

Received 25 October 2022, accepted 12 November 2022, date of publication 17 November 2022,
date of current version 23 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3223097

RESEARCH ARTICLE

A Deep Learning Network Planner: Propagation Modeling Using Real-World Measurements and a 3D City Model

LUKAS ELLER¹, (Graduate Student Member, IEEE),

PHILIPP SVOBODA¹, (Senior Member, IEEE), AND MARKUS RUPP¹, (Fellow, IEEE)

Institute of Telecommunications, TU Wien, 1040 Vienna, Austria

Corresponding author: Lukas Eller (lukas.eller@tuwien.ac.at)

This work was supported in part by the Institute of Telecommunications, TU Wien; and in part by the Open Access Funding Programme, TU Wien Bibliothek.

ABSTRACT In urban scenarios, network planning requires awareness of the notoriously complex propagation environment by accounting for blocking, diffraction, and reflection on buildings. To this end, deep learning-based signal-strength prediction directly operating on environmental data has recently gained attention, mainly as a computationally efficient alternative to ray-tracing. Our work combines RSRP measurements from an extensive drive-test campaign in a live 4G network with a 3D city model for the largest real-world assessment of such data-driven schemes to date. We compare three different encodings of the propagation environment and find that a neural network operating on a full 3D representation of the surroundings performs best with an RMSE of 7.06 dB. It is followed by a model using only the direct path profile with 7.78 dB and a reference neural network utilizing a binary line of sight indicator achieving 8.76 dB. The large size of our data set allows us to address several open questions regarding the inner workings of these black box approaches. In particular, we elaborate on different evaluation strategies, highlighting the importance of spatial separation of train and test areas, as the rich environmental data implicitly provides a spatial reference. Through model explainability, we further identify the area along the direct path between the user equipment and the transmitter as the input region with the highest feature importance — questioning the common practice of including large buffer areas. Evaluating the models in scenarios with artificially placed base stations reveals that the measurement campaign offers a sufficient basis for a prototypical network planner. The trained models, which we make publicly available, exhibit the dominant propagation mechanisms in urban areas and generate spatially consistent and physically sound signal-strength maps.

INDEX TERMS 5G, 6G, cellular network planning, deep learning, drive-test, LTE, machine learning, model explainability, pathloss prediction, propagation modeling, signal-strength maps.

I. INTRODUCTION

The efficient planning of cellular networks, seen as a key stepping stone toward greener networks [1], requires accurate and reliable propagation modeling. For this task, *mobile network operators* (MNOs) have long relied on empirical models based on extensive measurement campaigns [2], which only demand minimal computational resources. By design, these

models can only partially account for the particular geometry of an environment, but use rather high-level features such as the average building height or street width in the considered area [3]. Especially in urban regions, with their notoriously complex propagation environments, ray-tracing can be an alternative that natively accounts for blocking, diffraction and reflection effects induced by the urban geometry [4]. However, ray-tracing comes with a high computational cost, which can be an unbearable constraint when considering the highly heterogeneous and possibly non-static network layouts

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul S. Raval¹.

expected for 6G — specifically, the ongoing trend towards *unmanned aerial vehicle* (UAV) communications [5], [6]. To this end, there has been increased interest in using deep learning methods for pathloss prediction, moving the main computational burden to training and thus promising efficient operation during inference [7]. While the main focus has been on approximating ray-tracers, [8], models trained on measurement data have outperformed them in some real-world scenarios [9]. Considering the tendency toward predictive maintenance and digital network twins [10], we can envision such data-driven propagation schemes deployed in an online fashion, together with machine learning based network optimization [11], [12]. Clearly, classical offline system-level simulations can also benefit from more efficient, data-driven propagation modeling derived from real-world measurement campaigns [13]. Even though this area has seen extensive research [14], we still identify several open questions preventing the use in real-world applications. In particular, the black box nature of such approaches makes it hard to assess the propagation mechanisms learned from the measurements. Partially, this is due to the majority of existing studies being based on ray-tracing data, where it is straightforward to provide an unbiased and extensive training set covering all relevant scenarios [4]. In contrast, existing real-world assessments are often limited to relatively small data sets [15], consisting of only a few *base stations* (BSs) [9], [16]. Similarly, model explainability [17] has, to the author’s best knowledge, not been applied to the environmental data acting as the input, such that the role it plays in the prediction is still unclear. Due to the rich spatial information provided, we also see a high risk of overfitting on small data sets — thus requiring consistent model evaluation procedures to properly assess the generalization to unseen areas [7]. In our work, we aim to address these open questions. In particular, we:

i) conduct the largest evaluation of deep learning-based signal-strength prediction schemes to date, using $\approx 630\,000$ *reference signal received power* (RSRP) measurements from a live 4G network collected in an extensive drive-test campaign in Vienna, Austria — see Fig. 1.

ii) Moreover, we utilize a high-resolution 3D model consisting of the building outlines together with their respective height, further enriched with terrain elevation data. We deploy *convolutional neural networks* (CNNs) to process these data together with the raw measurements.

iii) Unlike existing work, we put our main emphasis on the generalization capabilities by examining different model evaluation procedures. To our best knowledge, we are the first to apply model explainability methods to identify the most important input regions.

Finally, we assess the learned propagation mechanisms by generating high-resolution signal-strength maps in realistic network planning scenarios. These scenarios, together with the trained model instances, are publicly available.¹

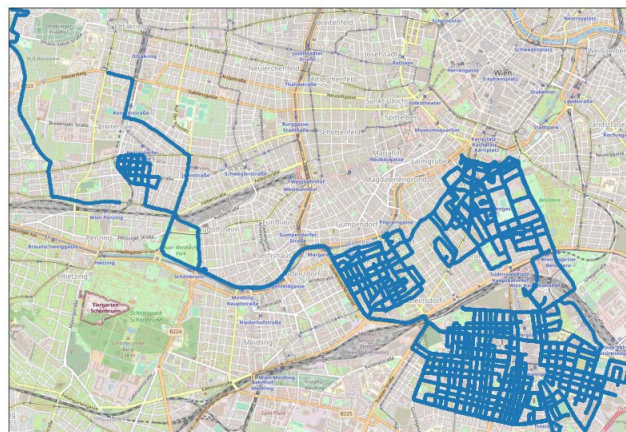


FIGURE 1. Route of the drive-test campaign covering 95 km of dense, inner-city streets of Vienna, Austria.

The remainder of the paper is organized as follows. After the related work in Sec. II, we detail the measurement data and city model in Sec. III — also introducing the empirical 3GPP *urban macro* (UMa) baseline. Then, Sec. IV elaborates on the deep learning formulation before the performance metrics are presented in Sec. V. We close with the model explainability results and comprehensive network planning scenarios in Sec. VI. Final remarks are drawn in Sec. VII.

II. RELATED WORK

In general, we identify two broad categories for data-driven signal-strength prediction: *i)* geospatial interpolation and *ii)* supervised machine learning. For interpolation, sparse measurement locations act as anchor points to construct dense signal-strength maps in the immediate surroundings [18], [19], [20], [21]. While well suited for performance estimates in existing networks [19], these approaches are of limited use for network planning, as adaptations can only be studied after deployment. Moreover, they can, with rare exceptions [21], not natively account for the propagation environment. For the second broad category of supervised machine learning [22], we also find approaches focusing on the generation of local performance maps [23], where the inclusion of absolute coordinates or cell-identifiers omits any cross-area generalization. Other supervised schemes limiting themselves to area-independent features, such as the distance between *user equipment* (UE) and BS, can, in principle, enable generalization to unseen areas and different network deployments [24], [25], [26]. However, in many cases, the error reductions can mainly be attributed to fitting to the distribution of a specific measurement campaign rather than improving upon classical methods on a larger scale [3], [27].

Overall, we thus see the real promise of machine learning methods in the native inclusion of environmental features, which are often unfeasible to process with classical approaches. Starting from relatively high-level input parameters such as average building height or land-use types in [28], [29], research has moved on to extensive feature

¹https://squad.nt.tuwien.ac.at/gitlab/l/eller/ieee_access_deep_learning_network_planner

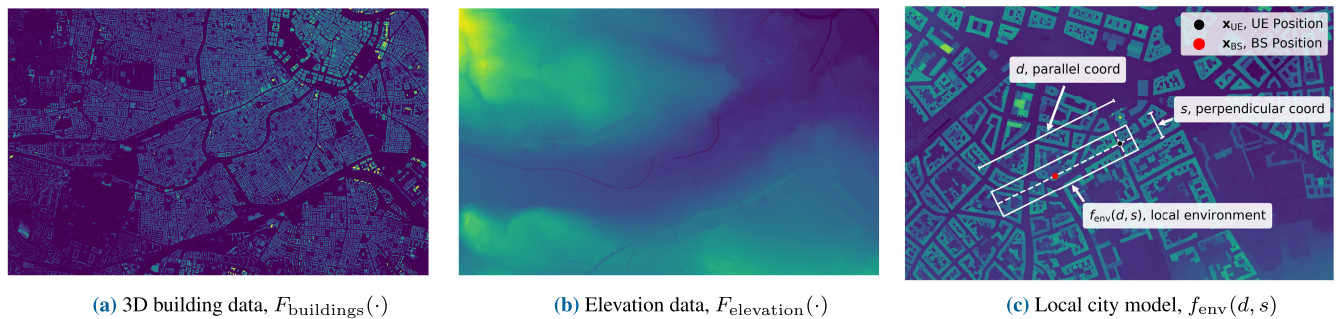


FIGURE 2. We combine building and terrain elevation data to obtain a local representation of the 3D city model.

engineering describing the path from BS to UE in great detail in the simulations in [30] and [31]. Meanwhile, deep learning methods can completely omit the abstractions introduced by the feature generation process by directly operating on raw environmental data. As such, CNNs have been shown capable of extracting pathloss distributions and their respective exponents from satellite images in ray-tracing scenarios [32], [33]. In this context also U-Nets have gained traction, utilizing two or three-dimensional blockage maps to generate a dense pixel-like output of the signal-strength in the area of interest [8], [34]. Instead of describing the surroundings through high-level features, the environment input is often even directly enriched with suitable system parameters to abstain from any abstractions [35]. While we are unaware of any measurement-based evaluation of U-Nets, most empirical validations follow a UE centered environment encoding, processing one measurement at a time. In [9], such a model using squared satellite images centered around the UE location significantly outperformed the ray-tracing baseline. Similar results were achieved with two-dimensional building outlines in [36], or for the mmWave drive-test evaluation in [15] using environmental data from Google Maps. An extension to three dimensions can be achieved by directly incorporating the building height or elevation into the CNN input channel [37]. In this context, also the fusion of different input modalities has been studied extensively [38]. The most prominent aspect here is the careful encoding of the direct path to relate it to the height of the 3D blockages [16], [39]. In [39], the authors propose the concept of the Fresnel height, which is further extended through side-view encodings of the propagation path in [16].

