RaySR: Super Resolution for Radio Maps

Marwin Haddad marwinh@kth.se
Simon Wanna wanna@kth.se
Sam Barati Bakhtiiari sambb@kth.se
Tom Boustedt tombo@kth.se

Abstract

Radio maps are spatial representations of propagation metrics that guide cellular network design, yet computing them at high resolution with ray tracing is computationally expensive. We investigate whether learning based super resolution can reconstruct high resolution radio maps from coarse simulations at reduced cost. Using Sionna city scenes, we built a fully automated data pipeline to generate realistic and consistent low and high resolution map pairs. We trained and evaluated image super resolution models and compared them to bicubic interpolation and non learning RME baselines. The models recover more realistic images with slightly finer structure and sharper coverage transitions than interpolation and yield higher PSNR and SSIM, but are not on pair with the model's performance on natural image super resolution. A practical obstacle was realistic transmitter placement, which we resolved through rooftop snapping and controlled spatial layouts. Code is available at: https://github.com/simonwanna/RaySR.

1. Introduction

Radio maps, also referred to as "coverage maps", depict the spatial distribution of radio propagation metrics, such as received signal power, path loss, or signal-to-interference-plus-noise ratio, over a geographic area. They are central to cellular network planning and optimization, guiding the network engineer's work of placing base stations, and setting the antenna parameters such as power levels, for optimal performance in complex urban environments. High-fidelity radio maps are typically generated either from extensive measurements or from physics-based simulators.

Prominent among these are ray-tracing tools, e.g., NVIDIA's Sionna RT [2], which provide high accuracy by explicitly modelling reflection, diffraction, and scattering. However, electromagnetic propagation in realistic environments cannot usually be solved analytically and must be approximated numerically, and the computational load of such methods increases rapidly with scene complexity and resolution [9]. For example, finer maps need more rays over the same area for a similar coverage density. As a result, full-resolution ray-traced radio maps are often prohibitively expensive to compute when network planners require meter-level resolution over large areas.

An appealing strategy is to decouple fidelity from simulation cost by computing a coarse map and then reconstructing fine-scale detail algorithmically. Super-resolution (SR) addresses this problem: given a low-resolution input, recover a high-resolution output that preserves edges and small structures. Classical SR relies on fixed interpolation schemes that oversmooth fine-scale detail, while learning-based SR leverages data-driven priors to restore sharper and more realistic structures.

Applying SR to radio maps is distinct from natural-image SR. Radio maps represent physical propagation fields, often in logarithmic units, with smooth large-scale structure and sharp discontinuities induced by occlusion and antenna patterns. Multipath effects generate spatial variations that interpolation typically fails to recover. Effective SR must therefore enhance detail while remaining consistent with underlying propagation behaviour. In network planning workflows, this enables a controllable accuracy-cost trade-off: run a fast low-resolution ray tracing simulation and reconstruct a denser map, which must be done with lower computational cost.

In this work, we explore the feasibility of learning-based SR as an alternative for high-resolution radio-map generation. Using pairs of high-resolution maps and their low resolution counterparts, we study how well SR can recover fine-scale structure and boundaries from coarse inputs, and how the computational cost for this alternative compare to other methods.

2. Background

2.1. Radio Map Estimation Methods

Radio maps, which visualize the radio frequency (RF) environment state, are crucial for tasks like enhancing spectrum efficiency and quality of service in wireless networks. In practice, generating detailed radio maps is difficult, and one often resorts to various estimation methods. The core challenge in Radio Map Estimation (RME) is accurately predicting the global RF parameter distribution from sparse measurements. Existing RME methods are categorized into

three major classifications: model-driven, data-driven and hybrid methods.[1]

Model-driven methods utilize explicit propagation models to estimate the radio map, eliminating the need for large training datasets but requiring substantial domain expertise. Representative approaches include ray-tracing and path-loss prediction, as well as non-parametric kernel-based reconstruction techniques such as adaptive RKHS methods [4], which could be considered data-driven, but are classified by Feng et al. [1] as model-driven because they approximate the physical channel gain function within a structured superposition model. In contrast, data-driven methods rely on measurement data and employ machine learning models to predict unobserved signal strengths. These include classical interpolation methods and deep learning approaches such as CNN-based RME [3] and UNet-based radio map prediction [5]. Finally, hybrid methods integrate physical propagation priors with data-driven learning, achieving higher accuracy in sparse sampling regimes by exploiting domain knowledge, as demonstrated by physics-aided or model-assisted spectrum cartography frameworks [7].

