Machine Learning-Based Urban Canyon Path Loss Prediction Using 28 GHz Manhattan Measurements

Ankit Gupta, *Student Member, IEEE*, Jinfeng Du, *Member, IEEE*, Dmitry Chizhik, *Fellow, IEEE*, Reinaldo A. Valenzuela, *Fellow, IEEE*, and Mathini Sellathurai, *Senior Member, IEEE*

Abstract—Large bandwidth at millimeter wave (mm-wave) is crucial for fifth generation (5G) and beyond, but the high path loss (PL) requires highly accurate PL prediction for network planning and optimization. Statistical models with slope-intercept fit fall short in capturing large variations seen in urban canyons, whereas ray tracing, capable of characterizing site-specific features, faces challenges in describing foliage and street clutter and associated reflection/diffraction ray calculation. Machine learning (ML) is promising but faces three key challenges in PL prediction: 1) insufficient measurement data; 2) lack of extrapolation to new streets; 3) overwhelmingly complex features/models. We propose an ML-based urban canyon PL prediction model based on extensive 28 GHz measurements from Manhattan where street clutters are modeled via a light detection and ranging (LiDAR) point cloud dataset and buildings by a mesh-grid building dataset. We extract expert knowledge-driven street clutter features from the point cloud and aggressively compress the 3-D building information using a convolutional autoencoder. Using a new street-by-street training and testing procedure to improve generalizability, the proposed model using both clutter and building features achieves a prediction error [root-mean-square error (RMSE)] of 4.8 ± 1.1 dB compared to 10.6 ± 4.4 and 6.5 ± 2.0 dB for 3GPP line of sight (LOS) and slope-intercept prediction, respectively, where the standard deviation indicates street-by-street variation. By only using four most influential clutter features, the RMSE of 5.5 ± 1.1 dB is achieved.

Index Terms—Machine learning (ML), mesh grid, millimeter wave (mm-wave), path loss (PL) prediction, point cloud, urban street canyon.

I. Introduction

The fifth generation (5G) of mobile networks has adopted a much broader spectrum at higher frequency bands, such as millimeter-wave (mm-wave) bands, which promises very high data rates. To unleash the full potential of mm-wave communications, highly accurate channel modeling and path loss (PL) prediction are essential in foretelling cell coverage, planning deployment of the base stations (BSs), and optimizing network performance [1]. However, high bands come with the challenge of higher free space, scattering, and diffraction losses from the propagation environment. For example, in a typical urban street, buildings and street clutter, such as scaffolding, vehicles, and tree canopies, can significantly impact PL compared to lower frequency bands (wavelength of tens of centimeters). Although accurate PL estimation by employing fast and straightforward models is pivotal in network planning and optimization, they are yet to be fully understood in mm-wave frequencies at various propagation environments.

Numerous PL prediction models have been established in the literature, which can be classified into three major categories: statistical-based [2]–[8], deterministic-based [9], [10], and learning-based models [11]–[30]. Statistical models such as [2] provide a computationally efficient method by fitting particular equations to measurements obtained in different propagation environments [2]–[8]. The most widely adopted heuristic channel models, referred to as slope-intercept model hereafter, apply a linear fit to the measured PL data against the logarithm of the Euclidean distance between the transmitter (Tx) and the receiver (Rx). Deterministic models such as ray tracing, on the other hand, are based on the principles of physics to simulate wave transmission, reflection, and diffraction. Its PL prediction depends not only on the environment abstraction (geometry and material properties) but also on subjective parameter settings (e.g., the number of rays and the maximum number of reflections). Ray tracing at the mm-wave band is especially challenging [9], [10] since the detailed characterization of foliage and street clutter and associated reflection/diffraction calculation at short wavelengths all require customized approximations (such as empirical reflection coefficients, rough surface, and the inclusion of diffuse reflection and diffraction), making "tuning" (i.e., parameter adjustment against field measurement [32]–[34]) an essential part of the commercial ray-tracing tools in mm-wave network planning. Therefore, machine learning (ML)-based techniques have appeared as a promising alternative.

A. Previous Work

PL prediction can be considered as a regression problem in ML, where the features extracted from the propagation environment become its input and PL as a continuous variable output. We summarize some of the ML-based approaches for propagation environment modeling and PL

TABLE I
COMPARISON OF LEARNING-BASED PL PREDICTION [11]–[30]