Even though an extensive body of research exists, we miss extensive empirical validation on measurement data — which is especially rare for 3D inputs. To our knowledge, the extent of the drive-test campaign used in this work even makes it the largest real-world evaluation to date. Unlike other studies, we do not solely focus on achievable error reductions but use the large size of our data set to target a better understanding of the propagation mechanisms facilitated by such data-driven approaches. Hereby, we purposefully limit ourselves to CNNs representative of the current state of the art [9], [16] and do not cover recently proposed transformer

architectures [40], [41]. Likewise, we also abstain from feature engineering, which has already been extensively studied in [30] and [31], but focus on the direct processing of raw environmental data through deep learning methods. Here, the role the rich environmental inputs play for prediction is still unclear, compared to the well-known attributions for high-level features in [30] and [31]. In addition to the first application of model explainability methods on CNNs trained on real-world measurements, we also assess different evaluation scenarios, so that we can fairly attribute the generalization to unseen areas. Ultimately, we want to clarify whether the drive-test campaign from Fig. 1 provides a sufficient basis to derive a prototypical network planning tool exhibiting physically sound propagation mechanisms.

III. PROBLEM STATEMENT

Overall, we want to learn a *neural network* (NN) parametrization θ_{nn} to accurately predict the RSRP at unseen locations

$$\hat{y}_{dB} = g_{nn}([\mathbf{E}_{loc}, \mathbf{m}]; \theta_{nn}), \quad (1)$$

using a suitable representation of the local propagation environment \mathbf{E}_{loc} and additional metadata \mathbf{m} as the input. Before elaborating on the selected model architecture and environmental encoding, we first describe the data sets providing the foundation for the considered use case. In particular, the extensive drive-test campaign acting as training data and ground-truth, and the 3D city model describing the propagation environment. These data will also be the basis for the 3GPP UMa pathloss model which we deploy for reference. Note, that Tab. 1 in the Appendix summarizes the notation used throughout this work.

A. 3D BUILDING MODEL WITH ELEVATION DATA

For our evaluation, we utilize 3D environmental data provided by the city of Vienna, Austria. As sketched in Fig. 2, we combine the official 3D building model with a detailed elevation map² to obtain a comprehensive 3D city model in the area surrounding the UE location \mathbf{x}_{UE} . The building

²The data is publicly available under <https://www.wien.gv.at/stadtentwicklung/stadtvermessung/geodaten/bkm/> and <https://www.wien.gv.at/stadtentwicklung/stadtvermessung/geodaten/dgm/>

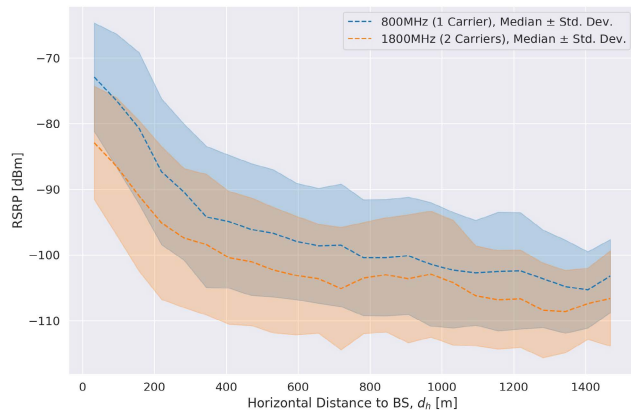


FIGURE 3. Measured RSRP over distance to BS.

model in Fig. 2a represents buildings as prisms³ together with their respective heights. In contrast, the elevation data in Fig. 2b is encoded in a raster format. Using *rasterio* [43], we combine the two data sources through rasterization of the building model and subsequent addition, such that we obtain a function $F_{\text{env}}(\cdot) = F_{\text{elevation}}(\cdot) + F_{\text{buildings}}(\cdot)$, offering a complete height profile with a resolution of 1 m. The final model thus covers the individual buildings and terrain features such as river beds or railway viaducts, offering a comprehensive description of the static propagation features. For a given measurement i , we can apply an affine transformation followed by masking to describe the environment through $f_{\text{env}}^{(i)}(d, s)$, acting as a local representation of $F_{\text{env}}(\cdot)$ centered around \mathbf{x}_{UE} and aligned towards the BS position \mathbf{x}_{BS} . As indicated in Fig. 2c, d and s represent the local coordinates parallel and perpendicular to the direct path, such that $(0, 0)$ describes the UE location. It follows, that the BS position is given by $(d_h^{(i)}, 0)$, where d_h is the horizontal distance between \mathbf{x}_{UE} and \mathbf{x}_{BS} . Throughout this work, we will further assume that $f_{\text{env}}^{(i)}(d, s)$ is constructed such that the elevation at \mathbf{x}_{UE} is set to zero.

B. DATA FROM VIENNA DRIVE-TEST CAMPAIGN

The measurements acting as the basis for our machine learning model stem from an extensive drive-test campaign conducted along the route shown in Fig. 1, covering over 95 km in a dense, urban environment. Using a *PCTEL MXflex* scanning receiver with omnidirectional antennas mounted onto the drive-test vehicle, we collected $\approx 750\,000$ RSRP measurements in a live 4G network. We provide a detailed description of the measurement equipment and potential uncertainties in Appendix VII-B. The passive scanner avoids exclusive BS assignment, such that we receive measurements from multiple BSs, sector antennas and frequencies at a single location. Overall, our data set includes 159 distinct *eNodeBs* (eNBs), covering 793 sectors among three carriers. We compensate for *global positioning system* (GPS) noise in a two-step

³The level of detail is LOD1.3, discarding overpasses for our purpose [42].

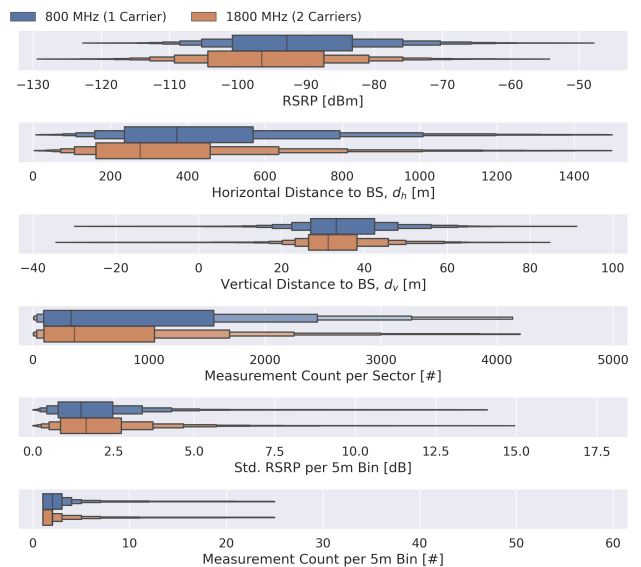


FIGURE 4. Characteristics of drive-test measurements.

procedure: First, we map each of the measurement locations onto the route traveled by the vehicle.⁴ In the second step, we filter for measurements with projection errors below 5 m and remove static measurements which can confound the data set [44]. This leaves us with a total of 629 624 measurements, 209 673 in the 800 MHz and 191 006 and 228 945 for the two distinct carrier frequencies in the 1800 MHz band. Through the cell identifiers, we can relate our measurements to network infrastructure data validated by the respective MNO. The resulting RSRP over the horizontal distance to the BS d_h is provided in Fig. 3 — as expected, the pathloss is higher for 1800 MHz as compared to 800 MHz. In Fig. 4, more detailed characteristics of the measurement campaign are provided in the form of boxenplots [45]. Through the operator data we can compute $d_v = (h_{\text{BS}} + e_{\text{BS}}) - (h_{\text{UE}} + e_{\text{UE}})$, the vertical distance to the BS derived from the BS and UE⁵ heights h_{BS} and h_{UE} , and the respective elevations e_{BS} and e_{UE} from $F_{\text{elevation}}(\cdot)$. Due to elevation differences, the resulting statistic also reports negative d_v in rare cases. In Fig. 4, we also assess the small-scale variance of the measurements by binning the data in a 5 by 5 m raster and reporting the standard deviation for individual sectors within each bin. We further provide the number of measurements for each of these bins, validating the absence of static measurements confounding the data set.

C. 3GPP URBAN MACRO BASELINE

Given detailed knowledge of the network infrastructure in the area of interest, we can utilize the empirical UMa channel model from 3GPP as a reference and baseline approach [3].

Based on features consisting of d_h , d_v , the transmitter frequency f , the horizontal ϕ_h'' and vertical ϕ_v'' alignment offset

⁴Due to the drive-test setup, we only consider outdoor measurements.

⁵We use $h_{\text{UE}} = 1.5$ m throughout this work.

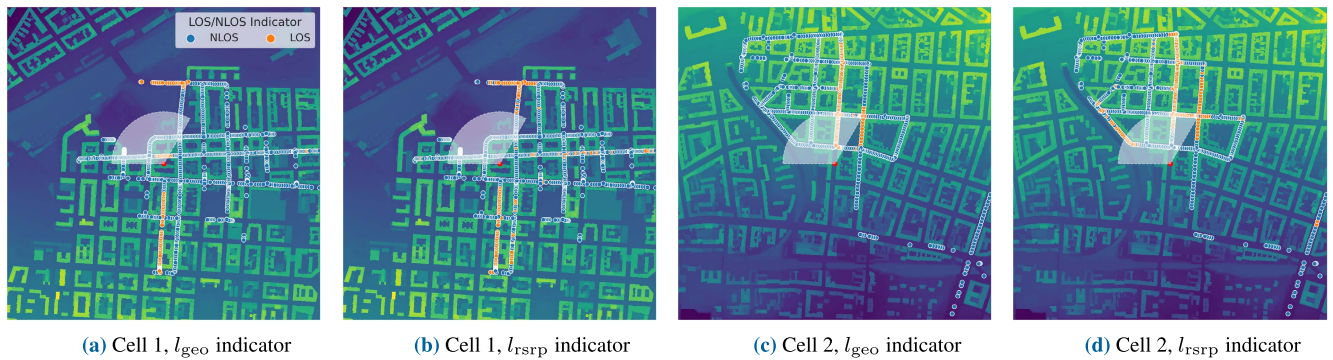


FIGURE 5. Obtained LOS indicators for exemplary cells from the drive-test data set, with indicated horizontal sector orientation.

with the sector antenna, as well as the transmit power $P_{Tx,ref}$. UMa provides an estimate of the received signal-strength. We collect all of these features in the vector \mathbf{m}_{uma} and provide a detailed description of the overall modeling procedure and the selected parameters in Appendix VII-A. This also involves the high-level environmental features which we set to values representative of the city of Vienna. Besides these high-level features, 3GPP UMa only distinguishes the *line of sight* (LOS) and *non line of sight* (NLOS) case, resulting in the estimate:

$$\hat{y}_{dB}^{(i)} = l_{ind}^{(i)} \cdot g_{los}(\mathbf{m}_{uma}^{(i)}) + (1 - l_{ind}^{(i)}) \cdot g_{nlos}(\mathbf{m}_{uma}^{(i)}). \quad (2)$$

Here, we introduce the LOS indicator $l_{ind} \in \{0, 1\}$, which can be determined geometrically from the 3D city model:

$$l_{geo}^{(i)} = \begin{cases} 1, & f_{env}^{(i)}(d, 0) < f_{los}^{(i)}(d) \text{ for } d \in (0, d_h^{(i)}] \\ 0, & \text{else} \end{cases}. \quad (3)$$

According to (3), a measurement location is in LOS if $f_{los}(d)$, the direct LOS propagation path from BS to UE

$$f_{los}(d) = d_v/d_h \cdot d + h_{UE}, \quad (4)$$

is above the blockages $f_{env}(d, 0)$ for the entire distance $d \in (0, d_h]$ — see Fig. 8b. For reference, we also consider an oracle l_{rsrp} , which sets the LOS indicator such that the error between prediction and measurement is minimized:

$$l_{rsrp}^{(i)} = \arg \min_{l_{ind}^{(i)} \in \{0,1\}} \left\{ \left| \hat{y}_{dB}^{(i)} - y_{dB}^{(i)} \right| \right\}. \quad (5)$$

Clearly, such an oracle indicator is not available prior to conducting the measurements and can thus only act as a reference.