The RME problem is analogous to the computer vision task of Image Super-Resolution (SR), as both are ill-posed spatial reconstruction problems aiming to infer a complete, high-resolution representation from sparse or low-resolution input data which is why we hypothesise that a SR-centric approach could be suitable.

2.2. Methods in Image Super-Resolution

Image super-resolution deals with converting a low resolution image to one of its high resolution counterparts. While ill-posed, the challenge of upsampling an image has been of great importance to many fields such as computer vision and medicine [8]. Before deep learning, super-resolution mainly employed interpolation algorithms. However, these methods come with problems such as noise amplification, blurring and computational complexity [8]. To remedy the aforementioned issues that follow traditional super-resolution techniques, researchers sought to combine them with deep learning methods.

Deep learning based image super-resolution can generally be divided into several different architectures based on where and how the upsampling occurs. For example, one of the earlier models, SRCNN, used a "pre-upsampling" strategy in which the low resolution images were first enlarged with traditional interpolation methods before the CNN refined the image [8]. This approach reduced the difficulty of learning since the most demanding part of the upsampling is performed by the interpolation algorithm. Whilst popular, it too came with the associated issues that plagued traditional interpolation, such as blurring, noise, and computational complexity since most operations are performed in high dimensional spaces. To deal with this, post-upsampling super-resolution was introduced. Here, most computations were moved into low dimensional space by replacing the upsampling algorithms with learnable layers in the final stage of the network. Learnable upsampling operations, such as transposed convolutions or sub-pixel convolution layers, replaced fixed interpolation methods and enabled the network to learn end-to-end mappings from low- to high-resolution representations. These models demonstrated superior reconstruction quality and efficiency [8].

To further improve upon the performance, methods such as progressive upsampling and iterative up-and-down sampling have been introduced. The former of the two aimed to gradually refine and reconstruct the high-resolution image through multiple stages, rather than doing it in one step like in the previous frameworks. By decomposing the upsampling, the learning difficulty is reduced and increases performance [8]. Iterative up-and-down sampling super-resolution is another method in which models such as DPBN iteratively refine high-resolution predictions by iterating between upsampling and downsampling operations, feeding the reconstruction errors back into the model [8].

Building on the architectures above, the PAN (Pixel Attention Network) proposed by Zhao et al. introduces a pixel-wise attention mechanism [10]. Unlike previous methods that focus solely on channel attention, PAN generates 3D attention maps to emphasize informative high-frequency details. This structural design improves feature representation, yielding higher fidelity in image reconstruction.

Similarly, ESRT (Efficient Super-Resolution Transformer) combines CNNs with transformers to capture local textures and maintain global context, whilst also ensuring computational efficiency. This enables the network to keep track of important regions and capture repetitive patterns and structures that standard CNNs may miss [6].

In this work, we adopt PAN and ESRT as our main learnable super-resolution backbones and re-implement them for radio-map super-resolution within a common PyTorch Lightning framework (see Section 4.1).