Ref	Environment	Frequency	Key Features	ML tools	Train/test	Data Source
[11]	Indoor	1.89 GHz	Tx Position and gain, Tx height, distance	ANN	Split dataset	Measurements
[12]	Rural	0.881 GHz	Tx height, TLA, land usage	ANN	Split dataset	Measurements
[13]	Desert like area	1.8 GHz	Terrain profile	ANN, CNN	Split dataset	Measurements
[14]	Railway	0.930 GHz	Viaducts, cuttings, plains	ANN	Split dataset	Measurements
[15]	Urban	0.9, 1.8 GHz	Building height, LOS path	ANN, RF	Split dataset	Simulated
[16]	Urban	0.9 GHz	2D satellite images	VGG-16	Split dataset	Simulated
[17]	Urban	2.6 GHz	Tx gain, position and height, distance	ANN	Split dataset	Measurements
[18]	Urban	2.1 GHz	3D point cloud	ANN	Split dataset	Measurements
[19]	Urban	−	Propagation loss, distance	ANN	Split dataset	Measurements
[20]	Urban	$3-6$ GHz	Distance, frequency	ANN	Split dataset	Measurements
[21]	Urban	5.8 GHz	Antenna separation, distance, frequency	ANN, RF, SVR	Split dataset	Measurements
[22]	Urban, Suburban, Rural	2.4 GHz	Satellite images used for image segmentation	AE	Split dataset	Measurements
[23]	Alpine (Dense Trees)	2.4 GHz	Distance, vegetation variability, terrain, canopy coverage	ANN, RF, kNN	Split dataset	Measurements
[24]	Urban, Rural, Suburban	1.8 GHz	Longitude, latitude, altitude, clutter height, elevation	ANN	route-wise split	Measurements
[25]	Desert like area	1.8 GHz	Terrain profile	RF AE	Spatially disjoint split	Measurements
[26]	Public square	high frequency	Building height, distance, LOS/NLOS	CNN	site-wise split	Simulated
[27]	Urban	2.6 GHz	Satellite images	CNN	site-wise split	Measurements
[28]	Urban	5.9 GHz	2D city map, street map, location of cars	CNN	map-wise split	Simulated
[29]	Urban	28 GHz	2D-collapse of 3D-buildings	CNN	Split dataset	Simulated
[30]	Urban	28 GHz	Building height, terrain height, tree foliage height, and LOS information	CNN-based U-Net	Split dataset & city-wise split	Simulated

prediction [11]–[30] in Table I, highlighting the propagation environment, frequency, key features, training and testing procedures, PL data source, and ML tools such as artificial neural networks (ANNs), random forest (RF), convolutional neural network (CNNs), autoencoder (AE), and support vector regression (SVR). These works showcased the capability of ML-based methods and their potential in improving PL prediction accuracy. A more comprehensive review on ML-based PL prediction can be found in [31].

Many of the ML-based approaches [11]–[23], [29] focus on prediction for nearby links (i.e., interpolation). For studies (see [24]–[28], [30]) that do predict PL for new streets/areas (i.e., extrapolation), the influence of street clutter, such as trees and street furniture on PL, is either minimal or nonexistent. Besides, in most previous works, complex ML models are adopted as a black box, making it hard to interpret the connection between features and PL prediction. These complex ML models also make PL prediction vulnerable to overfitting since the training data size from measurements is usually too small compared to adjustable parameters in ML models.

Compared to studies such as [11]–[28] that are dedicated to the sub-6 GHz bands, ML-based PL prediction for mm-wave bands requires a much finer level of details in the environment description as scattering by small objects (tenths of wavelength) and material absorption loss are more significant for mm-wave signals. For example, about 30 dB street-by-street variation in median PL has been observed from field measurements in urban street canyons at 28 GHz [8]. Recently, ML-based PL prediction at mm-wave bands in urban street canyons has been proposed using CNN-based models [29], [30]. Both models are trained using ray-tracing simulated PL data and thus provide computationally efficient ways to approximate ray-tracing prediction. However, ray tracing itself at the mm-wave band in a cluttered environment needs to be improved to better match field measurements [32]–[34]. For example, a ray-tracing calculation was found to predict

about 10 dB stronger received power than observed in urban street canyons at 28 GHz [8]. This is thought to be due to difficulty in representing and modeling street clutter and foliage in traditional ray tracing. Therefore, accurate and practical models are still needed.

B. Our Contributions

We address three key challenges in ML-based PL prediction for mm-wave bands: 1) reliability due to no/insufficient measurement data; 2) generalizability due to weak/no capability of extrapolation; and 3) interpretability due to complex ML models and high-dimensional features.

1) Reliability: We utilize a large-scale dataset from 28 GHz field measurements [8] in urban street canyons, consisting of 1028 continuous-wave links from 13 streets in Manhattan from multiple rooftop sites colocated with commercial BS. The street clutter information, such as tree canopies and lampposts, is obtained from the open-source light detection and ranging (LiDAR) point cloud dataset [35], and the building information is obtained from the open-source 3-D mesh grid [36], which includes building height, facade shape, separation between the buildings, roads, and elevation information.

2) Generalizability: The generalizability we address it from three aspects.

1) *Street-by-Street Training and Testing Policy:* Thirteen independent training–testing combinations are created by choosing one street at a time for testing and the rest 12 streets for training. Such policy would test extrapolation of trained models to "never seen" streets.

2) *Aggressive Street Clutter and Building Feature Compression:* For each link, the high-resolution point cloud raw data are compressed to two numbers using heuristic approaches devised from expert knowledge in wave propagation, and the 3-D building information is compressed to a length-12 vector using CNN-based AEs to preserve the spatial characteristics.
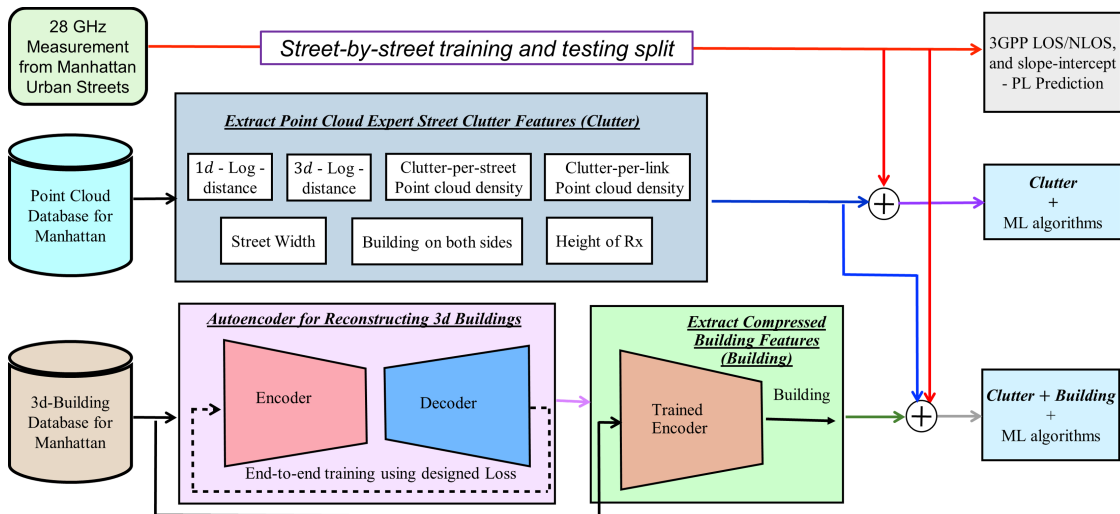
Fig. 1. Methodology adopted in this work.

