AI-assisted Agile Propagation Modeling for Real-time Digital Twin Wireless Networks

Ali Saeizadeh†, Miead Tehrani-Moayyed†, Davide Villa†, J. Gordon Beattie, Jr.*, Ian C. Wong*, Pedram Johari†, Eric W. Anderson†, Stefano Basagni†, Tommaso Melodia†
†Institute for the Wireless Internet of Things, Northeastern University, Boston, MA, U.S.A.
*VIAVI Solutions, Inc.
E-mail: †{saeizadeh.a, tehranimoayyed.m, villa.d, p.johari, er.anderson, s.basagni, melodia}@northeastern.edu,
*{gordon.beattiejr, ian.wong}@viavisolutions.com

***Abstract*—Accurate channel modeling in real-time faces remarkable challenge due to the complexities of traditional methods such as ray tracing and field measurements. AI-based techniques have emerged to address these limitations, offering rapid, precise predictions of channel properties through ground truth data. This paper introduces an innovative approach to real-time, high-fidelity propagation modeling through advanced deep learning. Our model integrates 3D geographical data and rough propagation estimates to generate precise path gain predictions. By positioning the transmitter centrally, we simplify the model and enhance its computational efficiency, making it amenable to larger scenarios. Our approach achieves a normalized Root Mean Squared Error of less than 0.035 dB over a 37,210 square meter area, processing in just 46 ms on a GPU and 183 ms on a CPU. This performance significantly surpasses traditional high-fidelity ray tracing methods, which require approximately three orders of magnitude more time. Additionally, the model's adaptability to real-world data highlights its potential to revolutionize wireless network design and optimization, through enabling real-time creation of adaptive digital twins of real-world wireless scenarios in dynamic environments.**

***Index Terms*—Deep Learning, Propagation Modeling, Channel Modeling, Ray Tracing, Digital Twin.**

I. Introduction

Real-time channel modeling is a cornerstone for modern and future wireless communication systems. It allows for network design and optimization in a risk-free Digital Twin (DT) environment to dynamically adapt to changing conditions of a real-world wireless network deployment. This enables better algorithmic designs and decision-making in a virtual environment, ultimately improving performance, efficiency, and user experience in real-world deployment. However, real-time modeling faces challenges due to the complexity of modern environments, including user mobility and varying conditions. Traditional methods like field measurements, ray tracing, and stochastic models fall short. Field measurements are accurate but costly, ray tracing is detailed but computationally intensive (e.g., in large Radio Frequency (RF) scenarios with mobility [1]), and stochastic models are less resource-demanding but sacrifice accuracy [2]. Attempts to address these shortcomings have provided limited results. For instance, efforts to accelerate ray tracing, such as parallelizing algorithms and using GPUs, have not yet achieved real-time capabilities due to lengthy computation times. Consequently, real-time support for mobility models and optimization remains impractical [1]. Ray tracing also lacks flexibility: Scenario changes necessitate rerunning the process, complicating adaptation to dynamic environments with new transmitters or receivers in real-time.

To address these limitations, advanced Artificial Intelligence (AI)-based techniques have been proposed. These methods leverage ground truth data from measurements or simulations to train data-driven models, enabling rapid and precise predictions of channel properties. By establishing a mapping function between the wireless environment and channel parameters, AI tools facilitate proactive network design. They effectively tackle challenges such as resource allocation, user mobility analysis, localization, and radio propagation modeling. AI-based techniques offer greater flexibility, scalability, and reduced computational complexity, thus enabling real-time propagation modeling in complex urban environments [3]–[9].

Leveraging AI for real-time modeling has multifold key motivations, including the need for advanced DT technology [10], [11] and enhanced telecommunication system design. Real-time DTs provide dynamic digital replicas of real-world network environments, enabling safe testing and evaluation of new configurations without impacting real-world performance [12]. They also facilitate data collection to train AI/Machine Learning (ML) models [13]. However, real-time DTs for telecommunication networks face challenges due to increased deployment density, complex architectures, and high-frequency communications. Accurate real-time propagation modeling is crucial for enhancing the physical layer of these DTs, ensuring they accurately reflect real-world conditions. Utilizing AI for this purpose can significantly improve network optimization

This work is supported in part by VIAVI Solutions, Inc., and by the National Telecommunications and Information Administration (NTIA)'s Public Wireless Supply Chain Innovation Fund (PWSCIF), Award No. 25-60-IF011.

and performance management in dynamic telecommunication environments [14]. AI-assisted propagation modeling enhances telecommunication system design by optimizing base station deployments, particularly RRHs or O-RUs, and visualizing signal propagation characteristics. When site conditions differ from the design, real-time modeling identifies discrepancies, enabling rapid adjustments to cell sites and sectors for optimal coverage and service.