3. Problem description

High-fidelity radio maps are indispensable for network planning, but generating them at meter-level resolution over large urban areas is computationally demanding. Physics-based approaches such as ray tracing approximate Maxwell's equations in complex 3D scenes and must resolve multiple reflections, diffractions, and transmissions per transmitter–receiver pair. The runtime scales with the number of receivers (grid points), candidate rays per point, scene complexity (triangles, materials), and allowed interaction depth; memory and compute demands are also high [9]. Full-wave CEM methods (e.g., FDTD, FEM, MoM) are even more expensive and generally impractical

for city-scale planning [9]. In practice, planners must evaluate many "what-if" configurations (site, tilt, power, carrier) across large regions, making repeated high-resolution simulations prohibitively costly. Parallel efforts in RME highlight a broader need to reconstruct dense, accurate fields from limited or coarse data, via model-driven, data-driven, or hybrid methods [1].

We study a cost-accuracy trade-off that replaces a single expensive high-resolution simulation with a two-stage pipeline: (i) compute a coarse map at low resolution or with reduced simulation fidelity, and (ii) reconstruct a high-resolution map algorithmically. Concretely, let Φ_{HR} denote the target high-resolution radio map over a region Ω at grid spacing Δ, and let Φ_{LR} be a low-resolution or coarsened version at spacing $r\Delta$ with $r \geq 2$, produced quickly by a ray tracer or derived from sparse samples. The objective is to obtain $\hat{\Phi}_{HR}$ such that the total wall-clock time $T(\Phi_{LR}) + T(SR)$ is significantly smaller than $T(\Phi_{HR})$ while maintaining accuracy adequate for planning tasks.

We cast the reconstruction as supervised super-resolution. Given training pairs $\{(\Phi_{LR,i}, \Phi_{HR,i})\}$, learn a mapping

$$\hat{\Phi}_{HR} = f_\theta(\Phi_{LR}),$$

optionally conditioned on auxiliary context (e.g., transmitter metadata, frequency, basic geometric layers). The loss initially follows the cited papers and are based on pixel-wise fidelity in dB (MAE/RMSE), but also explores structural similarity (SSIM), and task-aligned metrics such as height at building occlusions. Because radio maps represent physical fields, outputs must remain plausible: preserve large-scale path-loss trends, sharpen occlusion boundaries without hallucinating non-physical artifacts, and maintain consistent dynamic range.

Evaluation compares (a) full-resolution ray tracing, (b) classical upsampling (bicubic), and (c) the proposed SR. We report accuracy versus cost, investigation end-to-end speed-ups

$$S = \frac{T(\Phi_{HR})}{T(\Phi_{LR}) + T(SR)},$$

The problem is thus to achieve near–high-resolution fidelity for planning-relevant metrics at less computational cost, leveraging learning-based SR to bridge the gap between fast coarse simulation and expensive fine-grained propagation modelling.

4. Methodology

In the following, we describe RaySR (pronounced "racer"), our Lightning-based data generation and super-resolution pipeline for producing high-resolution radio maps from coarse simulations.

4.0.1. Scenes and radio-map solver

We utilise Sionna's built in 3D city scenes (buildings, ground, materials) and load them into the Sionna ray-tracing engine. For a given centre point and scene coverage size, we use Sionna's *RadioMapSolver* to compute planar radio maps for a chosen metric (path gain, RSS, or SINR). All samples share the same simulation backend and interaction depth so that differences come from geometry and transmitter layout, not from changing physics.

The transmitter placement is a integral part of our solution. Since we generate thousands of images which will not all be examined, it is important that the transmitters are not only placed realistically, but also generates a divers distribution of data for the machine learning model to be trained on. Sionna have no built in method to do this, neither does it have any method to recognise buildings which transmitters should be placed upon.

We separate transmitter placement from map computation: a `SceneTransmitterBuilder` is responsible for placing base stations in the scene, while a `RadioMapDataGenerator` calls the solver and packages the data.

4.0.2. Height map and rooftop snapping

Before placing any transmitters, the generator scans the scene geometry and builds a coarse height map over the whole city at a fixed step size (a regular grid in (x, y) with a height value per cell). Only objects above a minimum height are considered "valid" (e.g., buildings, not the ground plane). For each grid cell we store the maximum object height and pre-compute the index of the nearest valid neighbour for every invalid cell. When the builder proposes a transmitter position, we convert it to height-map indices and "snap" it to the nearest valid cell, seen in Fig. 1, where the boxes are building objects, coloured by height. The transmitter is then moved horizontally to that cell and placed on the roof with a small vertical margin. This snapping step ensures that transmitters that would otherwise land inside walls or in empty space are moved onto the roof of the nearest building, which makes the layouts more realistic without hand-coding city-specific rules.