3) *Reducing Adjustable Parameters for PL Prediction:* Simple ML-based regression algorithms, such as Lasso, Elastic-net, RF, and SVR, are adopted to mitigate overfitting.

3) Interpretability: We adopt human-friendly environment features and quantify the significance of each feature in the PL prediction. We define only seven expert knowledge-driven propagation environment features, referred to as clutter features hereafter, where each feature has a physical meaning attached. We quantify the importance of each clutter feature by the Lasso weight analysis and by comparing the PL prediction accuracy when only one feature is excluded.

To the best of our knowledge, this is the first time that both street clutter information and building information are used collectively for mm-wave PL prediction using a large-scale real-world propagation measurement at 28 GHz in urban streets. We show that our proposed model achieves the root-mean-square error (RMSE) of 4.8 dB averaged over all 13 streets with a 1.1 dB standard deviation that reflects the street-by-street variation. By only using the top four most influential features, our model achieves the prediction RMSE of 5.5 ± 1.1 dB (mean $\pm$ std). In contrast, the heuristic slope-intercept method and the 3GPP line of sight (LOS) model-based prediction have the RMSE of 6.5 ± 2.0 and 10.6 ± 4.4 dB, respectively. For the first time, we show that the ML-assisted PL predictions are more accurate than a measurement-based slope-intercept model with much smaller street-by-street variation. The methodology adopted in this work is summarized in Fig. 1.

C. Article Organization

The datasets of PL, point cloud, and 3-D building are described in Section II. Expert knowledge-based feature extraction from point cloud is presented in Section III and 3-D building feature compression using CNN-based AE is presented in Section IV. ML algorithms for PL prediction and the street-by-street training and testing methodology are elaborated in Section V. Performance evaluation is presented in Section VI and conclusions are drawn in Section VII.

II. PL DATA COLLECTION AND FEATURE SETS PREPROCESSING

In this section, we present an overview of the PL measurement in Manhattan and preprocessing of the point cloud dataset and the 3-D building mesh-grid dataset.

A. 28 GHz PL Measurement in Manhattan

The measurement campaign [8] is designed to characterize coverage in an urban street canyon, which can be coarsely defined as a straight road in an urban area that has buildings on both sides, with street clutter such as trees, vehicles, and lampposts placed along the road. Measurement was done from roof edge-mounted BS (i.e., urban macro) to user equipments (UEs) (1.5 m high) in the center of a sidewalk along the street, with no attempt to incorporate or eliminate blockage due to street clutter. The purpose is to resemble coverage of the street in the presence of such obstructions.

Measurements were performed from 7 building rooftops covering 13 streets from multiple areas of Manhattan of different street widths and widely varying amount of foliage, from the Pike Street in the lower east side to W 126th Street in West Harlem. The propagation environment differs significantly among the 13 streets. For example, street width ranges from 15 to 38 m, whereas building height varies from 11 to 93 m. Tree distribution along streets ranges from nothing to sparse, to very dense, and some streets even have road dividers with trees/bushes separating driveways. We summarize the diverse characteristics of the 13 streets in Table II.[1]

In total, 1028 links were measured from over 3800 m of street side walks with over 10 million individual power measurements, which were locally averaged per link to eliminate small-scale fading. In Fig. 2(a), we show the measured links and the slope-intercept fit to PL versus logarithmic Euclidean distance (d), which is given by

$$P = A + 10n \log_{10}(d) + \mathcal{N}(0, \sigma^2) \qquad (1)$$

where $A = 46.9$ dB is the 1 m intercept, $n = 3.1$ represents the slope, $\sigma = 6.3$ dB is the RMSE between fit and actual values,

[1]Details of clutter per street and clutter per link are deferred to Section III-A.