Ideally, the geometry-based indicator l_{geo} and the RSRP oracle indicator l_{rsrp} agree for all measurements. However, the confusion matrix in Fig. 6 shows that this is only true for 77% of all measurements in our drive-test data set. In particular, we find that $l_{rsrp} = 1$ while $l_{geom} = 0$ is the case for 19% of all measurements, indicating that the $g_{nlos}(\cdot)$ model regularly underestimates the RSRP. Presumably, it undervalues diffraction effects and over the rooftop propagation, which can only partially be accounted for through a simple binary

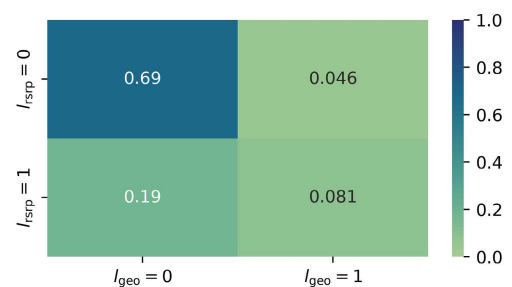


FIGURE 6. Confusion matrix for the LOS indicators.

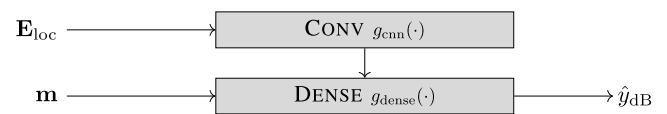


FIGURE 7. NN with E_{loc} and \mathbf{m} as input.

indicator and high-level environmental features. This effect is also apparent for the exemplary cells shown in Fig. 5, where the indicators agree qualitatively but still show several mismatches, especially in areas transitioning from NLOS to LOS. In the performance evaluation in Sec. V, we will show that this mismatch indeed results in a significant error.

IV. DEEP LEARNING FORMULATION

The preliminary results for the UMa model in Sec. III-C already highlight the difficulty of capturing a complex propagation environment through a single high-level indicator. For the deep learning formulation we directly operate on the available environmental data without introducing any abstractions. Given the black box nature of these approaches, this not only requires a careful encoding of the propagation environment, but also demands a suitable evaluation strategy which we will discuss in Sec. IV-B.

Generally, we follow the basic layout sketched in Fig. 7, such that the model receives two distinct inputs per measurement location, consisting of the tensor E_{loc} representing the local propagation environment as well as the

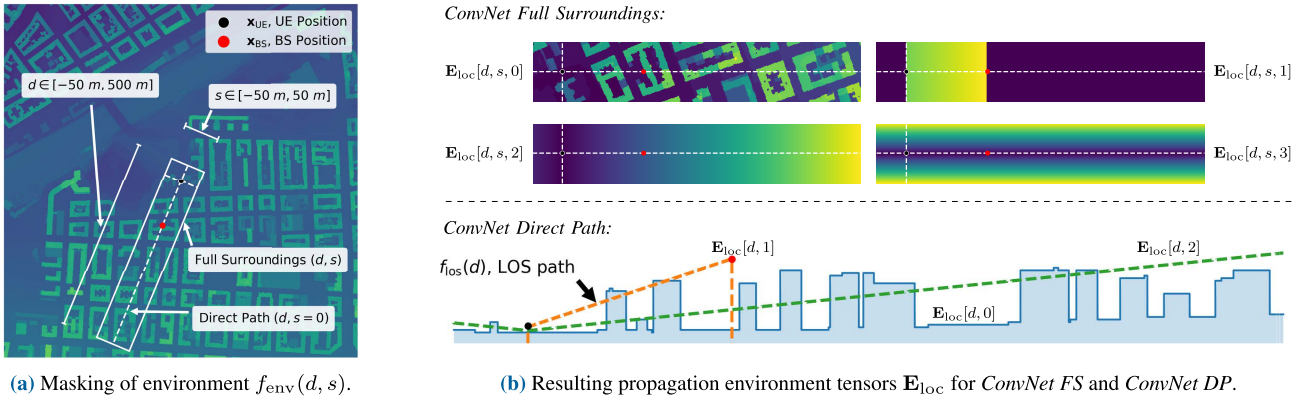


FIGURE 8. The \mathbf{E}_{loc} tensor encodes the full environment for *ConvNet FS* and the blockages along the direct path for *ConvNet DP*.

metadata vector \mathbf{m} :

$$\mathbf{m} = [d_h, d_v, f, l_{geo}, \hat{y}_{los}, \hat{y}_{nlos}, \phi_h'', \phi_v'']. \quad (6)$$

For consistency, the features in \mathbf{m} essentially mimic the input to the UMa model from Sec. III-C. However, to ensure generalization, we purposefully abstain from including features that can implicitly act as cell identifiers, such as the transmit power or the absolute antenna orientation. We instead pass the UMa predictions $\hat{y}_{los} = g_{los}(\cdot)$ and $\hat{y}_{nlos} = g_{nlos}(\cdot)$ from (2), which hide these absolute quantities but incorporate all the information. At the same time, this step also separates the influence of the antenna parameters from the processing of the blockages. As shown in Fig. 7, we concatenate this metadata vector \mathbf{m} with the output of a convolutional network $g_{cnn}(\cdot)$ processing the environmental encoding \mathbf{E}_{loc} , before a dense network $g_{dense}(\cdot)$ generates the final RSRP estimate \hat{y}_{dB} .

A. ENCODING THE ENVIRONMENT

In our case, the input tensor \mathbf{E}_{loc} is a fixed-sized representation of the propagation environment centered around the UE location with a consistent encoding of the BS position. Additionally, we guide the NN towards the physically relevant aspects by including a representation of the direct LOS path from BS to UE, relating it to the 3D blockages. Overall, we consider three different encoding variants termed *ConvNet Full Surroundings (ConvNet FS)*, *ConvNet Direct Path (ConvNet DP)* and *RefNet Metadata (RefNet MD)*.

i) The *ConvNet FS* variant operates on the full 3D environmental data in the surrounding of the UE. For an exemplary measurement with the local environment shown in Fig. 8a, we construct the four channel tensor \mathbf{E}_{loc} visualized in the upper part of Fig. 8b. In the first channel $\mathbf{E}_{loc}[d, s, 0]$, we directly collect the local environment $f_{env}(d, s)$, sampled in a 1 m resolution along $d \in [-50 \text{ m}, 500 \text{ m}]$ and $s \in [-50 \text{ m}, 50 \text{ m}]$.⁶ Hence, we cover up to 500 m of the path from UE to BS. In the second channel $\mathbf{E}_{loc}[d, s, 1]$, we encode

the direct LOS path $f_{los}(d)$ from (4), such that the network can immediately relate it to the respective blockages. Here, we also indicate the BS position by setting the entries to -1 for $d \notin (0, d_h]$. Inspired by [16], the remaining two channels $\mathbf{E}_{loc}[d, s, 2]$ and $\mathbf{E}_{loc}[d, s, 3]$ act as a linear encoding of the UE position in the interval $[0, 1]$ and also address the inability of CNNs to learn translation dependencies [46]. In summary, the $g_{cnn}(\cdot)$ input for this variant thus consists of the four image like inputs shown in Fig. 8b collected in the tensor \mathbf{E}_{loc} , which is combined with \mathbf{m} from (6) following Fig. 7.

ii) For *ConvNet DP*, we follow the same principle as above, but only pass the environment along the direct path given by $s = 0$ to the model — see Fig. 8a. As indicated in the lower part of Fig. 8b, the environmental representation in $\mathbf{E}_{loc}[d, 0]$ is now the sequence obtained by sampling $f_{env}(d, 0)$ along $d \in [-50 \text{ m}, 500 \text{ m}]$. The same is true for the second channel $\mathbf{E}_{loc}[d, 1]$, again consisting of the profile of the direct LOS path $f_{los}(d)$ from (4) with the BS position incorporated through the -1 entries. We also include the linear UE position encoding along the d coordinate in a third channel $\mathbf{E}_{loc}[d, 2]$. Overall, this variant thus receives the three channel sequence input \mathbf{E}_{loc} together with the metadata vector \mathbf{m} .

iii) As a reference, we also include the *RefNet MD* network, which completely omits the CNN input in Fig. 7 and only operates on the metadata vector \mathbf{m} . Hence, this reference network only receives environmental information through the binary indicator l_{geo} and can thus be seen as a machine learning equivalent to the UMa baseline. However, it allows us to differentiate the contribution of the full scale \mathbf{E}_{loc} inputs for *ConvNet FS* and *ConvNet DP* from error reductions achieved by simply fitting to the measurement distribution through the information in the metadata vector \mathbf{m} .

Overall, we see these environmental representations as a natural 3D extension of the UE centered images from the literature. Meanwhile, the different encodings help us to better understand how the environmental data is processed. This aspect will also be at the center of Sec. VI, where we compare

⁶We prioritize the path from BS to UE over the perpendicular buffer zone.

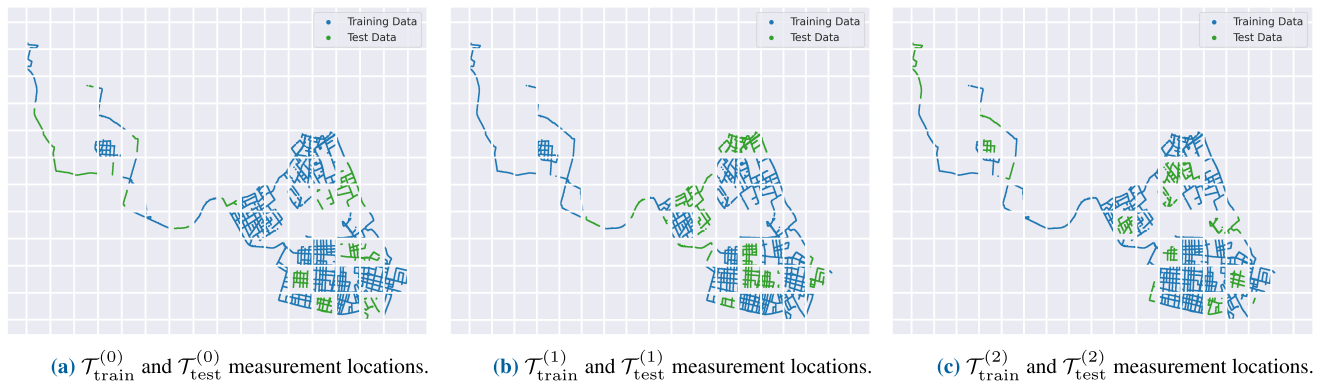


FIGURE 9. Resulting $\mathcal{T}_{\text{train}}^{(j)}$ and $\mathcal{T}_{\text{test}}^{(j)}$ folds for *Prediction Mode*, using $d_{\text{grid}} = 500 \text{ m}$ and $d_{\text{buffer}} = 100 \text{ m}$.

the three variants based on the propagation mechanisms they display. Similarly, we will also study how the rich spatial data relates to the risk of overfitting, when evaluated without the necessary caution.