4.0.3. Transmitter placement: grid and sliding window

After the height-map computations we, for each sample:

- Choose a square coverage region of fixed size (e.g., $(L \times L)$ meters) and randomize its centre inside the scene. Intuitively, this is a square "sliding" over the city from sample to sample.
- Inside this square, lay out (n_{tx}) transmitters on an approximately regular 2D grid. Each transmitter is then randomly perturbed within its grid cell.

This constitutes one valid scene sample which is sent to the next phase. This process gives two sources of randomness:

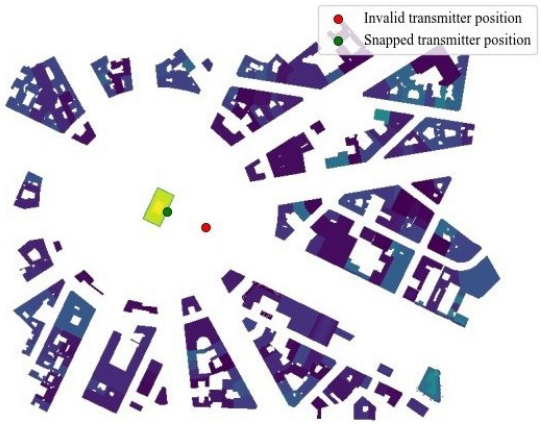

Figure 1. Demonstration of snapping mechanism for transmitters

(i) where the square lands in the city and (ii) where each transmitter ends up inside its cell, which increases diversity of layouts for the same scene.

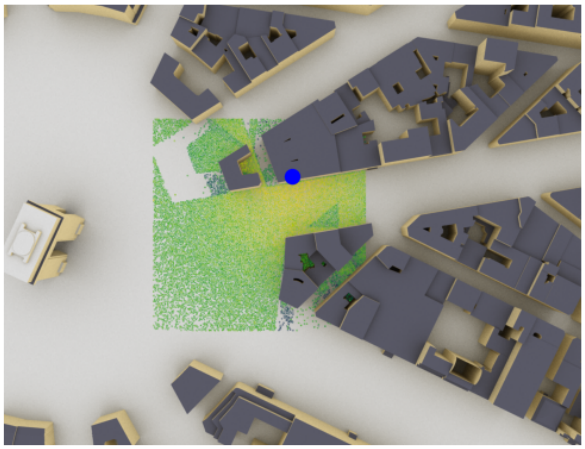

Figure 2. Data sample image overlaid on the Etoile scene

4.0.4. Low- and high-resolution radio maps

Given the final transmitter positions and the chosen coverage square, we query the radio-map solver twice: once at a coarse resolution (cell size (Δ_{LR})) and once at a finer resolution (cell size ($\Delta_{\text{HR}} = \Delta_{\text{LR}}/r$)). Note: there are no downsampling, rather a real low resolution creation since this most closely imitates a real scenario where you wouldn't have the high resolution images.

Both maps cover the same physical area and share the same centre, so the high-resolution map is a refined version of the coarse map rather than a different view. For scenes with multiple transmitters, we collapse the per-transmitter maps to a single "strongest-signal" map by taking the maximum over the transmitter dimension, mimicking typical coverage planning maps. All values are converted to dB (or dBm for RSS) and clipped to a floor value to avoid numerical issues in low-signal regions.

4.0.5. Dataset construction and normalization

Each sample is stored as a small file containing the low-resolution map, the high-resolution map, the transmitter positions, the scale factor (r), and metadata. A Lightning `DataModule` lazily loads these files, splits them into training and validation sets, and normalizes the maps to ([0,1]) using a fixed dB floor/ceiling so that the model always sees comparable dynamic ranges. This setup lets us swap scenes or change the scale factor purely through configuration, without touching the training code.

