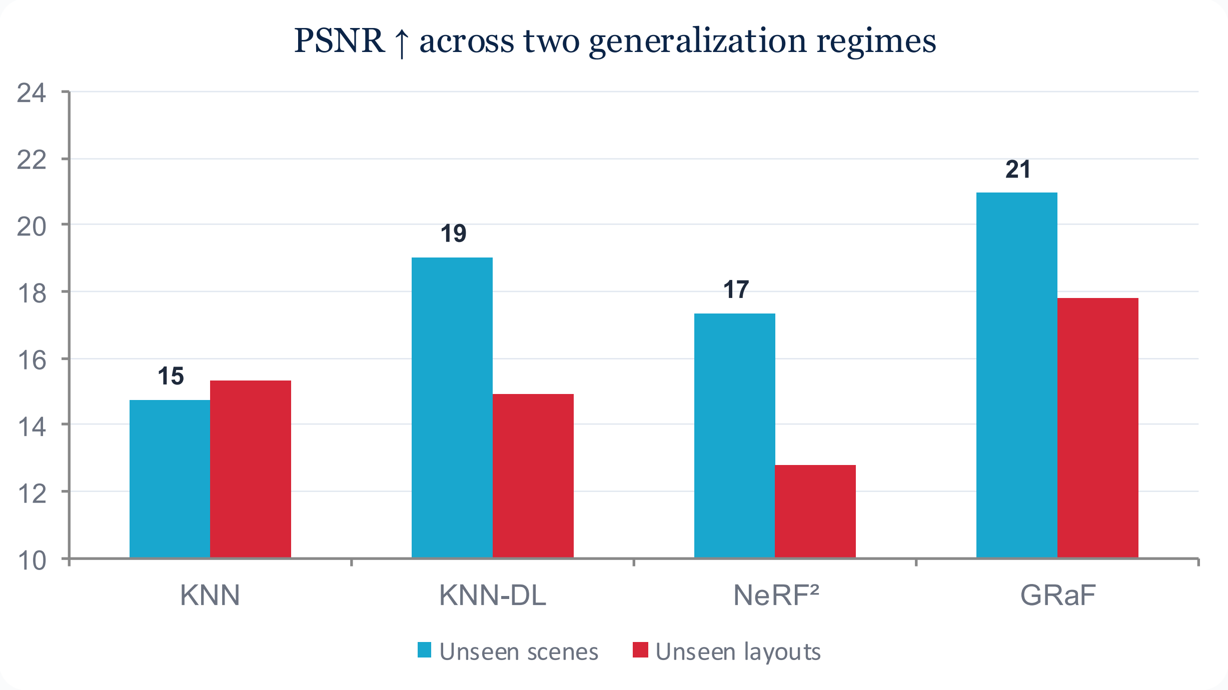
# Spectra synthesized by GRaF visibly track ground truth



*NeRF² blurs directional peaks. GRaF preserves the heatmap structure at seven held-out transmitter positions.*

6 GENERALIZATION

# GRaF holds up where baselines collapse



**UNSEEN SCENES**

PSNR drops:

**−8.5%** KNN-DL
**−12.9%** NeRF²
**Best** **GRaF**

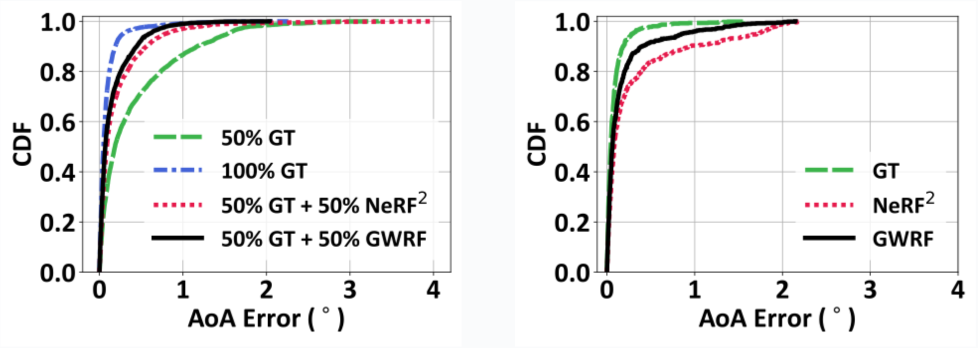
**UNSEEN LAYOUTS**

Bigger drops:

**−28.2%** KNN-DL
**−35.9%** NeRF²
**Robust** **GRaF**

6 DOWNSTREAM TASK · LOCALIZATION

# Synthetic spectra train better AoA estimators



(a) For AANN training

(b) For AANN testing

*AoA-error CDF when training (a) and testing (b) an AANN on synthesized spectra.*

vs 50% GT only

**−61.6%**

*AoA estimation error when GRaF spectra augment training data.*

vs NeRF²-augmented

**−25.8%**

*Lower AoA error than the NeRF²-augmented baseline.*

TAKEAWAYS

# GRaF, in three lines

## ✓ Theory first

An interpolation theorem in the RF domain — error bounded by neighborhood radius squared.

## ✓ Architecture follows

Geometry-aware Transformer learns the interpolation; neural ray tracing aggregates complex contributions.

## ✓ Generalization works

State-of-the-art across single-scene, unseen scenes, unseen layouts, and a downstream AoA task.

Code: `github.com/kangyang73/GRaF`