Early efforts in AI-based channel modeling utilized conventional ML techniques like Random Forest, KNN, and SVM to model channel path loss, with Random Forest showing the best performance [15]. The introduction of Convolutional Neural Network (CNN) advanced the field by leveraging spatial correlations in images to enhance model accuracy using propagation features like distance and building maps [16]. Fully connected layers were used for regression, incorporating additional features such as frequency and antenna tilt. Further advancements included the use of satellite and aerial images as input channels, significantly improving prediction accuracy [17], [18]. Hybrid approaches combining rough estimates from physics-based models with Neural Network (NN) models also improved prediction performance [19].

Recent works have focused on incorporating more comprehensive input information and predicting channel parameter heat maps in one shot. CNN-based Auto Encoders and U-Net networks have shown promising results for fast and accurate predictions, with U-Net providing better performance at the cost of increased computational demands [3]–[9]. These advancements in AI-based channel modeling highlight its potential to revolutionize wireless network design by enabling real-time, accurate propagation modeling across various scenarios and environments. However, despite their promising performance, questions remain regarding the generalizability of these models and their evaluation with real-world measurements.

In this paper we address these concerns with the aim of enhancing the reliability of AI-based approaches in diverse and dynamic wireless environments. To achieve this, we employ a U-Net structure inspired by Lee and Molish [7], which allows the model to cover an entire area with a single inference while capturing spatial features effectively. We leverage two key inputs: an elevation map to accurately convey 3D geographical information to the model, and a rough estimation of the propagation model to maintain generalizability (Fig. 1).

This rough estimation is subsequently upsampled and refined into a high-fidelity propagation model. Unlike other approaches, we place the Transmitter (TX) at the center to reduce the model's complexity. This improves model performance compared to using an additional channel for the TX location. By implementing these changes, we address the generalizability issues of previous works, enabling real-time propagation modeling for any environment with available 3D geographical data. Our approach significantly improves both accuracy and computational efficiency. Specifically, we achieve a normalized Root Mean Squared Error (RMSE) of less than 0.035 dB over a 37,210 square meter area, with processing times of just 46 ms on a GPU. These results demonstrate the model's ability to rapidly provide high-fidelity propagation predictions, surpassing traditional ray tracing methods that require over 387.6 seconds. Additionally, refining the model with a small amount of measurement data shows an Root Mean Squared Error (RMSE) of 0.0113, demonstrating its adaptability to real-world data. The remarkable performance and efficiency of AI-driven techniques underscore their potential to revolutionize wireless network design and optimization, enabling real-time adaptation to dynamic and complex telecommunication environments.

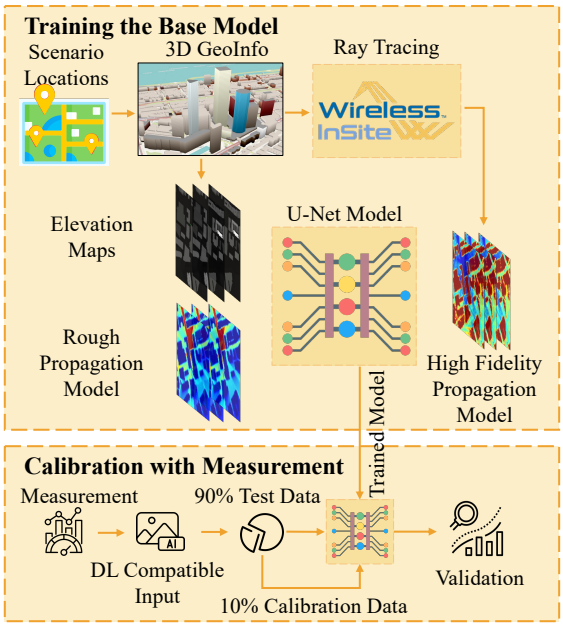


Fig. 1: Training process, data calibration, and model validation.