We also use a relatively small dataset size of 3200 images, 800 from each city, and out of these, 90% were used for training and 10% for validation. This is mainly since we did not see much improvement from increasing the dataset size further (see discussions 6). Interestingly, the SR models was not trained on large datasets themselves, only 2000 images [10].

4.1. Super-resolution models and training

As super-resolution backbone we look for well cited papers with available open source and clean code repository.

4.1.1. Pixel Attention Network architecture

Our PAN implementation follows the official reference repository by Zhao et al. [10] and keeps the original network structure (convolutional head, pixel-attention residual blocks and upsampling tail). We make the following adaptations for the radio-map setting:

- We modify the input and output layers to, by default, operate on single-channel maps in dB instead of three-channel RGB images, while preserving the original number of feature layers and blocks.
- We make it optional to use a second feature channel based on our height map, since our initial experiments showed signal hallucinations in building cites, and we therefore hypothesise this might help prevent such noise.
- We remove image-specific preprocessing and data augmentations from the original code (e.g., RGB normalization) and instead use the fixed dB normalization defined in Section 4.0.5.
- We wrap the model in a PyTorch LightningModule that exposes a common forward(Φ_{LR}) interface and handles loss computation, logging and optimizer configuration, so PAN can be trained with the same loop as other implemented models.

Apart from these adaptations for modality and framework, the PAN architecture closely follows the official implementation to keep our results comparable to the paper [10].

4.1.2. Efficient Transformer for Single Image Super-Resolution

ESRT is implemented analogously, starting from the official PyTorch code by Lu et al. [6]. After observing that PAN sometimes struggled with capturing sharp discontinuities, we introduced ESRT as a complementary, transformer-based backbone to better model the structures in the radio maps. We retain the original lightweight hybrid design (shallow CNN head, transformer blocks in low-resolution feature space, and learnable upsampling), and adapt it to our setting as follows:

- The network is configured for single-channel inputs and outputs, matching our radio-map tensors, with the same upscaling factor r as in the data pipeline.
- As for PAN, dataset-specific preprocessing from the reference code is disabled, and we reuse the common normalization and batching provided by our Lightning DataModule.
- ESRT is also wrapped as a LightningModule and integrated to our pipeline as PAN, enabling a better comparison between the two backbones under identical training conditions.

In summary, both PAN and ESRT are implemented as close to their official repositories as possible, with only small, necessary changes such as removing RGB channels and integration into our Lightning-based training pipeline.

4.2. Training objective and metrics

During training and validation, we use a smooth variant of the L1 loss. This choice follows the loss functions used in the reference implementations of our baseline models [10] [6]. We also verified experimentally that alternative losses (including L2, and perceptual variants) did not provide improvements for our radio-map super-resolution case.

Performance evaluation focuses on peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) as well as root mean squared error (RMSE), which despite its similarity to PSNR in reflecting absolute reconstruction error in dB space, primarily is included for its intuitive interpretability. All metrics are computed between the predicted and ground-truth high-resolution maps on a per-sample basis. SSIM captures structural fidelity, which is important for preserving large-scale path-loss patterns and sharp coverage transitions. We additionally compare all models against a bilinear-interpolation baseline to contextualize the achievable gains. The quantitative results are presented in Section 5.

In addition to numerical metrics, we also perform qualitative assessment. For each experiment, we visually inspect the generated high-resolution maps to verify that the reconstructions are physically plausible and do not introduce artefacts that would artificially inflate PSNR or SSIM. This qualitative step ensures that the models do not exploit metric-specific biases, for example, by oversmoothing or attenuating regions in a way that improves the score but degrades the usefulness of the resulting radio maps. It is also important to identify hallucinations, where the model might induce a signal in places were buildings are located.