B. APPROPRIATE EVALUATION STRATEGY

As discussed in Sec. I, many works in the literature evaluate comparable signal-strength predictors on relatively small data sets with only a hand full of BSs [9], [16]. Frequently, also the assignment of measurements to the training and test sets is done randomly [16], [37]. While this can be valid for certain scenarios, we see the risk of a biased evaluation with drastically exaggerated performance, especially when considering the implicit spatial reference provided by the rich environmental data. We find that an evaluation without dedicated spatial separation does not clearly distinguish the performance in a local interpolation use case from genuine cross-area generalization. To study this effect, we assess our models in two different evaluation scenarios, one forcing the models to operate in a *Prediction Mode*, while the other one does not explicitly rule out error reductions by mimicking local *Interpolation*.

i) The *Interpolation* scenario essentially mimics a random split of the measurement campaign as commonly found in the literature. To address repeated measurements, we bin the data sets following a rectangular spatial grid with a grid-distance of $d_{\text{grid}} = 5 \text{ m}$, such that each bin covers an area of 5 by 5 m. We then randomly assign each of the bins either to the train $\mathcal{T}_{\text{train}}$ or test $\mathcal{T}_{\text{test}}$ sets, not enforcing any additional spatial separation. Hence, the *Interpolation* scenario allows us to quantify the risk of a biased evaluation due to the spatial reference provided by the environmental data. ii) The data set for the *Prediction Mode*, in contrast, is constructed with $d_{\text{grid}} = 500 \text{ m}$ and additionally incorporates a buffer distance of $d_{\text{buffer}} = 100 \text{ m}$ around $\mathcal{T}_{\text{test}}$ samples bordering $\mathcal{T}_{\text{train}}$ areas. As apparent in Fig. 9, this buffer further separates neighboring train and test locations by a minimum distance of d_{buffer} . Hence, local interpolation is unfeasible in this scenario, allowing us to study the generalization capabilities required for network planning.

As shown for the *Prediction Mode* scenario in Fig. 9, we additionally utilize cross-validation for both evaluation designs, splitting the binned data set into three subsets $\mathcal{D} = \mathcal{D}^{(0)} \cup \mathcal{D}^{(1)} \cup \mathcal{D}^{(2)}$. With the individual procedures from above, we obtain three different train $\mathcal{T}_{\text{train}}^{(j)} = \bigcup_{i \neq j} \mathcal{D}^{(i)}$ and test $\mathcal{T}_{\text{test}}^{(j)} = \mathcal{D}^{(j)}$ data sets each. Appendix VII-C details the characteristics of all individual folds for both the *Interpolation* and the *Prediction Mode* evaluation.

C. MODEL CONFIGURATION & TRAINING

We implement our models in *Tensorflow* and use *Keras-Tuner* for hyperparameter optimization [47], [48]. In general, we keep the same network layout from Fig. 7 for each of the three variants but adapt it to the requirements of the individual input encodings. While we completely omit the $g_{\text{cnn}}(\cdot)$ block for *RefNet MD*, we construct it from 2D convolutional layers for *ConvNet FS* and from 1D convolutional layers for *ConvNet DP*. Similarly, we use the same basic blocks for $g_{\text{dense}}(\cdot)$ across all scenarios. Note that for ease of readability, we move the detailed description of the particular model configurations to Appendix VII-C. Starting from the same basic blocks, we still adapt the individual hyperparameters — such that the number of neurons and filters, the kernel size, as well as the selected dropout values in $g_{\text{dense}}(\cdot)$ and $g_{\text{cnn}}(\cdot)$ reflect the level of regularization required for each of the three variants. Note, that this optimization is conducted individually for *Interpolation* and *Prediction Mode*. To ensure a proper evaluation, we use a consistent hyperparameter configuration for all three folds. For training, we rely on the well-known *Adam* optimizer with a *mean squared error* (MSE) loss function, a batch size of 32, a learning rate of 5×10^{-4} and further utilize *EarlyStopping*. During training, we conduct data augmentation for *ConvNet FS* by randomly flipping \mathbf{E}_{loc} along the d axis. In contrast to continuous rotations from the literature [9], [36], this does not disrupt the inputs but rather exploits their inherent symmetry.

V. PERFORMANCE EVALUATION & RESULTS

We first evaluate the performance at the unseen test positions in *Prediction Mode* mode before we compare the results to

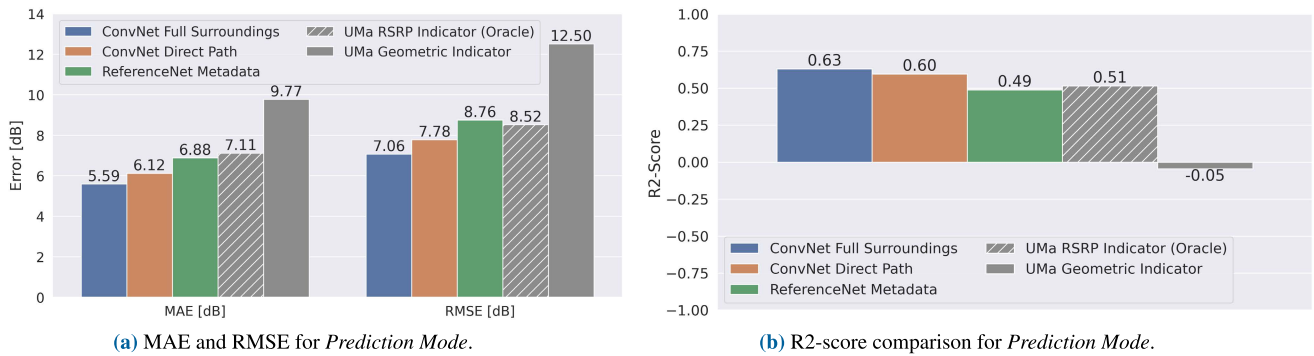


FIGURE 10. Performance in *Prediction Mode*, computed over all test folds $\mathcal{T}_{\text{test}}^{(j)}$ with $j \in \{0, 1, 2\}$.

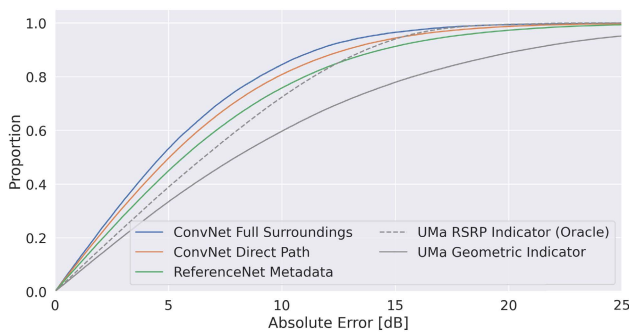


FIGURE 11. Error ECDF for *Prediction Mode*.

the *Interpolation* scenario, studying the random training and test split commonly found in the literature.

A. FAIR EVALUATION IN PREDICTION MODE

Fig. 10 summarizes the error for the considered models in the *Prediction Mode* evaluation. Here, Fig. 10a provides *mean absolute error* (MAE) and *root mean squared error* (RMSE), while Fig. 10b relates the error to the variance through the R2-score — the error ECDF is further shown in Fig. 11. Across all these metrics, we observe a consistent error reduction with an increasing degree of environmental data provided. As such, *ConvNet FS* is the best performing variant, with a MAE of 5.59 dB and an RMSE of 7.06 dB. It is followed by *ConvNet DP*, only considering the direct path between UE and BS, achieving an MAE of 6.12 dB and an RMSE of 7.78 dB. Interestingly, the *RefNet MD* model with 6.88 dB MAE and 8.76 dB RMSE outperforms the 3GPP UMa baseline from Sec. III-C with 9.77 and 12.50 dB respectively, even though it does not have access to any additional environmental information. This gap between empirical models and the *RefNet MD* showcases the ability of deep learning methods to adapt to the characteristics of a specific measurement campaign — which do not necessarily generalize to other data sets. However, the *RefNet MD* performance provides a machine learning baseline allowing for fair attribution of the error reduction achieved through the environmental encodings. In particular, we observe an RMSE reduction

of around 1 dB for *ConvNet DP* over *RefNet MD*, which we can attribute to the environmental data along the direct path from UE to BS. A subsequent improvement of 0.70 dB can then be achieved for *ConvNet FS*, processing the complete three-dimensional environment in the area surrounding the propagation path. Meanwhile, the poor performance of the *UMa Geometric Indicator* is not surprising, considering the findings in Sec. III-C. We have seen that capturing the environment only through a simple binary indicator can not sufficiently account for diffraction and reflection effects — inducing significant errors for the 23% indicator mismatch with the oracle model. While *RefNet MD* can compensate for that by fitting to the distribution of the measurements, the network planning scenarios in Sec. VI-B will show that it too provides inherently binary signal-strength maps with discrete jumps from NLOS to LOS. In contrast, the results show a consistent error reduction achieved by processing environmental data directly through the deep learning framework. While a significant error floor remains, we attribute it mainly to material properties and non-static blockages not being accounted for.

B. EFFECT OF INSUFFICIENT SPATIAL SEPARATION

Before we assess the propagation mechanisms also qualitatively in Sec. VI, we first examine the performance under the *Interpolation* scenario obtained through a random assignment of measurements into $\mathcal{T}_{\text{train}}$ and $\mathcal{T}_{\text{test}}$.

In Sec. IV-B, we discussed the problem that environmental data could implicitly expose a spatial reference, allowing the NNs to potentially learn a local interpolation setup instead of the underlying propagation mechanisms. To study this effect, we first quantify the error for the worst-case scenario, assuming the networks can perfectly recover the spatial reference. For this, we consider a *K-Nearest-Neighbor* (kNN) scheme with $k = 1$, which, for each sample in $\mathcal{T}_{\text{test}}$, simply reports the RSRP value of the closest measurement from the same sector in $\mathcal{T}_{\text{train}}$ [49]. Fig. 12 highlights the apparent difference between the *Interpolation* and the *Prediction Mode* scenarios with regards to the potential error reductions through interpolation they provide. We conclude that the *Prediction*

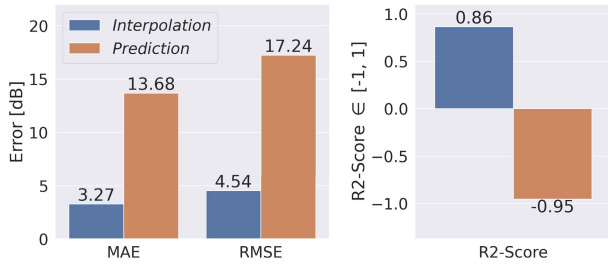


FIGURE 12. kNN performance for *Interpolation* and *Prediction*.

Mode does indeed prohibit the use of an absolute spatial reference — with an MAE of 13.68 dB and a negative R2-score of -0.95 . For the *Interpolation* scenario, the error reductions of kNN compared to *Prediction Mode* are significant, with an MAE of 3.27 dB and an R2-score of 0.86. If the networks can exploit the spatial reference in the environmental data to tap into this potential, the reported performance metrics will be substantially exaggerated for network planning scenarios.