The paper is organized as follows. Section II describes data collection for training the model. Section III details the architecture and components of the Deep Learning (DL) model. Section IV presents performance results and an evaluation of the model. Finally, Section V concludes the paper.

II. Training the Model: Data Collection

To construct a comprehensive and precise dataset for predicting path gain (expressed in dB throughout this study), we utilized both ray tracing simulations and empirical measurements. Our objective is to forecast the Path Gain (PG), as delineated in equation (1), by leveraging the received power at the receiver (P_{RX}) and the transmitted power (P_{TX}).

$$\mathrm{PG}(t) = P_{\mathrm{RX}}(t) - P_{\mathrm{TX}}(t), \tag{1}$$

The Channel Impulse Response (CIR) can be defined as:

$$h(t,\tau) = \sum_{i=1}^{N} \alpha_i(t)\delta\left(\tau - \tau_i(t)\right), \tag{2}$$

where $\alpha_i(t)$ is the time-varying complex amplitude of the i-th path and $\tau_i(t)$ is the time-varying delay of the i-th path.

To extract the received power from the channel, we use the CIR as shown in (3)

$$\begin{aligned} P_{\mathrm{RX}}(t) &= 10\log_{10}\left(\int_{-\infty}^{\infty} |h(t,\tau)|^2 d\tau\right) \\ &= 10\log_{10}\left(\sum_{i} |\alpha_i(t)|^2\right). \end{aligned} \tag{3}$$

In a stationary environment with a mobile receiver, $PG(t)$ and $\mathrm{PG}(\boldsymbol{q}_{\mathrm{RX}})$, where $\boldsymbol{q}_{\mathrm{RX}}$ is the location of Receiver (RX) can be considered equivalent since the position $\boldsymbol{q}_{\mathrm{RX}}$ of the receiver varies with time t. Thus, modeling $\mathrm{PG}(\boldsymbol{q}_{\mathrm{RX}})$ effectively captures the time-varying nature of the path gain $\mathrm{PG}(t)$ as the receiver moves through different locations $\boldsymbol{q}_{\mathrm{RX}}$.

A. Ray Tracing

Urban environments pose challenges for conventional channel models, which often fail to accurately characterize channel properties. Ray tracing methods offer a potential solution for these complex scenarios. To build a comprehensive dataset, we employed Wireless InSite Ray Tracing software [20] by RemCom to collect high-fidelity data. The ray tracing model is configured by one diffraction and four reflections. In the scenario described below, simulating one transmitter takes 1:25:12 using CPU or 0:03:16 using GPU on a machine with two Intel Xeon E5-2660 processors with 28 cores and one Nvidia Tesla K40c GPU with 2880 CUDA cores.

In this study, we focus on the Northeastern University campus in Boston shown in Fig. 2, as an urban use case scenario to train and test the model. We consider a grid of potential RX locations, comprising 7,569 points (red points) spread over a 435×435 square meters area, and 61 TX locations (green points) situated at the corners of building rooftops for potential Base Station (BS). Additionally, we examine Fenway Park (42°20'26"N 71°05'38"W), as a separate location with a lower density of buildings (and not seen in the DL training phase) with another 16 TXs to evaluate the model's generalization capabilities. To ensure accuracy, we import a precise 3D model of the area using data gathered by Boston Planning and Developing Agency [21].