Finally, the feasibility of using super resolution as a technique is partially based on inference performance. Computing a higher resolution image with similar coverage requires changes in the simulator. Simplest way is to emit more rays such that they hit the new pixels not present in the lower resolution simulation. Hence, computing the low resolution image and then upsampling it with Super Resolution must therefore be done quicker than simply generating the full high resolution image with ray tracing. Because of this, we compare Sionna's solver runtime given different resolution but with constant coverage, which requires using different number of emitted rays, to see if there is indeed a speed up from the proposed method.

5. Evaluation and results

We evaluate the super-resolution models using PSNR, RMSE and SSIM on the radio-map test set and compare them against a bicubic baseline. The focus is to assess how much structure the models recover relative to simple interpolation and to quantify the performance gap to the values reported in the original PAN and ESRT papers. Table 1 summarises the 2× results on our dataset.

Table 1. Model performance on the radio-map test set with 2x scaling factor.

Method	PSNR ↑	SSIM ↑	RMSE ↓
Bicubic	21.97	0.289	24.21
PAN	24.21	0.377	18.69
ESRT	14.25	0.5416	58.83

For completeness, we also evaluate the models at 2× scaling factors and compare the results with the values reported in the original PAN paper. Although the datasets differ, this comparison provides a coarse indication of how much performance is lost when moving from natural-image benchmarks to our radio-map setting. The results in Table 2 below therefore serve to contextualize our results rather than to draw direct conclusions across datasets.

Table 2. Comparison of SR results across datasets

Method	PSNR	SSIM
PAN (Urban100 dataset 2x)	32.01	0.9273
PAN (Set14 dataset 2x)	33.59	0.9181
PAN (our dataset 2x)	24.21	0.377

We also compare Sionna's ray-tracing runtime against model inference speed to quantify potential cost savings.

Table 3 verifies that the cost in speed of increasing number of rays (here 9M compared to 3M), to preserve coverage when increasing resolution, exceeds that of using SR ($S \gg 1$). The SR inference time for SR on a GPU was negligible, hence no comparison of 3x and 4x.

Table 3. Sionna HR simulation time vs. LR + SR inference time

Method	Resolution	Runtime
HR Solver	1.5m	0.5171
LR Solver + SR (2x)	3m	0.0443s

In addition to the quantitative metrics, we inspect the reconstructed maps visually to assess whether the models preserve physically relevant structures. A representative example is shown in Fig. 3.

Figure 3. Top row: LR, HR. Bottom row: Bicubic, PAN. (2x)

Comparing the low resolution (LR) to the high resolution (HR) on the top row we notice that the LR have some noise artefacts not present in the HR image. Purple indicates zero signal and are usually from building sites. The bicubic interpolation method (bottom left) hallucinates even more of these noisy pixels, which represent a signal in pixels which should have zero signal strength. The PAN model however (bottom right), is to a great extent capable of keeping the non-signal areas as they are.

Nevertheless, we can also see that in the high resolution image that there are some areas where around every other pixel is hit by a ray, and it therefore oscillates between purple and green/blue colours. While the model is doing a better job than simple interpolation, this pattern is not fully reconstructed.

5.0.1. Height map ablations

The height map shows the height of the buildings which are non-zero in places where there should not be a signal. Introducing the height map feature channel increased the performance of PAN as seen in table 4.

Table 4. Model performance with and without height map feature channel

Method	PSNR ↑	SSIM ↑	RMSE ↓
PAN (without)	23.44	0.264	20.18
PAN (with)	24.21	0.377	18.69

6. Summary and discussion

The objective of this work is to evaluate whether learning-based super-resolution (SR) can act as a viable substitute for HR radio maps. Accordingly, we designed a complete data-generation pipeline in Sionna, including automated transmitter placement using a height-map snapping procedure, and the creation of paired LR and HR radio maps for supervised training. Using these data, we trained two SR models, PAN and ESRT, to reconstruct HR images from the courser LR inputs, and evaluated the models using PSNR, SSIM, and visual inspection, with bicubic interpolation as baseline. The study aims to assess the achievable fidelity of the image-reconstruction and the potential computational savings relative to running high-resolution ray tracing.