Fig. 13, suggest, that this is indeed the case for *ConvNet FS* which comes very close to the kNN performance in the *Interpolation* scenario. Clearly, these reductions are not caused by different baseline errors, as the UMA results only marginally differ from *Prediction Mode*. While we also observe an error reduction for the *RefNet MD* model, presumably due to a closer match of training and test distributions, the environmental data seems to play a significant role. Apparently, the local encoding acts not only as a model of the propagation environment but also as an implicit spatial reference. Hence, *ConvNet FS* benefits disproportionately from the *Interpolation* scenario, with a substantial improvement of 2.03 and 2.43 dB over the *Prediction Mode* metrics for MAE and RMSE respectively. Of course, we can not completely distinguish this effect from a closer match of the training and test distributions. Still, the significant performance increase for *Interpolation*, which does not translate to *Prediction Mode*, already rules out such a random train and test split to assess the generalization capabilities. Only in a suitable *Prediction* scenario with spatial separation can we isolate the influence of the spatial reference on the performance metrics and correctly assess the role of the environmental data for propagation modeling.

VI. MODEL EXPLAINABILITY & PLANNING SCENARIOS

With the errors quantified in Sec. V, we now take a closer look at the trained models in *Prediction Mode*. In particular, we apply explainability methods to better understand how the environmental data is processed and then assess the learned propagation mechanisms in network planning scenarios.

A. INPUT REGIONS MOST SENSITIVE TO BLOCKAGES

We have seen in Sec. V that the role the environment plays for prediction is not always apparent. Often in the literature, the environmental data is also not limited to the input regions most relevant for propagation modeling, but included within

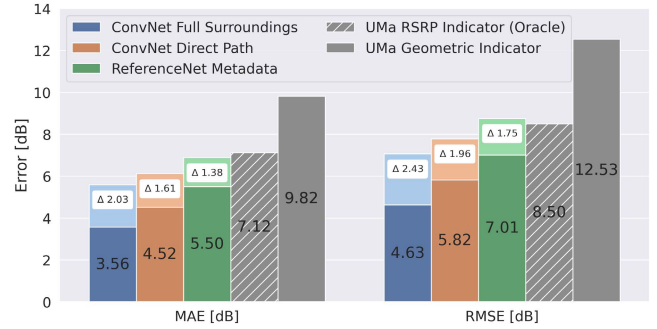


FIGURE 13. Error under *Interpolation* evaluation, with reductions Δ computed from the *Prediction Mode* baseline.

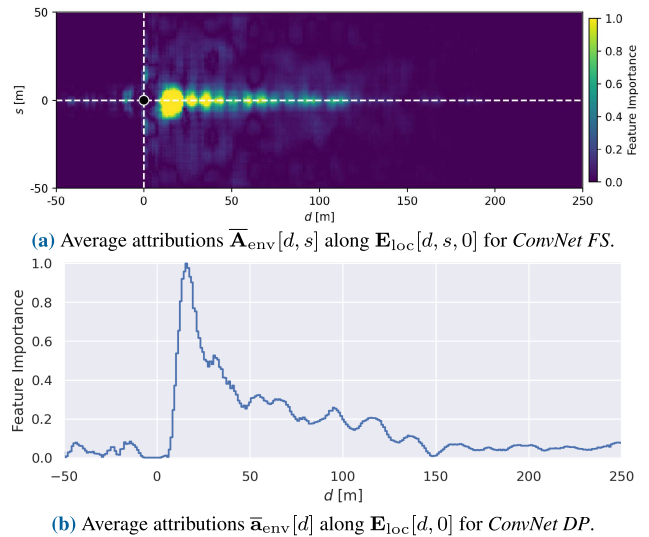


FIGURE 14. Attributions for trained models, revealing the input regions most sensitive to blockages.

large square buffer areas [9], [36], [50]. In our encoding, see Fig. 8a, we prioritize the environment along the direct propagation path, where a physically sound model should exhibit the highest feature importance. By applying *integrated gradients* (IG) [17] to our trained models, we can identify relevant input regions, i.e., the areas of \mathbf{E}_{loc} where blockages have the highest effect on the predicted RSRP. This not only validates our input region selection, but offers first insights into the learned propagation mechanisms.

In particular, we obtain an attribution matrix \mathbf{A}_{env} for each d and s value in $\mathbf{E}_{loc}[d, s, 0]$, representing the propagation environment for *ConvNet FS*. Following [17], this requires the definition of a suitable baseline, which we select as the all zero tensor $\mathbf{E}'_{loc}[d, s, 0]$ representing a flat terrain and the absence of blockages. \mathbf{A}_{env} is then obtained by integrating the gradients from the baseline \mathbf{E}'_{loc} to the original input \mathbf{E}_{loc} . In practice, we use a Riemann approximation with M steps:

$$\begin{aligned} \mathbf{A}_{env}[d, s] &= (\mathbf{E}_{loc}[d, s, 0] - \mathbf{E}'_{loc}[d, s, 0]) \\ &+ \sum_{m=1}^M \frac{\partial g_{nn}([\mathbf{E}'_{loc} + \frac{m}{M} \cdot (\mathbf{E}_{loc} - \mathbf{E}'_{loc}), \mathbf{m}])}{\partial \mathbf{E}_{loc}[d, s, 0]}. \quad (7) \end{aligned}$$

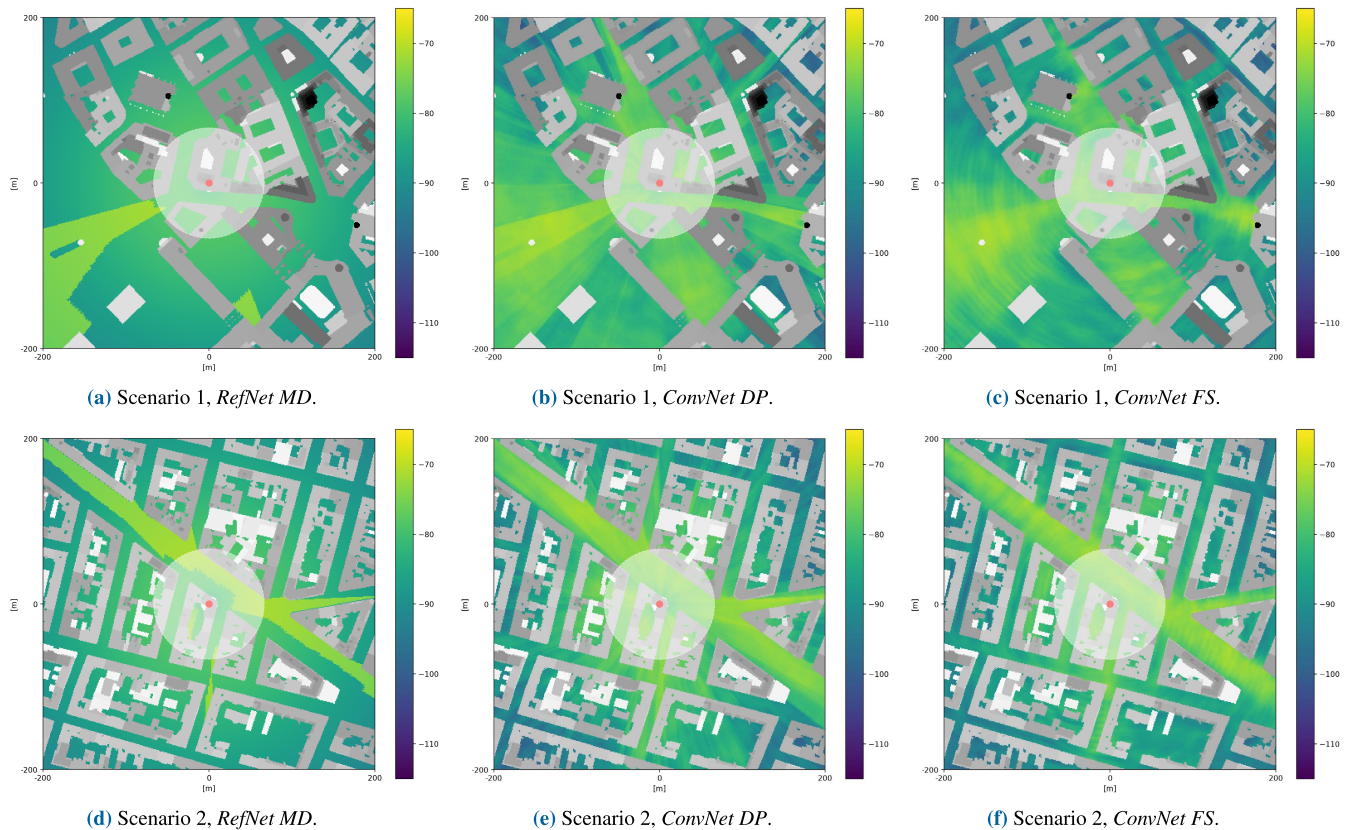


FIGURE 15. Comparison of RSRP maps for different encoding variants in unseen areas with artificially placed BSs, [dBm].

Similarly, we can also retrieve the attribution vector $\mathbf{a}_{\text{env}}[d]$ for *ConvNet DP*, describing the influence of blockages along $\mathbf{E}_{\text{loc}}[d, 0]$. To obtain representative results, we then average the individual attributions over a subset of all measurements [51]. This way, we generate Fig. 14, displaying the normalized attributions $\bar{\mathbf{A}}_{\text{env}}[d, s]$ and $\bar{\mathbf{a}}_{\text{env}}[d]$ for *ConvNet FS* and *ConvNet DP* respectively. Both variants were trained on $\mathcal{T}_{\text{train}}^{(0)}$ under the *Prediction Mode* evaluation strategy, with the attributions averaged over a random subset of 5 000 measurements from $\mathcal{T}_{\text{test}}^{(0)}$. We further used $M = 100$ for the Riemman approximation in (7).

It is apparent from Fig. 14a, that the feature importance $\bar{\mathbf{A}}_{\text{env}}[d, s]$ for *ConvNet FS* is highly concentrated along the direct path with $s = 0$. As such, the buildings blocking the direct propagation path also have the highest effect on the predicted RSRP — a first sign that the trained models exhibit physically sound propagation mechanisms. Meanwhile, the buildings perpendicular to the direct path are of limited relevance, which questions the common practise of including the large square buffer areas commonly found in the literature. For *ConvNet DP* in Fig. 14b, $\bar{\mathbf{a}}_{\text{env}}[d]$ follows a similar characteristic as $\bar{\mathbf{A}}_{\text{env}}[d, s]$ when evaluated along $s = 0$. Both display only limited feature importance in the area behind the measurement, while the peaks are reached a few meters along the direct path followed by an exponential decrease. It seems reasonable, that the first buildings along

the direct path have the highest effect on average. Interestingly, we also identify a wobble effect in Figs. 14a and 14b, which could be caused by the consistent pattern of the street widths in our data set. Likewise, we explain the zero attribution around the UE position by the absence of buildings on the streets where the measurements were collected. Besides these training data artifacts, the IG results illustrate attributions expected from physically sound propagation models — focusing on the direct path between UE and BS.