TABLE II
DIVERSE CHARACTERISTICS OF 13 MEASURED STREETS AND THEIR POINT CLOUD-BASED CLUTTER FEATURES

Street Number	Building height range	Terrain variation	#measured links	1D- distance range	Clutter-per-street	Clutter-per-link range	Street width	Building on both sides	Height Rx
1	$23 - 69$ m	$0 - 3$ m	88	$32 - 165$ m	0.51	$0 - 7$	25 m	✓	15 m
2	$11 - 43$ m	< 1 m	77	$69 - 338$ m	2.71	$0 - 59$	23 m	✓	35 m
3	$14 - 72$ m	$0 - 24$ m	94	$36 - 492$ m	1.86	$0 - 14$	28 m	✓	15 m
4	$12 - 61$ m	< 1 m	105	$35 - 453$ m	2.74	$0 - 26$	36 m	✓	25 m
5	$16 - 67$ m	$0 - 3$ m	85	$32 - 163$ m	0.51	$0 - 8$	25 m	✓	15 m
6	$14 - 72$ m	$0 - 23$ m	131	$34 - 495$ m	1.86	$0 - 17$	28 m	✓	15 m
7	$12 - 30$ m	< 1 m	87	$153 - 497$ m	1.70	$0 - 19$	15 m	✓	20 m
8	$14 - 93$ m	< 1 m	64	$92 - 490$ m	0.32	$0 - 1$	30 m	✓	22 m
9	$16 - 67$ m	< 1 m	66	$36 - 235$ m	1.51	$0 - 20$	27 m	✓	15 m
10	$22 - 66$ m	< 1 m	73	$37 - 217$ m	3.42	$0 - 13$	27 m	✗	40 m
11	$15 - 74$ m	< 1 m	49	$151 - 411$ m	0.84	$0 - 20$	38 m	✓	54 m
12	$15 - 52$ m	< 1 m	49	$182 - 497$ m	5.04	$0 - 84$	18 m	✓	22 m
13	$30 - 70$ m	< 1 m	63	$36 - 255$ m	4.18	$0 - 16$	27 m	✗	40 m
Overall	$11 - 93$ m	$0 - 24$ m	$49 - 131$	$32 - 497$ m	$0.51 - 5.04$	$0 - 84$	$15 - 38$ m	11 Yes, 2 No	$15 - 54$ m

and $\mathcal{N}(\cdot)$ is the normal distribution representing shadow fading. Comparing our data against standard PL models such as 3GPP UMa LOS and Non LOS (NLOS) [38] leads to the RMSE of 11.3 and 18.1 dB, respectively. Fig. 2(b) shows the distributions of path gain[2] for individual streets, with median spanning over a range of about 30 dB.
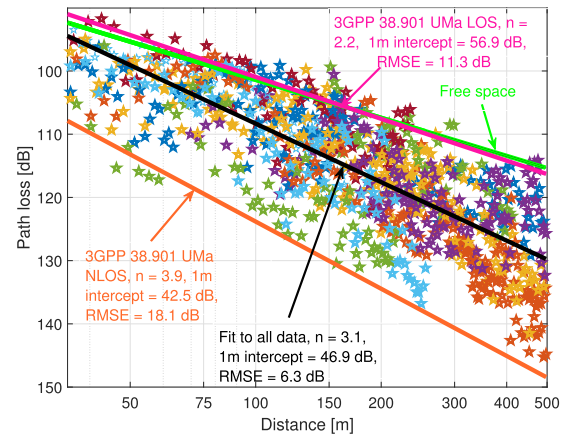
The presence of street clutter may be the cause for about 10 dB excess loss compared to the UMa LOS model in Fig. 2(a) and the large street-by-street variations in Fig. 2(b). This is due to the short wavelengths of mm-wave signals, approximately 1 cm at 28 GHz, making them more susceptible to intense diffused scattering and poorer rough surface reflection [39]. Furthermore, mm-wave has a tighter first Fresnel zone directly proportional to its wavelength, causing objects as small as tens of centimeters to appear to be substantial in impairing link quality [40]. This motivates us to capture detailed environment features, such as street clutter and 3-D building.

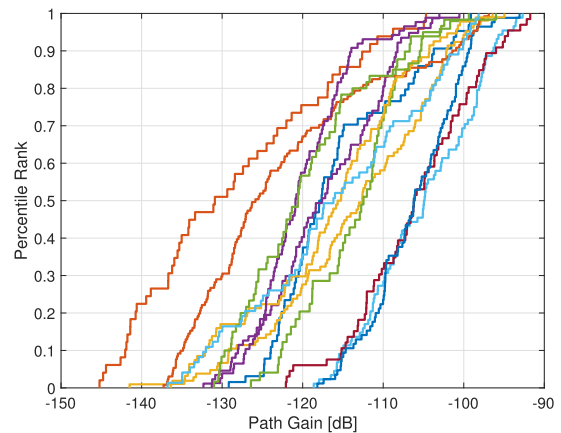### B. Street Clutter Modeling Using LiDAR Point Cloud Dataset

Let us consider the measurements done from the same rooftop for two Manhattan streets, the 7th Avenue with a handful of young trees and the W 11th Street with many tall tree canopies, as shown in Fig. 3(a) and (b), respectively. Measured PL and their slope-intercept fits are shown in Fig. 3(c). The distance exponent of the W 11th Street is significantly higher, 8.7, compared to 3.4 on the 7th Avenue, with a 23 dB gap in average PL at 500 m. Therefore, street clutter information, which includes tree canopies, cars, and lampposts, plays a crucial role in PL prediction.

To capture the street clutter, we use the USGS CMGP LiDAR point cloud repository [35], where each object is described by a set of points on its external surfaces acquired at 1 cm resolution. For each street, we change the origin to the ground location of the Rx position and align the X-axis with the street along which the Tx is moving, the Y-axis along the

(a)

(b)

Fig. 2. Manhattan measurement data and street-by-street variation (different colors per street). (a) Measured PL of 1028 links from 13 streets. (b) Large street-by-street variation.

width of the street, and the Z-axis pointing to the Rx placed at the top of the building. We also utilize a k-nearest neighbor-based point cloud denoising [41]. The processed point cloud representing the street clutter for the 7th Avenue and the W 11th Street is shown in Fig. 3(d) and (e), respectively.