6.1. Interpretation of findings

The experimental results indicate that learning-based SR can recover more structure from LR radio maps than bicubic interpolation. The PAN model consistently improves all metrics over the bicubic baseline, showing that the model is able to reconstruct and preserve large-scale propagation trends more effectively. We can especially note that the LR and the bicubic interpolation hallucinates signals in places where no signal is present. Looking at the HR image in the top right image of figure 3, we clearly see the blue introduced noise.

6.1.1. Ablation studies

Our ablation study shows that the introduction of a second height channel improves the result significantly. The PAN model was introducing noise similar to the Bicubic interpolation prior to adding the extra feature. However, when the feature is introduced the model makes a much better job in those low signal areas. This intuitively makes sense since the height map introduces a signal where the low signal are usually in buildings or in a shadow caused by a building.

6.1.2. Model performance

ESRT does achieve higher SSIM but is far off in the PSNR and RMSE. The model was expected to recover global structure better with its transformer based backbone but struggled a lot with mean collapsing, i.e. the model regressing its predictions towards some average value and loosing expressivity. The repository implementation was not as straight forward as the one from PAN and might have needed other modifications to work as well as expected. This however, highlight the importance of our qualitative analysis since the high SSIM score might indicate better results than the model achieved. Only from inspecting the reconstructed images visually, one could see that the importance of a high PSNR is critical to the image quality and can not be compensated for by a high SSIM.

6.1.3. Data considerations

Although the results indicate that PAN performed relatively well compared to the interpolation baseline, the achieved scores remain below those reported in the work of Zhao et al. on standard super-resolution benchmark datasets shown in table 2.

This indicates that the degradation might not be model-specific but intrinsic to the propagation domain. Radio maps exhibit smooth path-loss gradients combined with sharp occlusion-induced discontinuities, which might make the SR task more challenging than reconstructing textured natural images. In addition, because the data is generated from top-down projections with limited scene diversity, many samples share similar building layouts and transmitter configurations, reducing variation in the training set. Consequently, the models may not fully capture the range of propagation patterns present in more diverse environments.

6.1.4. Computational considerations

The results demonstrate a clear speed advantage when combining low-resolution simulation with super-resolution (LR+SR) compared to native high-resolution (HR) ray tracing. As shown in Table 3, the HR solver requires around half a second, while the LR+SR pipeline takes only considerably less in this configuration.

It must be noted that this is not a perfectly fair comparison: true HR ray tracing requires a higher density of rays to maintain consistent coverage across more pixels. If one merely increases resolution without more rays, coverage gaps occur while computational cost might be very similar. The LR+SR approach bridges this gap by computationally interpolating structure, providing superior coverage at a fraction of the GPU cost. While overheads in data transfer and scene snapping exist and made our comparison a bit difficult, the inference time remains negligible.

Nevertheless, with more transmitters and more complex scenes, the HR computations is expected to increase while the SR should be constant. These findings suggest a substantial speedup, though real-world performance may vary based on specific testing environments and system overheads. If the SR accuracy is retained is left for future studies.

6.2. Practical Implications

The results suggest that learning-based SR represents a promising, albeit preliminary, candidate for accelerating radio map workflows. By decoupling the ray-tracing grid from the final visualization density, there is potential for engineers to evaluate a higher volume of transmitter configurations. While our current implementation is limited to simplified setups, this approach hints at a scalable path for modeling 5G/6G deployments without the exponential increase in hardware resources typically required by high-resolution physics-based simulations.

Qualitative evaluation indicates that SR-reconstructed maps preserve large-scale path-loss trends and shadowing effects better than classical interpolation, though they remain distinct from high-resolution ground truths. This suggests that while SR is not yet suitable for high-precision final validation, it could eventually serve early-stage ”what-if” planning phases where rapid iteration and relative coverage trends are prioritized over absolute meter-level accuracy.