B. GENERATING DENSE SIGNAL-STRENGTH MAPS

With the IG analysis providing basic model explainability, we now want to assess the propagation mechanisms the models exhibit in realistic network planning scenarios. As such, we generate dense signal-strength maps in unseen areas of Vienna, with artificially placed BSs. Fig. 15, shows the obtained RSRP predictions of the three considered encoding variants for two artificial BS placed on buildings in the city center of Vienna. To generate these plots, we queried the trained models⁷ from Sec. V in the 400 by 400 m area surrounding the BS — in particular for every location not covered by a building on a grid with 1 m resolution. To better visualize the effect of the blockages, we further use a flat

⁷We use the model trained on $\mathcal{T}_{\text{train}}^{(1)}$ throughout the following evaluation because it offers a larger choice of unseen areas in the city-center.

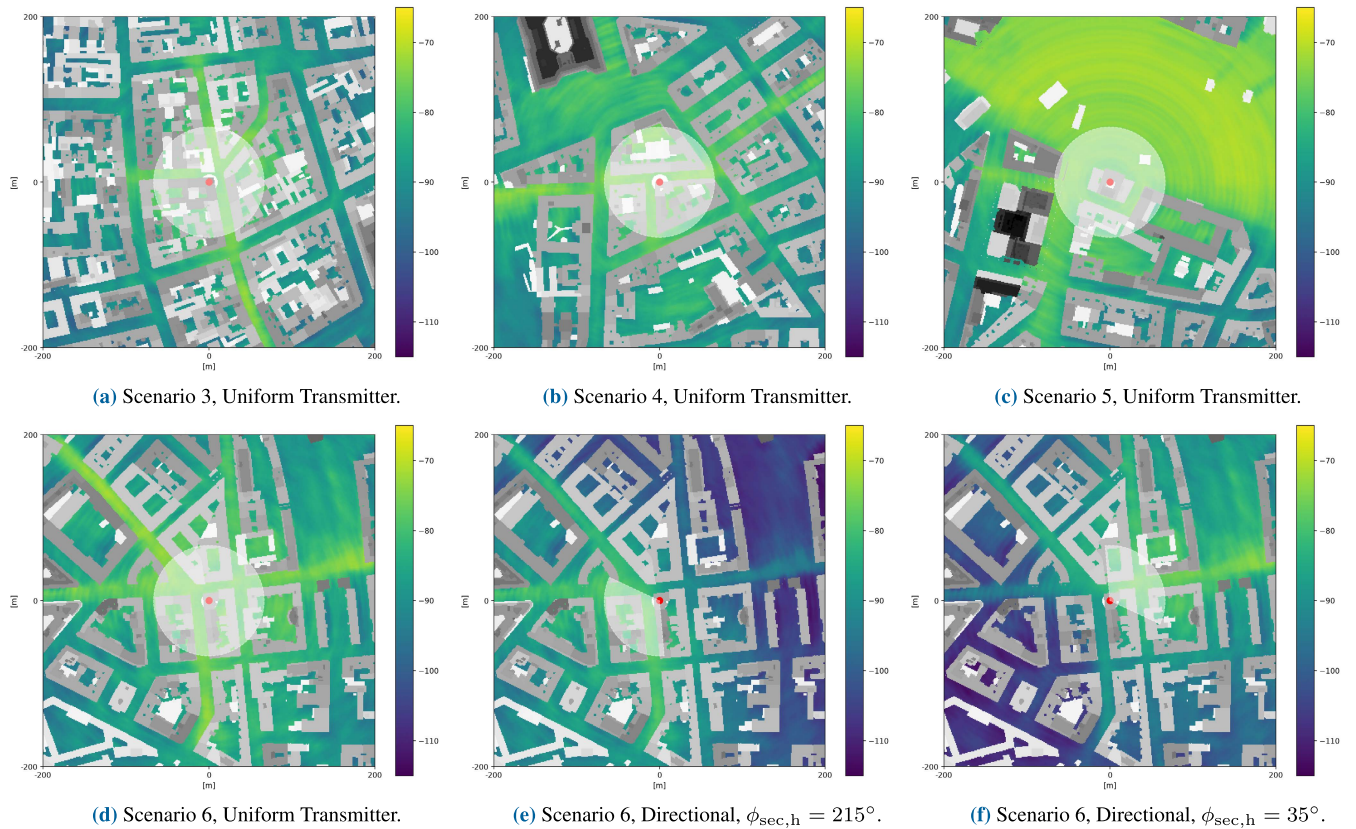


FIGURE 16. Exemplary generated RSRP maps for *ConvNet FS* in unseen areas with artificially placed BSs, [dBm].

terrain and, for now, consider a uniform antenna pattern by fixing $\phi_h'' = 0$, as indicated by the white circle in Fig. 15. We moreover select $h_{BS} = 30\text{ m}$, $f = 1800\text{ MHz}$, a transmit power of $P_{tx,ref} = 15\text{ dBm}$ and consider a vertical tilt $\phi_{sec,v}$ of zero, such that the antenna is aligned parallel to the ground. Note also that we abstain from prediction when $d_h < 10\text{ m}$, to adhere to the valid input range for UMa.

The resulting plots in Fig. 15 clearly reveal the inner workings of the three individual approaches. We observe that the *RefNet MD* model in Figs. 15a and 15d, which can only account for the environment through the I_{geo} indicator, reports sharp transition between the LOS and NLOS cases. We argue that this behavior also explains the frequent mismatches between the LOS indicators in Fig. 6, where the UMa model underrates the RSRP for NLOS cases. In contrast, the *ConvNet DP* model in Figs. 15b and 15e correctly identify diffraction over rooftops as one of the key propagation mechanisms in urban areas — leading to a smooth transition from NLOS to LOS. Still, it is apparent from the above results that the profile of the direct path alone is insufficient to learn diffraction effects along the horizontal plane. While the predictions for *ConvNet DP* are spatially consistent along the radial path from the BS position, we observe sharp transitions perpendicular to it. This is in stark contrast with the *ConvNet FS* model shown in Figs. 15c and 15f, which also incorporates the environment surrounding the

direct path. Apparently, this facilitates learning of horizontal diffraction effects — while *ConvNet DP* predicts a drop of the RSRP behind the single tall tower acting as a blockage in the upper left part of Fig. 15b, the *ConvNet FS* model in Fig. 15c accounts for the small width of the object, such that the drop is compensated by diffraction. Together with the performance evaluation in Fig. 10a, we thus conclude that the horizontal component is relevant and using only a side-view representation is inadequate.

Further network planning scenarios⁸ for *ConvNet FS* are provided in Fig. 16, again indicating a physically sound propagation model accounting for blockages in a reasonable way. It is apparent that the model can generalize the propagation mechanisms learned from a street-centric measurement campaign to unfamiliar surroundings such as courtyards. Again, it provides RSRP estimates within an adequate range of -110 to -70 dBm across all scenarios. Unsurprisingly, we can still observe some artifacts of our measurement campaign, in particular, the same wobble effect in Fig. 16c already observed in the attributions in Figs. 14b and 14a. While it is hard to draw concrete conclusions, we again explain this by a dominant street pattern in our data set. Overall, we are confident that such artifacts will average out for larger and more balanced data sets. Finally, we show that

⁸We again use the same scenario configuration as in Fig. 15.

the signal-strength map generation is not limited to specific BS configurations but can also be used to study the effect of different antenna heights or horizontal and vertical sector orientations. An example of this is shown in Figs. 16d, 16e and 16f, where the same scenario is evaluated for an omnidirectional and two directional antenna configurations with different horizontal alignments $\phi_{\text{sec,h}}$. Again we obtain qualitatively reasonable predictions, with signal-strength estimates significantly lower in the areas opposed to the sector antenna. At the same time, the predictions facing the sector antenna are consistent with Fig. 16d. While we can not expect generalization too far beyond the configurations in our data set, we see these results as a promising stepping stone toward future data-driven network optimization schemes.

VII. CONCLUSION

In this work, we combined measurements from an extensive drive-test campaign with a detailed 3D city model to derive a purely data-driven prototypical network planner. The large size of our data sets allows us to address several open questions, unveiling the inner workings of these black box approaches. Our results suggest that random train and test splits commonly found in the literature exhibit a high risk for a biased evaluation due to the rich spatial reference provided through the environmental data. In contrast, we find that a spatially separated training and test set enables a meaningful assessment of the generalization capabilities required for network planning. At the same time, our model explainability results also question the common practice of including large buffer areas around the UE location — which seem unnecessary given the highly concentrated feature importance along the direct path. In the surroundings of this path, a complete 3D representation is still beneficial, enabling the most comprehensive propagation model among the trained networks. In contrast, the *RefNet Metadata* model using a simple LOS indicator also generates inherently binary signal-strength maps, while the *ConvNet Direct Path* network can not account for diffraction in the horizontal plane. The 3D *ConvNet Full Surroundings* model also achieves the lowest prediction error with an RMSE of 7.06 dB for unseen locations. It is followed by the *ConvNet Direct Path* with 7.78 dB and the *RefNet Metadata* with 8.76 dB, both already significantly outperforming the UMa baseline. Overall we find the obtained results promising, considering that the trained *ConvNet Full Surroundings* was able to learn the dominant propagation mechanisms from a single drive-test campaign alone. Even though we can still observe some artifacts of our measurements, it generates spatially consistent and physically sound signal-strength maps, which proved suitable for a prototypical network planner. Considering the vast amounts of network traces and channel information already available for existing networks, we could imagine a prominent role for such data-driven schemes in the future. Deployed in an online fashion, they could effectively bridge the gap from passive monitoring and facilitate active, possibly dynamic network optimization.

APPENDIX

Tab. 1 summarizes the notation used throughout this work.

TABLE 1. Overview of used notation.

$\mathbf{x}_{\text{BS}}, \mathbf{x}_{\text{UE}}$...	BS and UE position
$h_{\text{BS}}, h_{\text{UE}}$...	BS and UE height
$e_{\text{BS}}, e_{\text{UE}}$...	elevation at BS and UE position
$f_{\text{env}}(d, s)$...	environment around \mathbf{x}_{UE} , towards \mathbf{x}_{BS}
$f_1(d)$...	direct path from BS to UE
d, s	...	parallel/perpendicular coords
\mathbf{m}_{uma}	...	input vector for UMa
d_h, d_v	...	horizontal/vertical distance to \mathbf{x}_{BS}
ϕ_h'', ϕ_v''	...	horizontal/vertical alignment offset
$\phi_{\text{sec,h}}, \phi_{\text{sec,v}}$...	horizontal/vertical antenna orientation
f	...	transmitter frequency
$P_{\text{tx,ref}}$...	reference signal transmit power
$l_{\text{geo}}, l_{\text{rsrp}}$...	geometric/RSRP based LOS indicator
$\mathbf{m}, \mathbf{E}_{\text{loc}}$...	metadata/environmental input for NNs
$\mathcal{T}_{\text{train}}^{(j)}, \mathcal{T}_{\text{test}}^{(j)}$...	train / test splits for fold j
\mathbf{A}	...	feature attributions for NNs