---

[2]Path gain, instead of PL, is shown in Fig. 2(b) to emphasize the deteriorating link quality at lower percentile rank.
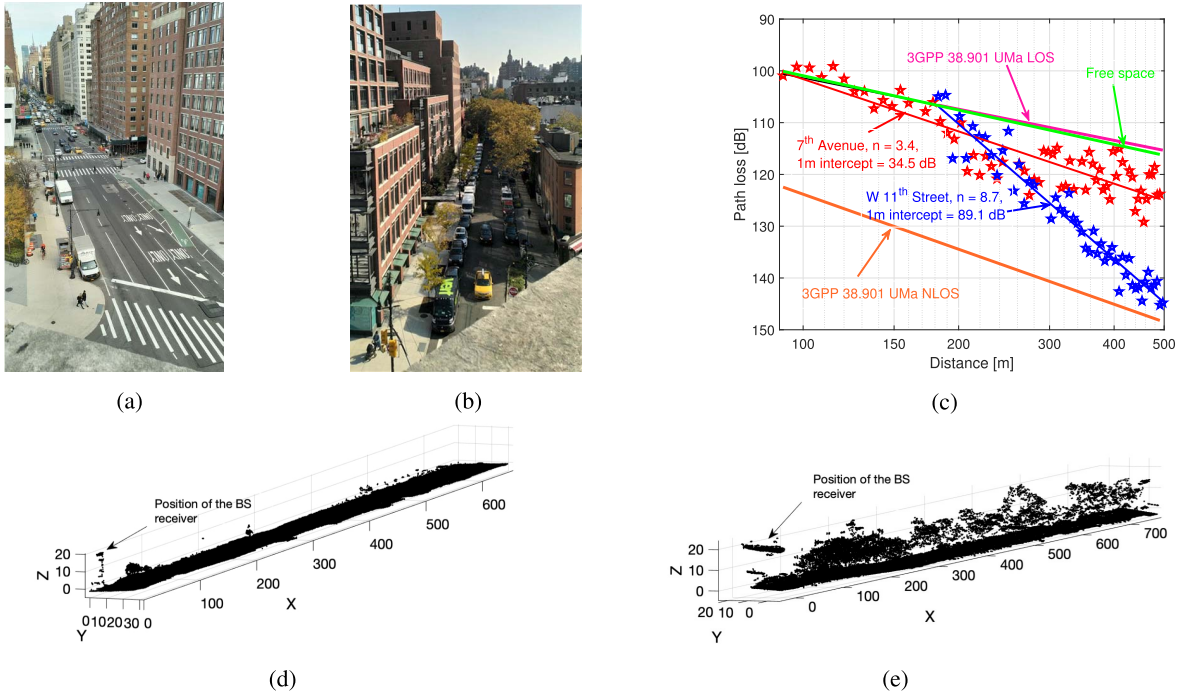
Fig. 3. Two streets covered from the same rooftop with diverse street clutter density. (a) Seventh avenue, Manhattan. (b) W 11th street, Manhattan. (c) Comparison of measured PL. (d) Point cloud for seventh. (e) Point cloud for W 11th.

Fig. 4. Aerial view of W 11th street.

C. 3-D Building Mesh Grid

Reflection and scattering from urban buildings can be significant and impose a wave-guiding effect on the signals. We extract the 3-D building mesh grid from the Cadmapper [36]. The aerial view of the W 11th Street is shown in Fig. 4 and the extracted 3-D buildings are shown in Fig. 5(a). We first convert the high-dimensional mesh grid into a Euclidean space by assigning each $1 \times 1 \times 1$ m cube a value 1/0 indicating the presence/absence of a mesh grid. We then reposition the origin such that Rx is at $(0, 0, \text{Rx height})$ and align the X-axis along the street and the Y-axis along street width (i.e., align the coordinates with those used for point cloud). To reduce the dimensionality of the dataset while preserving height information, we collapse it along the Z-axis into 2D grids (of 1×1 m) and assign the entry of each grid the

height of building at that location (0 if there is no building), as shown in Fig. 5 for the W 11th Street where the color bar indicates the height.

Remark 1: Please note that accurate point cloud and 3-D mesh-grid datasets collected along with the PL measurements are desirable, but they would significantly increase the cost and overhead. Since we are interested in capturing the general characteristics of the street clutter and 3-D buildings rather than the fine details, we find it is sufficient to use publicly available point cloud and 3-D mesh-grid datasets from [35] and [36], respectively, independent of the PL measurements [8].

III. STREET CLUTTER FEATURE EXTRACTION FROM POINT CLOUD

The LiDAR point cloud dataset [35] contains a massive amount of data that cannot be directly used for training or interpretation. Thus, we focus on expert knowledge-driven feature extraction from the point cloud dataset for modeling the street clutter.

A. Street Clutter Feature Compression

We compress all the LiDAR point cloud information for each link into two numbers using heuristic approaches devised from expert knowledge in wave propagation, where each number is proportional to the count of points in a 3-D volume.

1) *Street-Specific Clutter Density (Clutter per Street):* Averaged point cloud density in a 3-D volume defined by (along-the-street distance between Rx and furthest Tx) $\times$ (street width) $\times$ (Rx height above ground).