To prepare the input data for the model, we simulate ray tracing for all transmitter locations in the Northeastern University scenario, as shown in Fig. 2, to obtain path gain heat maps. These heat maps serve as the ground truth for training the model, with Fig. 3 presenting an example. We convert these maps

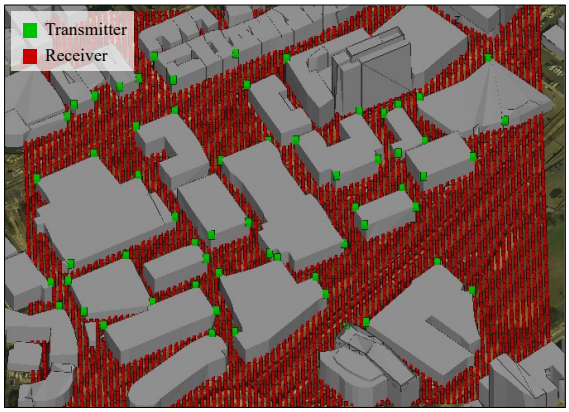
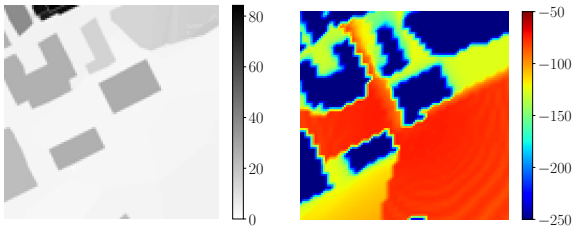


Fig. 2: Northeastern University, Boston Campus, used for generating the main dataset consisting of 61 TX and 7,569 RX. Location: 42°20'22"N, 71°05'14"W.

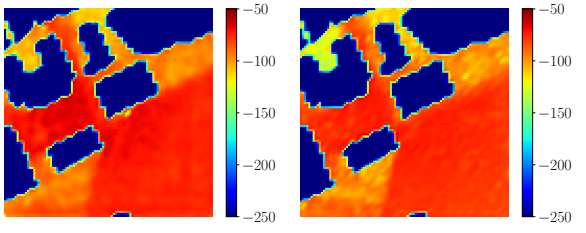
into gray-scale single-channel images to reduce the data requirements and mitigate overfitting. Since DL models require fixed input sizes, we crop the images to ensure uniform input sizes across different scenarios. Each image in our dataset measures 100×100 pixels, representing approximately a 1.929×1.929 meters area per pixel. Generating data for numerous scenarios is impractical, so we employ data augmentation techniques to create an augmented dataset from a limited synthetic dataset. Specifically, we use random rotations to incorporate the transmitter location into various input configurations, ensuring a substantial and diverse training dataset to enhance the model's robustness and accuracy.

B. Measurement Campaign

Although ray tracing is a powerful tool for modeling channels, real-world scenarios present unique variations that can significantly alter the channel charac-



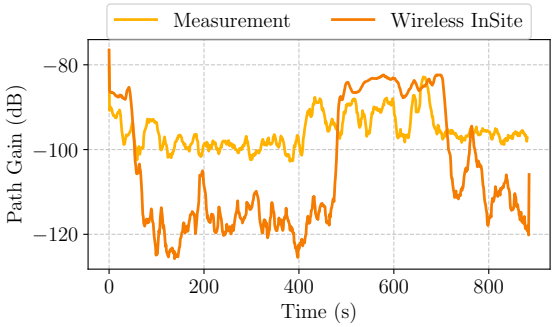
(a) Elevation Map (M) (b) Propagation Estimate (dB)



(c) Model's Output (dB) (d) Ray Tracing (dB)

Fig. 3: A comparison of the model's output (Fig. 3c) with the ground truth heatmap generated by Wireless InSite (Fig. 3d), showing its superior performance compared to (Fig. 3b).

(a) Path followed during the measurement campaign.



(b) The path gain derived from the measurement campaign with moving average applied.

Fig. 4: Measurement campaign at Northeastern University Campus in collaboration with VIAVI to gather real-world data for model validation and refinement.

teristics (see Fig. 4b). Three main factors contribute to these variations: (1) environmental dynamics, such as moving vehicles, constantly change the channel conditions; (2) the precise shapes and materials of buildings, which 3D models may not accurately capture, can affect channel behavior, as materials are often not modeled with exact fidelity; (3) interference from other users and background noise introduce additional environment noise into the channel, complicating accurate modeling.

To refine, calibrate, and validate the DL model in a real-world scenario, we conducted a measurement campaign in collaboration with VIAVI Solutions around the Northeastern University campus, as depicted in Fig. 4a. The details of the measurements are provided in Table I.