The practicality of this method currently remains dependent on model robustness. Since our models were trained on a limited set of synthetic urban scenes, their performance in real-world environments—characterized by diverse architectural styles and multi-transmitter interference—remains unproven. Nevertheless, the findings support the hypothesis that SR is a feasible direction for bridging the gap between coarse, low-cost simulations and high-density propagation fields, warranting further investigation into its reliability in complex propagation environments.

6.3. Limitations and Future Work

While the efficiency gains are evident, several constraints limit the direct applicability of our current model to universal urban environments:

- **Scenario Complexity:** Our training used simplified isotropic transmitter configurations. Real-world complexities, such as frequency-dependent diffraction, beamforming patterns, and diverse frequency bands, may introduce propagation structures that our experiments did not address, leaving their learnability an open question.
- **Transmitter Density:** Our experiments focused on limited transmitter counts. In realistic multi-cell deployments, signal interference and cell-edge discontinuities become more complex. Future studies should verify if the SR accuracy is maintained when predicting these high-interference regions.

- **Environmental Generalization:** The limited scene diversity in our synthetic dataset may lead to performance degradation in cities with significantly different architectural styles, such as highly industrial zones or historical centres with irregular building layouts.
- **Input Feature Richness:** While the inclusion of a second channel (height map) yielded significant gains, relying on single-metric path-loss maps remains limited. Incorporating richer auxiliary data, such as for example binary line-of-sight (LoS) indicators, represents a critical next step to resolve discontinuities and shadow boundaries more accurately.

Future work should focus on scaling the transmitter count and complexity to determine the point where the fidelity of SR reconstructions might diverge from ground truth propagation, particularly in complex ”deep-shadow” urban areas.

References

[1] Bin Feng, Meng Zheng, Wei Liang, and Lei Zhang. A recent survey on radio map estimation methods for wireless networks. *Electronics*, 14(8), 2025. 2, 3

[2] Jakob Hoydis, Sebastian Cammerer, Fayçal Ait Aoudia, Merlin Nimier-David, Lorenzo Maggi, Guillermo Marcus, Avinash Vem, and Alexander Keller. Sionna, 2022. https://nvlabs.github.io/sionna/. 1

[3] T. Imai, K. Kitao, and M. Inomata. Radio propagation prediction model using convolutional neural networks by deep learning. In *2019 13th European Conference on Antennas and Propagation (EuCAP)*, pages 1–5, 2019. 2

[4] Martin Kasparick, Renato L. G. Cavalcante, Stefan Valentin, Sławomir Stańczak, and Masahiro Yukawa. Kernel-based adaptive online reconstruction of coverage maps with side information. *IEEE Transactions on Vehicular Technology*, 65(7):5461–5473, 2016. 2

[5] Ron Levie, Çağkan Yapar, Gitta Kutyniok, and Giuseppe Caire. Radiounet: Fast radio map estimation with convolutional neural networks. *IEEE Transactions on Wireless Communications*, 20(6):4001–4015, 2021. 2

[6] Zhisheng Lu, Juncheng Li, Hong Liu, Chaoyan Huang, Linlin Zhang, and Tieyong Zeng. Transformer for single image super-resolution, 2022. 2, 5

[7] Daniel Romero and Seung-Jun Kim. Radio map estimation: A data-driven approach to spectrum cartography. *IEEE Signal Processing Magazine*, 39(6):53–72, 2022. 2

[8] Zhihao Wang, Jian Chen, and Steven C. H. Hoi. Deep learning for image super-resolution: A survey, 2020. 2

[9] Zhengqing Yun and Magdy F. Iskander. Ray tracing for radio propagation modeling: Principles and applications. *IEEE Access*, 3:1089–1100, 2015. 1, 2, 3

[10] Hengyuan Zhao, Xiangtao Kong, Jingwen He, Yu Qiao, and Chao Dong. Efficient image super-resolution using pixel attention, 2020. 2, 4, 5