2) *Link-Specific Clutter Density (Clutter per Link):* Total number of points contained in a string of contiguous
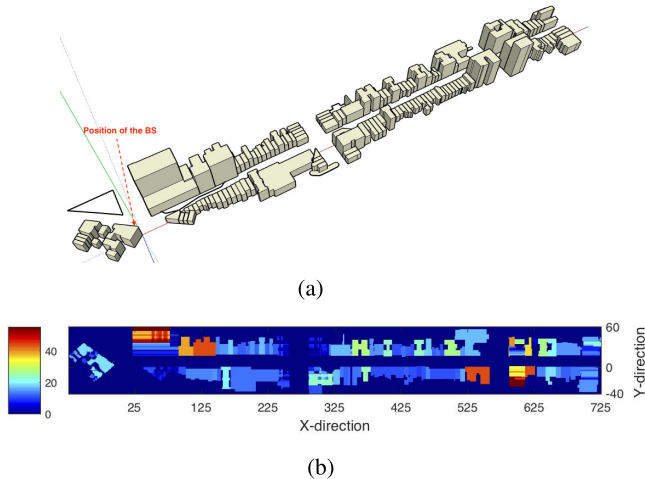
(a)

(b)

Fig. 5. Example of 3-D building mesh-grid data. (a) 3-D building (mesh grid) acquired for the W 11th street. (b) 2-D collapse for the W 11th street mesh-grid buildings.
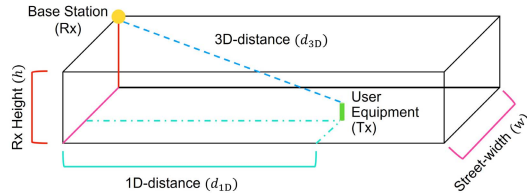

Fig. 6. Illustration of point cloud-based clutter features.

$1 \times 1 \times 1$ m cubes traversed by the straight line connecting Tx and Rx.

The clutter per link represents the accumulated clutter density along the direct path within the first Fresnel zone (about or smaller than the 1 m × 1 m cross section) and thus may be interpreted as a blockage indicator of the direct path for each link. The clutter per street represents the overall clutter density of the entire street and remains the same for all links from the same street.

B. Point Cloud-Based Expert Street Clutter Features (Clutter)

The following seven expert street clutter features are defined for PL prediction.

1) *Clutter 1 (Log 3-D Distance):* Euclidean distance (log scale) between the Tx and Rx.
2) *Clutter 2 (Log 1-D Distance):* Along-the-street distance (log-scale) between the Tx and Rx.
3) *Clutters 3 and 4 (Clutter per Street and Clutter per Link):* Clutter density information as defined in Section III-A.
4) *Clutter 5 (Street Width):* It spans from 15 to 38 m.
5) *Clutter 6 (Buildings on Both Sides):* Indication of guiding effect from street canyon.
6) *Clutter 7 (Rx Height):* It spans from 15 to 54 m.

Among the defined seven clutter features, four of them have been used in different 3GPP models: 1-D distance (d_{1D}), 3-D distance (d_{3D}), street width, and the BS or Rx height (h), as shown in Fig. 6. Such features provide us a way to interpret the trained models and compare them against the 3GPP and slope-intercept models. The other three features capture street-specific (clutter per street) and link-specific (clutter per link)

clutter information as well as the street canyon information (building on both sides of the street).

We summarize the defined expert features in Table II, in which we omit the details of log 3-D distance, log 1-D distance, and clutter per link features because they have a separate entry for each link. We also report the range of building height, terrain variation, and the number of measurement links on each street. Please note that the ten-time variation in the clutter per street feature (0.51–5.04) indicates the diverse nature of street clutter in the measurement streets, consistent with the 30 dB street-by-street variation seen in Fig. 2(b).

We standardize the defined clutter features before training and testing. For feature f, we compute its mean μ and variance σ^2 on the training dataset; then, we rescale the feature in both the training and testing datasets as $\hat{f} = (f - \mu)/\sigma$.

IV. AE-BASED FEATURE EXTRACTION FROM BUILDING DATASET

Although we have converted the 3D building mesh grid into a 2-D matrix representation, as explained in Section II-C, the building features are still much richer than the PL data. We further reduce the 2-D collapse of buildings to a size of $(500, 40)$ by removing buildings beyond the maximum measurement distance of 500 m and by including the building facade only (taking grids 20 m from the center of the street on each side). As the Tx–Rx distance increases, the number of 2-D grids in between the two also increases. We append zero to the 2-D matrix representation at shorter distances to preserve the distance dependency and avoid information loss by downsampling.

We then use CNN-based AE to compress the extensive building facade data to a few most relevant features for PL prediction. CNN captures the spatial dependencies with the help of kernels and filters, and AE learns an efficient encoder (for feature compression) and a matching decoder (to reproduce the original input signal) in an unsupervised manner. Therefore, a combination of CNN and AE, as shown in Fig. 7, can help us to reduce the massive feature dimension of the building data to 12 features while preserving the spatial characteristics.

The encoder in the CNN-based AE, as shown in Fig. 8(a), takes 2-D buildings $\mathbf{I} \in \mathbb{R}^{(500,40)}$ as input and reduces it to compressed features $\mathbf{X} \in \mathbb{R}^{(12)}$. This is achieved by the convolutional and max-pooling layers that help to reduce the dimension while preserving the spatial characteristics, whereas the dense layers extract compressed nonlinear features. After the first max-pooling layer, we perform grouped convolutions—convolutions in parallel, in which two identical CNNs (Conv-Net-1 and Conv-Net-2) are processed in parallel before their addition.