For channel sounding, the VIAVI Solutions Ranger, an RF waveform generator and capture platform, was used. It supports two full-duplex channels with 200 MHz bandwidth up to 6 GHz and captures three and one-half hours of recordings. Analysis and waveform modifications are done using the Signal Workshop application, which can be run locally or remotely.

The recorded In-phase and Quadrature (IQ) samples by the Ranger were processed in two steps: (1) extracting CIR ($h(\tau, t)$) from the raw IQ samples ($R(\tau, t)$); and (2) calibration to determine the actual Path Gain ($PG(\mathbf{q}_{\mathrm{RX}})$). For the first step, we utilized a Galois Linear Feedback Shift Register 14 (GLFSR-14) codeword, modulated it using Binary Phase-shift keying

TABLE I: Measurement Setup and Equipment

TX	
Radio Unit (RU)	Ettus USRP X410
Amplifier	Minicircuits ZHL-1000-3W+ (38 dB)
Antenna	Pasternack PE51OM1014 (6 dBi)
Location	42°20'25"N 71°05'16"W
RX	
RU	Ranger provided by VIAVI Solutions
Antenna	Waveform directional antenna (10 dBi)
Location	Mobile (see Fig. 4a)
Measurement Details	
Frequency	910 MHz
Bandwidth	122.88 MHz
Codeword	GLFSR-14
Synchronization	GPS clock for both TX and RX

(BPSK), resampled it to match the RX sampling frequency ($\mathbf{s}(t)$), and then correlated it with the received data. This process was repeated for each second of the data. Thus, CIR at the receiver $h[\tau, t = n]$ at the n-th second at location $\mathbf{q}_{\mathrm{RX}}$ using GPS logs is

$$h(\tau, t) = R(\tau, t) * \mathbf{s}(t). \tag{4}$$

To remove background noise from the channel, we identified the significant peaks in the channel for every length of the codeword:

$$\alpha = h(\tau, t) \cdot \mathbf{1}_{\{\mathrm{peaks}(h(\tau,t))\}}. \tag{5}$$

The indicator function $\mathbf{1}_{\{\mathrm{peaks}(h(\tau,t))\}}$ is defined as

$$\mathbf{1}_{\{\mathrm{peaks}(h(\tau,t))\}} = \begin{cases} 1 & \text{peaks of } h(\tau, t) \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

This function is 1 at the peak (threshold of $+3dB$ of the neighboring points) positions of $h(\tau, t)$ and 0 elsewhere, effectively isolating the peak values of the CIR. After extracting the sparse channel, we sum the values like in (3).

To accurately determine the path gain, a conducted calibration test Over-the-Cable (OTC) was performed with all equipment in the loop except for the antennas. Instead of antennas, 60 dB attenuators (L_{ATT}) were placed between RX and TX to prevent Analog to Digital Converters (ADCs) saturation, and the cable loss (L_{c}) was measured.

$$P_{\mathrm{RX}}^{\mathrm{OTC}} = P_{\mathrm{TX}} + G_{\mathrm{AMP}} - L_{\mathrm{c}} - L_{\mathrm{ATT}} \tag{7}$$

After field measurements, the average received power from the calibration data ($P_{\mathrm{RX}}^{\mathrm{OTC}}$) was subtracted from the received power in the actual measurement ($P_{\mathrm{RX}}^{\mathrm{OTA}}$) to compensate for amplifiers (G_{AMP}), cable losses (L_{c}), and TX power (P_{TX}).

$$P_{\mathrm{RX}}^{\mathrm{OTA}} = P_{\mathrm{TX}} + G_{\mathrm{AMP}} + G_{\mathrm{ANT}} + \mathrm{PG}^{\mathrm{OTA}} \tag{8}$$

The nominal gain of the antennas (G_{ANT}) was also considered in $\mathrm{PG}^{\mathrm{OTA}}$:

$$\mathrm{PG}^{\mathrm{OTA}} = P_{\mathrm{RX}}^{\mathrm{OTA}} - P_{\mathrm{RX}}^{\mathrm{OTC}} - L_{\mathrm{c}} - L_{\mathrm{ATT}} - G_{\mathrm{ANT}} \tag{9}$$

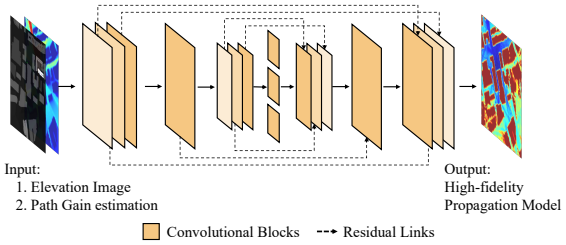


Fig. 5: U-Net architecture adapted from PMNet [7]. We use elevation maps and propagation estimates as inputs to achieve an accurate propagation model.