A. PARAMETERS FOR UMa PATHLOSS MODEL

Following model components from [3], we can expand the prediction of UMa from (2) for a given measurement i as:

$$\hat{y}_{\text{dB}}^{(i)} = P_{\text{tx,ref}}^{(i)} + P_{\text{offset}} + A_{\text{pat,h}}(\phi_h''^{(i)}) + A_{\text{pat,v}}(\phi_v''^{(i)}) - PL_{\text{UMa}}(d_h^{(i)}, h_{\text{ue}}, h_{\text{bs}}^{(i)}, f^{(i)}, l_{\text{ind}}^{(i)}), \quad (8)$$

where we get the reference signal transmit power $P_{\text{tx,ref}}$, the horizontal $\phi_{\text{sec,h}}$ and vertical $\phi_{\text{sec,v}}$ antenna orientations and other parameters directly from the operator. From $\phi_{\text{sec,h}}$ and $\phi_{\text{sec,v}}$, we can then account for the antenna characteristics. For the vertical pattern we directly use the model from [3], such that

$$A_{\text{pat,v}}(\phi_v'') = -\min \left\{ 12 \left(\frac{\phi_v''}{\phi_{3\text{dB}}} \right)^2, A_{\text{max}} \right\}, \quad (9)$$

with the default values of $\phi_{3\text{dB}} = 65^\circ$ and $A_{\text{max}} = 30$ dB. We select a flatter pattern using $\phi_{3\text{dB}} = 110^\circ$ and $A_{\text{max}} = 20$ dB for the horizontal characteristics which better matches our measurements and thus reduces the prediction error. For the pathloss component PL_{UMa} in (8), we further set an average street width of 10 m and the average building height of 25 m — representative of the drive-test area. We also introduce a small constant offset term through a line search

$$P_{\text{offset}} = \arg \min_{P_{\text{offset}}} \frac{1}{N} \sum_{i=1}^N |\hat{y}^{(i)} - y^{(i)}|, \quad (10)$$

minimizing the absolute error assuming $l_{\text{ind}}^{(i)} = l_{\text{rsrp}}^{(i)}$. The inclusion of $P_{\text{offset}} = 2$ dB accounts for our measurement equipment as well as potential model mismatches, and thus reduces the prediction error for UMa.



FIGURE 17. Drive-test vehicle with measurement antennas.

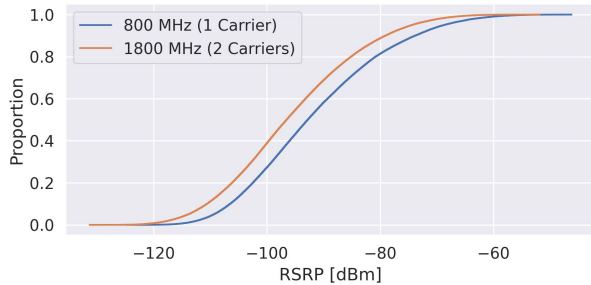


FIGURE 18. ECDF of measured RSRP.

TABLE 2. ConvNet FS for Prediction and Interpolation.

Layer / Parameter	Configuration
Conv2D Kernel Size ($g_{\text{cnn}}, \text{Prediction}$)	{(2, 2), (3, 3), (3, 3)}
Conv2D Filters ($g_{\text{cnn}}, \text{Prediction}$)	{32, 32, 10}
Spatial Dropout 2D $p(\cdot)$ ($g_{\text{cnn}}, \text{Prediction}$)	{0.52, 0.52}
Dense Layer Size ($g_{\text{dense}}, \text{Prediction}$)	{256, 256}
Dense Dropout $p(\cdot)$ ($g_{\text{dense}}, \text{Prediction}$)	{0.38, 0.38}
Conv2D Kernel Size ($g_{\text{cnn}}, \text{Interpolation}$)	{(3, 3), (4, 4), (4, 4)}
Conv2D Filters ($g_{\text{cnn}}, \text{Interpolation}$)	{128, 128, 32}
Spatial Dropout 2D $p(\cdot)$ ($g_{\text{cnn}}, \text{Interpolation}$)	{0.25, 0.25}
Dense Layer Size ($g_{\text{dense}}, \text{Interpolation}$)	{512, 512}
Dense Dropout $p(\cdot)$ ($g_{\text{dense}}, \text{Interpolation}$)	{0.10, 0.10}

B. DRIVE-TEST MEASUREMENT EQUIPMENT

Our measurements were conducted with a PCTEL MXflex scanning receiver [52], with two omnidirectional Panorama Antennas LGAMM-BC3G-26-3SP⁹ mounted onto the roof of the drive-test vehicle in Fig. 17. We further used an external EVK-M8U - U-blox¹⁰ GPS receiver to record the drive-test route. The MXflex is specifically designed for benchmarking use cases and allows us to collect RSRP measurements from up to 16 physical cell ids (PCIs) in parallel with a minimum RSRP detection level of -140 dBm and an accuracy of $\pm 1 \text{ dB}$ [52]. In our data set we measured a minimum RSRP of -131.3 dBm , see Fig. 18 for the complete distribution. Before conducting the drive-test campaign we calibrated our measurement equipment by directly wiring the MXflex to a reference eNB through tunable attenuators — the considered lab environment is described in detail in [53, p. 138]. Throughout this validation the combined error introduced by the BS transmit power instability and the MXflex scanning

⁹[https://www.panorama-antennas.com/site/LP\[G\]AMM?search=LGAMM-BC3G-26-3SP&description=1](https://www.panorama-antennas.com/site/LP[G]AMM?search=LGAMM-BC3G-26-3SP&description=1)

¹⁰<https://www.u-blox.com/en/product/evk-8evk-m8>

TABLE 3. ConvNet DP for Prediction and Interpolation.

Layer / Parameter	Configuration
Conv1D Kernel Size ($g_{\text{cnn}}, \text{Prediction}$)	{2, 4, 4}
Conv1D Filters ($g_{\text{cnn}}, \text{Prediction}$)	{32, 10, 10}
Spatial Dropout 1D $p(\cdot)$ ($g_{\text{cnn}}, \text{Prediction}$)	{0, 0}
Dense Layer Size ($g_{\text{dense}}, \text{Prediction}$)	{512, 512}
Dense Dropout $p(\cdot)$ ($g_{\text{dense}}, \text{Prediction}$)	{0.75, 0.75}
Conv1D Kernel Size ($g_{\text{cnn}}, \text{Interpolation}$)	{4, 2, 2}
Conv1D Filters ($g_{\text{cnn}}, \text{Interpolation}$)	{256, 256, 256}
Spatial Dropout 1D $p(\cdot)$ ($g_{\text{cnn}}, \text{Interpolation}$)	{0, 0}
Dense Layer Size ($g_{\text{dense}}, \text{Interpolation}$)	{512, 512}
Dense Dropout $p(\cdot)$ ($g_{\text{dense}}, \text{Interpolation}$)	{0, 0}

TABLE 4. Refnet MD for Prediction and Interpolation.

Layer / Parameter	Configuration
Dense Layer Size ($g_{\text{dense}}, \text{Prediction}$)	{512, 512}
Dense Dropout $p(\cdot)$ ($g_{\text{dense}}, \text{Prediction}$)	{0.52, 0.52}
Dense Layer Size ($g_{\text{dense}}, \text{Interpolation}$)	{512, 512}
Dense Dropout $p(\cdot)$ ($g_{\text{dense}}, \text{Interpolation}$)	{0, 0}

TABLE 5. Folds for Prediction and Interpolation.

	Fold	N_{train}	N_{test}
Prediction	$T_{\text{train}}^{(0)}, T_{\text{test}}^{(0)}$	453 045	79 503
	$T_{\text{train}}^{(1)}, T_{\text{test}}^{(1)}$	381 645	140 397
	$T_{\text{train}}^{(2)}, T_{\text{test}}^{(2)}$	424 558	105 687
Interpolation	$T_{\text{train}}^{(0)}, T_{\text{test}}^{(0)}$	422 268	207 356
	$T_{\text{train}}^{(1)}, T_{\text{test}}^{(1)}$	421 391	208 233
	$T_{\text{train}}^{(2)}, T_{\text{test}}^{(2)}$	415 589	214 035

receiver uncertainty was below 1 dB . Similar to the inherent fluctuations from the non-static drive-test scenario, we can safely assume that these deviations are zero mean and thus average out over the complete data set given the high number of measurements and the diverse set of BSs considered. We thus have no reason to believe that the NNs are able to compensate for these errors under our spatially separated evaluation strategy. Instead, the identified fluctuations will inherently be part of the reported prediction error of the NNs and the UMa baseline alike.

C. NEURAL NETWORK PARAMETERS AND TRAINING

We follow the same basic architecture for the individual blocks from Fig. 7 with all models implemented in Tensorflow [47]. $g_{\text{dense}}(\cdot)$ consists of two consecutive Dense layers with Relu activation functions, each followed by BatchNormalization and Dropout. Meanwhile, $g_{\text{cnn}}(\cdot)$ contains three consecutive Convolutional layers again with Relu activation functions, each followed by Spatial Dropout, $2 \times 2 \text{ MaxPooling}$ and BatchNormalization. We then arbitrarily select the first fold $T_{\text{train}}^{(0)}$ and $T_{\text{test}}^{(0)}$ to conduct hyperparameter optimization of the individual layer configurations using Keras-Tuner. The final layer configurations for the different encoding variants and evaluation scenarios are provided in Tabs. 2, 3 and 4. It is apparent, that the Interpolation scenario requires higher capacity models, while the main challenge for the Prediction Mode is sufficient regularization. Thus, we also observed

longer training processes of around 50 epochs in *Interpolation* mode, while 20 epochs were typically sufficient for *Prediction*. The sizes of T_{train} , T_{test} for the individual folds are further provided in Tab. 5. Note that N_{train} is smaller for *Prediction* due to the buffer area introduced in Sec. IV-B.