Then, the decoder in the CNN-based AE, as shown in Fig. 8(b), takes $\mathbf{X}$ as its input to reconstruct the 2-D buildings $\mathbf{Y} \in \mathbb{R}^{(500,40)}$. This is achieved by the dense layers that decode information from the compressed feature representation and the convolutional and upsampling layers that revert the building information to its original form.

We design a loss function $\mathcal{L}(\cdot)$ based on the log-cosh loss instead of MSE to increase its robustness against outliers in
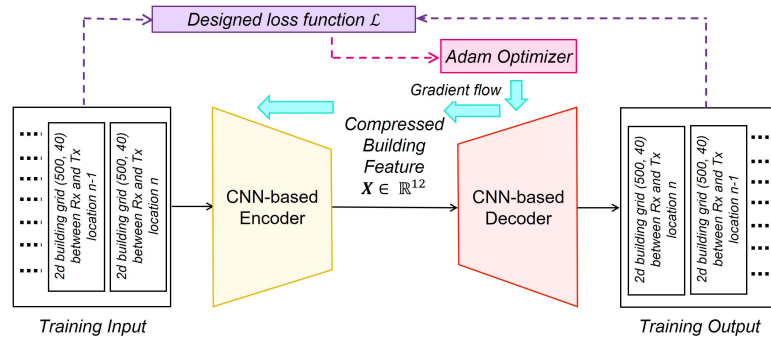
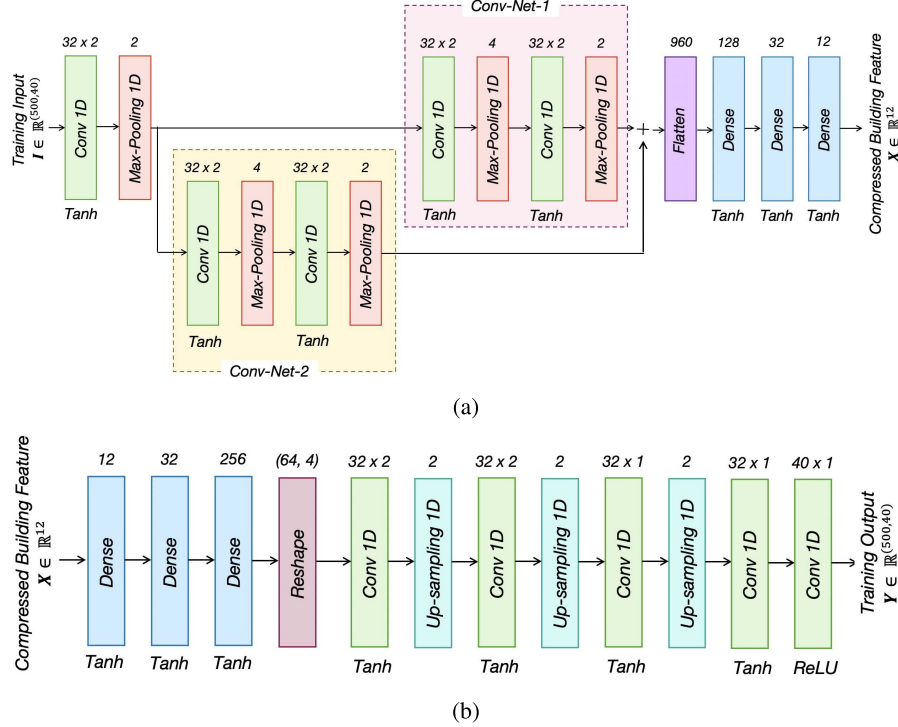Fig. 7. Proposed CNN-based AE for building feature compression.

Fig. 8. Neural network architectures for the proposed CNN-based AE. (a) Architecture of CNN-based encoder of AE. (b) Architecture of CNN-based decoder of AE.

the building data and also reduces the impact of appended zeros in the input **I**. Note that the values in 2-D matrix representation is standardized between 0 and 1 before feeding to the AE. For a grid (a, b), we find the maximum and minimum values for that grid in all the training data and denote it by $\max(a, b)$ and $\min(a, b)$. Then, we rescale all the values in the training and testing data for the $(a, b)^{\text{th}}$ grid as (value in (a,b) $- \min(a, b))/(\max(a, b) - \min(a, b))$. The details of the designed loss function $\mathcal{L}(\cdot)$ and the CNN-based AE architecture can be found in Appendix A.

We implement the AE in Keras [46] with TensorFlow as a backend. We keep the learning rate of 0.0012, the batch size of 16, and the total number of epochs 100. We train the network end-to-end using the Adam optimizer [47] over the time, to reconstruct the input 2-D collapse of the building at the output of the decoder. Once the AE is converged,[3]

---

[3]The designed AE with parallel Conv-nets (CNNs) converges after 50 epochs and has better reproducibility than using a single or two serially concatenated Conv-nets, shown in Appendix B.

we utilize the encoder to design compressed building features $\mathbf{X} \in \mathbb{R}^{(12)}$ and then feed them to PL prediction.

V. ML-BASED MODELS AND TRAINING–TESTING METHODOLOGY

We utilize the extracted clutter and compressed building features and compare the following regularized linear and nonlinear ML algorithms [45] for PL prediction using a street-by-street training and testing methodology to emphasize generalizability.

A. ML-Based Models for PL Prediction

Let P denote the true PL value, $\mathcal{F}_{(\cdot)}$ represent the input feature vector, and w indicate the designed weight matrix. The following ML algorithms are used for PL prediction.