III. DL-Based Propagation Model

We propose a novel model shown in Fig. 5 inspired by PMNet [7], which originally takes two inputs: (1) a building map, indicating building locations with binary values (ones and zeros); and (2) a one-hot encoded TX location. We have modified these inputs for improved performance. Instead of using a binary building map, we now input an elevation map ($\mathbf{I}_{el}$) that shows building heights (Fig. 3a). Additionally, rather than a one-hot encoded TX location, we use a rough estimation of the propagation model derived from real-time ray tracing by Wireless InSite ($\mathbf{I}_{est}$), which runs in approximately 30 ms, as illustrated in Fig. 3b. The TX location is always at the center of the image, so there is no need to include the TX location in another channel. These modifications to the input enable our model to deliver high-fidelity propagation predictions in real time for different scenarios.

The DL model is designed to predict the path gain ($PG_{dB}(\boldsymbol{q}_{\mathrm{RX}})$) given the input $\mathbf{x}$ and the model parameters θ (weights and biases). However, the model's output is not limited to the path gain for a specific location, i.e., pixels, but rather it provides the path gain for an entire area, i.e., a heatmap. Mathematically, this can be represented as $P(PG_{dB}(\boldsymbol{q}_{\mathrm{RX}})|\mathbf{x},\theta)$, where $\mathbf{x}$ is the concatenated input $[\mathbf{I}_{el},\mathbf{I}_{est}]$, and θ encompasses all the parameters of the model. The objective is to learn the distribution of path gain conditioned on the input data and model parameters. This is achieved by optimizing the model parameters θ during training to minimize the prediction error.

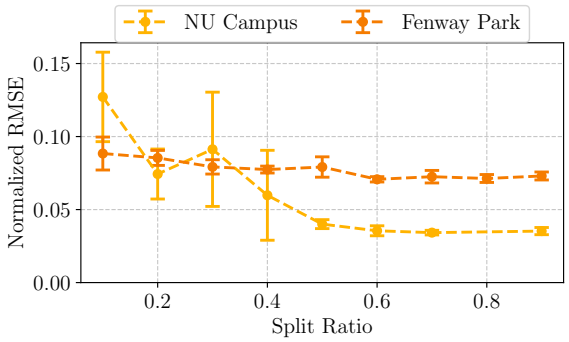
As mentioned in Section II, the main dataset consists of 61 different scenarios (TX locations), each with 100 augmented scenarios, resulting in a total of $6,100$ images. To validate the robustness and ensure a fair comparison of the model, we randomly split the data into different ratios ten times. This approach helps to: (1) ensure there is no bias towards specific scenarios; and (2) evaluate the model's generalization across various test ratios. Additionally, the trained model is tested in a different type of environment at Fenway Park with 16 TXs to further assess its performance. Besides ray tracing, measurement data is also used to validate the model. The entire process is illustrated in the diagram in Fig. 1.

IV. Metrics and Results

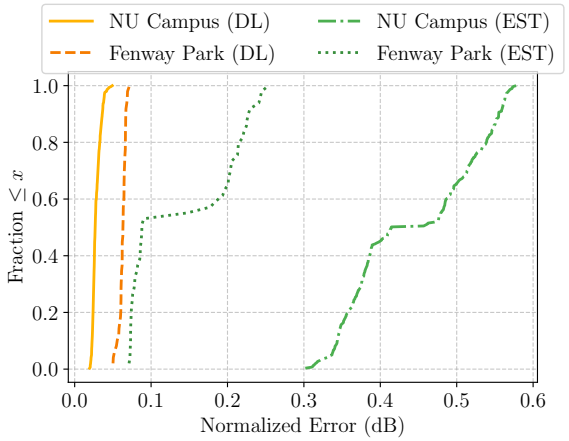
As described in Section III, the model is trained and tested with different split ratios using 10-fold cross-validation to find the optimal configuration. The results are shown in Fig. 6a. As observed, the model stabilizes after a 0.6 split ratio, indicating that no further training is required for effective inference. The error has been normalized across all plots, with the path gain range spanning from -50 to -250 dB.