REFERENCES

- [1] Z. Niu, S. Zhou, and N. Crespi, "Greening 6G: New horizons," in *Shaping Future 6G Networks: Needs, Impacts, and Technologies*. Hoboken, NJ, USA: Wiley, 2021, pp. 39–53.
- [2] H. L. Bertoni, *Radio Propagation for Modern Wireless Systems*. London, U.K.: Pearson, 1999.
- [3] 5G; *Study on Channel Model for Frequencies From 0.5 to 100 GHz*, 3GPP, document TR 38.901, 2020.
- [4] Z. Yun and M. F. Iskander, "Ray tracing for radio propagation modeling: Principles and applications," *IEEE Access*, vol. 3, pp. 1089–1100, 2015.
- [5] M. Z. Chowdhury, M. Shahjalal, S. Ahmed, and Y. M. Jang, "6G wireless communication systems: Applications, requirements, technologies, challenges, and research directions," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 957–975, 2020.
- [6] J. Chen, U. Yatnalli, and D. Gesbert, "Learning radio maps for UAV-aided wireless networks: A segmented regression approach," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.
- [7] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: <http://www.deeplearningbook.org>
- [8] E. Krijestorac, S. Hanna, and D. Cabric, "Spatial signal strength prediction using 3D maps and deep learning," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.
- [9] J. Thrane, D. Zibar, and H. L. Christiansen, "Model-aided deep learning method for path loss prediction in mobile communication systems at 2.6 GHz," *IEEE Access*, vol. 8, pp. 7925–7936, 2020.
- [10] H. X. Nguyen, R. Trestian, D. To, and M. Tatipamula, "Digital twin for 5G and beyond," *IEEE Commun. Mag.*, vol. 59, no. 2, pp. 10–15, Feb. 2021.
- [11] R. M. Dreifuerst, S. Daulton, Y. Qian, P. Varkey, M. Balandat, S. Kasturia, A. Tomar, A. Yazdan, V. Ponnampalam, and R. W. Heath, Jr., "Optimizing coverage and capacity in cellular networks using machine learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 8138–8142.
- [12] M. Skocaj, L. M. Amorosa, G. Ghinamo, G. Muratore, D. Micheli, F. Zabini, and R. Verdone, "Cellular network capacity and coverage enhancement with MDT data and deep reinforcement learning," *Comput. Commun.*, vol. 195, pp. 403–415, Nov. 2022.
- [13] M. K. Müller, F. Ademaj, T. Dittrich, A. Fastenbauer, B. R. Elbal, A. Nabavi, L. Nagel, S. Schwarz, and M. Rupp, "Flexible multi-node simulation of cellular mobile communications: The Vienna 5G system level simulator," *EURASIP J. Wireless Commun. Netw.*, vol. 2018, no. 1, pp. 1–17, Dec. 2018.
- [14] Y. Zhang, J. Wen, G. Yang, Z. He, and J. Wang, "Path loss prediction based on machine learning: Principle, method, and data expansion," *Appl. Sci.*, vol. 9, no. 9, p. 1908, May 2019.
- [15] H. Cheng, S. Ma, and H. Lee, "CNN-based mmWave path loss modeling for fixed wireless access in suburban scenarios," *IEEE Antennas Wireless Propag. Lett.*, vol. 19, no. 10, pp. 1694–1698, Oct. 2020.
- [16] N. Kuno, M. Inomata, M. Sasaki, and W. Yamada, "Deep learning-based path loss prediction using side-view images in a UMA environment," in *Proc. 16th Eur. Conf. Antennas Propag. (EuCAP)*, Mar. 2022, pp. 1–5.
- [17] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3319–3328.
- [18] M. Molinari, M.-R. Fida, M. K. Marina, and A. Pescapè, "Spatial interpolation based cellular coverage prediction with crowdsourced measurements," in *Proc. ACM SIGCOMM Workshop Crowdsourcing Crowdsharing Big (Internet) Data*, Aug. 2015, pp. 33–38.
- [19] J. Riihijarvi and P. Mahonen, "Machine learning for performance prediction in mobile cellular networks," *IEEE Comput. Intell. Mag.*, vol. 13, no. 1, pp. 51–60, Feb. 2018.
- [20] S. Tripkovic, P. Svoboda, V. Raida, and M. Rupp, "Cluster density in crowdsourced mobile network measurements," in *Proc. IEEE 93rd Veh. Technol. Conf. (VTC-Spring)*, Apr. 2021, pp. 1–7.
- [21] L. Eller, P. Svoboda, and M. Rupp, "Propagation-aware Gaussian process regression for signal-strength interpolation along street canyons," in *Proc. IEEE 93rd Veh. Technol. Conf. (VTC-Spring)*, Apr. 2021, pp. 1–7.
- [22] O. Rozenblit, Y. Haddad, Y. Mirsky, and R. Azoulay, "Machine learning methods for SIR prediction in cellular networks," *Phys. Commun.*, vol. 31, pp. 239–253, Dec. 2018.
- [23] E. Alimpertis, A. Markopoulou, C. Butts, and K. Psounis, "City-wide signal strength maps: Prediction with random forests," in *Proc. World Wide Web Conf.*, 2019, pp. 2536–2542.
- [24] O. O. Shoewu, M. A. Adedoyin, L. A. Akinyemi, and L. I. Oborkhale, "Fuzzy-logic based path loss models for metropolitan environment," in *Proc. IEEE Int. Workshop Signal Process. Syst. (SiPS)*, Oct. 2018, pp. 71–76.
- [25] H.-S. Jo, C. Park, E. Lee, H. K. Choi, and J. Park, "Path loss prediction based on machine learning techniques: Principal component analysis, artificial neural network, and Gaussian process," *Sensors*, vol. 20, no. 7, p. 1927, Mar. 2020.
- [26] S. I. Popoola, E. Adetiba, A. A. Atayero, N. Faruk, and C. T. Calafate, "Optimal model for path loss predictions using feed-forward neural networks," *Cogent Eng.*, vol. 5, no. 1, Jan. 2018, Art. no. 1444345.
- [27] H. K. Hoomod, I. Al-Mejibli, and A. I. Jabboory, "Analyzing study of path loss propagation models in wireless communications at 0.8 GHz," *J. Phys., Conf. Ser.*, vol. 1003, May 2018, Art. no. 012028.
- [28] R. Enami, D. Rajan, and J. Camp, "RAIK: Regional analysis with geodata and crowdsourcing to infer key performance indicators," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2018, pp. 1–6.
- [29] L. Wu, D. He, B. Ai, J. Wang, H. Qi, K. Guan, and Z. Zhong, "Artificial neural network based path loss prediction for wireless communication network," *IEEE Access*, vol. 8, pp. 199523–199538, 2020.
- [30] U. Masood, H. Farooq, A. Imran, and A. Abu-Dayya, "Interpretable AI-based large-scale 3D pathloss prediction model for enabling emerging self-driving networks," 2022, *arXiv:2201.12899*.
- [31] M. Uccellini, F. Facchini, M. Sola, E. Sirignano, G. M. Vitetta, A. Barbieri, and S. Tondelli, "On the use of support vector machines for the prediction of propagation losses in smart metering systems," in *Proc. IEEE 26th Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Sep. 2016, pp. 1–6.
- [32] H. F. Ates, S. M. Hashir, T. Baykas, and B. K. Gunturk, "Path loss exponent and shadowing factor prediction from satellite images using deep learning," *IEEE Access*, vol. 7, pp. 101366–101375, 2019.
- [33] O. Ahmadien, H. F. Ates, T. Baykas, and B. K. Gunturk, "Predicting path loss distribution of an area from satellite images using deep learning," *IEEE Access*, vol. 8, pp. 64982–64991, 2020.
- [34] R. Levie, C. Yapar, G. Kutyniok, and G. Caire, "Pathloss prediction using deep learning with applications to cellular optimization and efficient D2D link scheduling," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 8678–8682.
- [35] X. Zhang, X. Shu, B. Zhang, J. Ren, L. Zhou, and X. Chen, "Cellular network radio propagation modeling with deep convolutional neural networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 2378–2386.
- [36] J. Thrane, B. Sliwa, C. Wietfeld, and H. L. Christiansen, "Deep learning-based signal strength prediction using geographical images and expert knowledge," in *Proc. IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–6.
- [37] T. Nagao and T. Hayashi, "Geographical clustering of path loss modeling for wireless emulation in various environments," in *Proc. 16th Eur. Conf. Antennas Propag. (EuCAP)*, Mar. 2022, pp. 1–5.
- [38] S. P. Sotiropoulos, P. Sarigiannidis, S. K. Goudos, and K. Siakavara, "Fusing diverse input modalities for path loss prediction: A deep learning approach," *IEEE Access*, vol. 9, pp. 30441–30451, 2021.
- [39] T. Imai, K. Kitao, and M. Inomata, "Radio propagation prediction model using convolutional neural networks by deep learning," in *Proc. EuCAP*, Mar. 2019, pp. 1–5.
- [40] Y. Tian, S. Yuan, W. Chen, and N. Liu, "Transformer based radio map prediction model for dense urban environments," in *Proc. 13th Int. Symp. Antennas, Propag. EM Theory (ISAPE)*, Dec. 2021, pp. 1–3.
- [41] H. Yu, Z. Hou, Y. Gu, P. Cheng, W. Ouyang, Y. Li, and B. Vucetic, "Distributed signal strength prediction using satellite map empowered by deep vision transformer," in *Proc. IEEE Globecom Workshops*, Dec. 2021, pp. 1–6.
- [42] H. Lehner and L. Dorffner, "Digital geoTwin Vienna: Towards a digital twin city as Geodata hub," *PFG-J. Photogramm., Remote Sens. Geoinf. Sci.*, vol. 88, no. 1, pp. 63–75, Mar. 2020.
- [43] S. Gillies. (2013). *Rasterio: Geospatial Raster I/O for Python Programmers*. [Online]. Available: <https://github.com/rasterio/rasterio>
- [44] V. Raida, P. Svoboda, and M. Rupp, "On the inappropriateness of static measurements for benchmarking in wireless networks," in *Proc. IEEE VTC-Spring*, May 2020, pp. 1–5.

- [45] H. Hofmann, H. Wickham, and K. Kafadar, "Letter-value plots: Boxplots for large data," *J. Comput. Graph. Statist.*, vol. 26, no. 3, pp. 469–477, Jul. 2017.
- [46] R. Liu, J. Lehman, P. Molino, F. P. Such, E. Frank, A. Sergeev, and J. Yosinski, "An intriguing failing of convolutional neural networks and the CoordConv solution," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–12.
- [47] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Oper. Syst. Design Implementation. (OSDI)*. USENIX, 2016, pp. 265–283. [Online]. Available: <https://www.tensorflow.org/about/bib>
- [48] T. O'Malley. (2019). *Kerastuner*. [Online]. Available: <https://github.com/keras-team/keras-tuner>
- [49] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, and M. Blondel, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Jan. 2011.
- [50] M. Lopez, T. B. Sorensen, I. Z. Kovacs, J. Wigard, and P. Mogensen, "Experimental evaluation of data-driven signal level estimation in cellular networks," in *Proc. IEEE 94th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2021, pp. 1–6.
- [51] I. Sarasua, S. Pölsterl, and C. Wachinger, "Hippocampal representations for deep learning on Alzheimer's disease," *Sci. Rep.*, vol. 12, no. 1, pp. 1–13, 2022.
- [52] (2022). *PCTEL MXflex Scanning Receiver Brochure*. [Online]. Available: <https://www.pctel.com/resource/mxflex-scanning-receiver-brochure/>
- [53] V. Raida, "Data-driven estimation of spatiotemporal performance maps in cellular networks," Ph.D. thesis, Inst. Telecommun., TU Wien, Vienna, Austria, 2021.



LUKAS ELLER (Graduate Student Member, IEEE) received the Dipl.Ing. degree in telecommunications from Technische Universität Wien (TU Wien), in 2020. He is currently a Project Assistant with the Institute of Telecommunications, TU Wien, focusing on end-user performance estimates for 4G and 5G networks. His research interest includes assessing the deployment of data-driven deep learning methods for the generation of performance models based on crowdsourced measurements.



PHILIPP SVOBODA (Senior Member, IEEE) received the Dr.Ing. degree in electrical engineering from Technische Universität Wien (TU Wien). He is currently a Senior Scientist with TU Wien, with his research focusing on the performance aspects of mobile cellular technologies. He is currently examining the feasibility of using crowdsourcing to conduct performance measurements on 4G and 5G mobile networks. His research interests include a common framework for evaluating the performance of mobile networks, guaranteeing reliable, and fair connectivity for end-users.



MARKUS RUPP (Fellow, IEEE) received the Dipl.Ing. degree from the University of Saarbrücken, Germany, in 1988, and the Dr.Ing. degree from the Technische Universität Darmstadt, Germany, in 1993. Until 1995, he held a Postdoctoral Researcher at the University of California at Santa Barbara, Santa Barbara, CA, USA. From 1995 to 2001, he was at the Wireless Technology Research Department, Nokia Bell Laboratories, Holmdel, NJ, USA. Since 2001, he has been a Full Professor of digital signal processing in mobile communications with TU Wien.

...