Also, the Empirical Cumulative Distribution Function (eCDF) of error for the best-performing model across 10-fold cross-validation has been plotted using two different test datasets (unseen by the model): (1) the same area at Northeastern; and (2) a different area at Fenway Park (42°20'26"N, 71°05'38"W), where the environment is more open compared to the dense structure of the Northeastern Campus.

Results in Fig. 6b show that the model outperform traditional propagation estimation. Specifically, the median error for DL at Northeastern Campus is 0.0268, and at Fenway Park it is 0.0484. However, for the traditional model, the median error at Northeastern Campus is 0.4146, and at Fenway Park it is 0.0778. These results highlight the model's robustness and



(a) RMSE versus split ratio to evaluate the model's generalizability.



(b) eCDF of error for the DL model compared to real-time propagation modeling from the Wireless InSite model (EST), illustrating the distribution of errors.

Fig. 6: Model performance comparison at Northeastern University and Fenway Park.

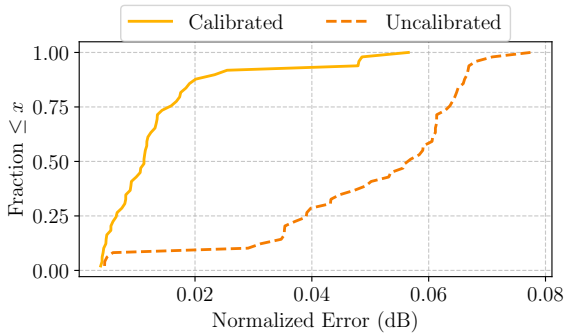


Fig. 7: eCDF of error w/ and w/o calibration using the measurement data (10% of the dataset).

adaptability to different environments, maintaining superior performance in both familiar and new scenarios.

In addition to ray tracing, we evaluate the model's performance after adding measurement data in the training phase (Fig. 7). For this, we use 90% of the points for testing. Initially, the error is relatively high (median of 0.0569 dB), but after refining the model with a small amount of measurement data, the performance improves significantly, reducing the median error to 0.0113 dB. This demonstrates that the DL model can be calibrated with measurement data, unlike common ray tracing software.

Besides the low overall error of the propagation model, using DL models enables near real-time accurate propagation modeling. From our test benches, real-time ray tracing by Wireless Insite takes approximately 30 ms to run, while high-fidelity ray tracing takes over 387.6 s to complete. However, our model requires only 46 ms on a GPU and 183 ms on a CPU to run with minimal error.

V. Conclusions

We introduce a novel real-time path gain estimator leveraging advanced deep learning techniques. Our approach integrates elevation maps and rough propagation model estimations for highly accurate propagation modeling. Placing the transmitter at the center simplifies the model while enhancing performance, with precise path gain predictions in near real-time. Our experimental findings demonstrate that the proposed model achieves a normalized RMSE of less than 0.035 dB across a 37,210 square meter area, executing in just 46 ms on a GPU and 183 ms on a CPU. This represents a significant advancement over traditional high-fidelity ray tracing methods, which typically require over 387.6 seconds on a GPU to complete. Our research underscores the potential of AI-enhanced techniques to transform wireless network design. It lays a foundation for real-time digital twins, promising efficient deployment and maintenance of future wireless infrastructure.

References

[1] M. Zhu, L. Cazzella, F. Linsalata, M. Magarini, M. Matteucci, and U. Spagnolini, "Digital twins of the EM environment: Benchmark for ray launching models," *arXiv preprint arXiv:2406.05042*, 2024.

[2] J. S. Seybold, *Introduction to RF propagation*. John wiley & sons, 2005.

[3] A. Gupta, J. Du, D. Chizhik, R. A. Valenzuela, and M. Sellathurai, "Machine learning-based urban canyon path loss prediction using 28 Ghz Manhattan measurements," *IEEE Transactions on Antennas and Propagation*, vol. 70, no. 6, pp. 4096–4111, 2022.

[4] V. V. Ratnam, H. Chen, S. Pawar, B. Zhang, C. J. Zhang, Y.-J. Kim, S. Lee, M. Cho, and S.-R. Yoon, "FadeNet: Deep learning-based mm-wave large-scale channel fading prediction and its applications," *IEEE Access*, vol. 9, pp. 3278–3290, 2020.

[5] R. Levie, Ç. Yapar, G. Kutyniok, and G. Caire, "RadioUNet: Fast radio map estimation with convolutional neural networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 6, pp. 4001–4015, 2021.

[6] S. Bakirtzis, J. Chen, K. Qiu, J. Zhang, and I. Wassell, "EM DeepRay: An expedient, generalizable, and realistic data-driven indoor propagation model," *IEEE Transactions on Antennas and Propagation*, vol. 70, no. 6, pp. 4140–4154, 2022.

[7] J.-H. Lee and A. F. Molisch, "A scalable and generalizable pathloss map prediction," *arXiv preprint arXiv:2312.03950*, 2023.

[8] K. Qiu, S. Bakirtzis, H. Song, J. Zhang, and I. Wassell, "Pseudo ray-tracing: Deep leaning assisted outdoor mm-wave path loss prediction," *IEEE Wireless Communications Letters*, vol. 11, no. 8, pp. 1699–1702, 2022.

[9] O. Ozyegen, S. Mohammadjafari, M. Cevik, K. El Mokhtari, J. Ethier, and A. Basar, "An empirical study on using CNNs for fast radio signal prediction," *SN Computer Science*, vol. 3, no. 2, p. 131, 2022.

[10] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, "Characterising the digital twin: A systematic literature review," *CIRP Journal of Manufacturing Science and Technology*, vol. 29, pp. 36–52, 2020.

[11] P. Testolina, M. Polese, P. Johari, and T. Melodia, "Boston twin: the Boston digital twin for ray-tracing in 6G networks," in *Proceedings of the 15th ACM Multimedia Systems Conference*, 2024, pp. 441–447.

[12] D. Villa, M. Tehrani-Moayyed, C. P. Robinson, L. Bonati, P. Johari, M. Polese, and T. Melodia, "Colosseum as a Digital Twin: Bridging Real-World Experimentation and Wireless Network Emulation," *IEEE Transactions on Mobile Computing*, pp. 1–17, In press 2024.

[13] A. Alkhateeb, S. Jiang, and G. Charan, "Real-time digital twins: Vision and research directions for 6G and beyond," *IEEE Communications Magazine*, 2023.

[14] L. U. Khan, W. Saad, D. Niyato, Z. Han, and C. S. Hong, "Digital-twin-enabled 6G: Vision, architectural trends, and future directions," *IEEE Communications Magazine*, vol. 60, no. 1, pp. 74–80, 2022.

[15] Y. Zhang, J. Wen, G. Yang, Z. He, and X. Luo, "Air-to-air path loss prediction based on machine learning methods in urban environments," *Wireless Communications and Mobile Computing*, vol. 2018, no. 1, 2018.

[16] T. Imai, K. Kitao, and M. Inomata, "Radio propagation prediction model using convolutional neural networks by deep learning," in *EuCAP 2019*, 2019, pp. 1–5.

[17] T. Hayashi, T. Nagao, and S. Ito, "A study on the variety and size of input data for radio propagation prediction using a deep neural network," in *EuCAP 2020*, 2020, pp. 1–5.

[18] 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.

[19] T. T. Nguyen, R. Caromi, K. Kallas, and M. R. Souryal, "Deep learning for path loss prediction in the 3.5 GHz CBRS spectrum band," in *IEEE WCNC 2022*, 2022, pp. 1665–1670.

[20] REMCOM, "Wireless Insite," https://www.remcom.com/wireless-insite-em-propagation-software.

[21] Boston Planning and Development Agency, "3D data and maps," 2024, accessed: 2024-06-30. [Online]. Available: https://www.bostonplans.org/3d-data-maps/