1) *Lasso Regression (Lasso):* It optimizes the regression weights by minimizing the least-square error, including

---

Algorithm 1 Links-Shuffle-Split Training and Testing Procedure

---

Require: $\mathcal{F}_{(\cdot)}$ and $\mathbf{P}$ for 1028 PL measurements.   ▷ Collection of all the measurements from 13 streets.
Ensure: $\mathcal{D} = [\mathbf{x} := \mathcal{F}_{(\cdot)}, \mathbf{y} := \mathbf{P}]$.   ▷ This symbolizes our total dataset.
1: **for** $i = 1$ **to** *iterations* **do**   ▷ Loop for varying shuffle and split of dataset.
2:     Randomly shuffle and split $\mathcal{D}$ in $4:1$ ratio to form $[\mathbf{x}_{\text{train}}^{i}, \mathbf{y}_{\text{train}}^{i}]$ and $[\mathbf{x}_{\text{test}}^{i}, \mathbf{y}_{\text{test}}^{i}]$.
3:     Train ML-based PL models (detailed in Section V-A), using $[\mathbf{x}_{\text{train}}^{i}, \mathbf{y}_{\text{train}}^{i}]$ to obtain trained model $\mathcal{M}^{i}$.
4:     Test model $\mathcal{M}^{i}$ using $\mathbf{x}_{\text{test}}^{i}$ to predict PL $\mathbf{y}_{\text{pred}}^{i}$.
5:     Calculate $RMSE^{i}$ in PL between ground-truth PL $\mathbf{y}_{\text{test}}^{i}$ and predicted PL $\mathbf{y}_{\text{pred}}^{i}$.
6: **end for**
7: Mean RMSE $:= \mu(RMSE^{\{1,2,\ldots,iterations\}})$ and std. deviation $:= \sigma(RMSE^{\{1,2,\ldots,iterations\}})$.

---

a supplementary l_1-norm penalty on the regression coefficients (weights)

$$\min_{w}\left(1/2n_{\text{samples}}\right) \times \left\|\mathcal{F}_{(\cdot)}w - P\right\|_2^2 + \alpha\|w\|_1 \qquad (2)$$

where n_{samples} is the sample size and $\alpha > 0$ imposes the l_1 penalty on the weights.

2) *Elastic-Net Regression (Elastic-net):* It imposes both l_1- and l_2-norm penalties on the weights, where convex combination of l_1 and l_2 penalties is controlled by parameter δ

$$\min_{w}\left(1/2n_{\text{samples}}\right) \times \left\|\mathcal{F}_{(\cdot)}w - P\right\|_2^2 + \alpha\delta\|w\|_1$$
$$+ (\alpha(1-\delta)/2) \times \|w\|_2^2. \qquad (3)$$

3) *RF:* It is an ensemble learning method, where multiple decision trees' average is utilized to predict the PL. We consider 20 estimators with a maximum tree depth of 25.

4) *SVR:* It solves the following primal problem:

$$\min_{w,b,\zeta} \ \left(w^T w\right)/2 + C\sum_{n=1}^{T}\zeta_n$$
$$\text{s.t.} \ P_n\left(w^T\phi\left(\mathcal{F}_{(\cdot)_n}\right) + b\right) \geq 1 - \zeta_n$$
$$\zeta_n \geq 0, \quad \forall n \in [1, T] \qquad (4)$$

where C denotes the penalty term, ζ_n indicates the distance of the nth sample from the decision boundary, and b represents the bias term and $\phi(\mathcal{F}_{(\cdot)_n})$ maps $\mathcal{F}_{(\cdot)_n}$ to a higher dimensional space. We train the SVR with an radial bias function (RBF) kernel, given by $\exp(-\gamma \|\mathcal{F}_{(\cdot)_i} - \mathcal{F}_{(\cdot)_j}\|^2)$ for any two samples i and j, and $\gamma > 0$.

All the PL prediction methods are implemented using scikit-learn [48]. We use grid search with fivefold cross validation [45] over the training set to obtain the best parameters. In particular, the hyperparameter α in Lasso and Elastic-net and C, γ in SVR is grid-searched from $\{10^{-4}, 10^{-3}, \ldots, 10^3, 10^4\}$ during the training and the best fit parameter is used for testing.

B. Feature Vectors

The proposed ML-based PL prediction models can be implemented using either of the following descriptive features.

1) Point Cloud-Based Expert Street Clutter Features (Clutter) Only: It consists of the seven expert features extracted from the street clutter information (in Section III-B), which is represented as

$$\mathcal{F}_{\text{Clutter}} = \{\text{log-3-D distance, log-1-D distance, street width,}$$
$$\text{Clutter-per-link, Clutter-per-street, Rx height,}$$
$$\text{buildings on both sides}\}. \qquad (5)$$

2) Combination of Clutter and Building Features (Clutter + Building): Here, we first use the encoder of the trained AE (proposed in Section IV) to obtain compressed features $\mathbf{X} \in \mathbb{R}^{(12)}$ and concatenate with the clutter in (5), denoted by $\mathcal{F}_{\text{Clutter\_Building}} = [\mathcal{F}_{\text{Clutter}}, \mathbf{X}]$.

C. Training and Testing Methodology

In conventional ML-based training–testing, the 1028 PL measurements (collection of all the measurements from 13 streets) dataset is randomly shuffled and divided into a 4:1 ratio for training and testing sets. We refer to it as links-shuffle-split training and testing, briefly described in Algorithm 1. To capture the impact of random shuffling and splitting, we perform the process multiple times and obtain the mean RMSE and standard deviation in RMSE due to random shuffling and splitting of the dataset. Data in the testing set are statistically similar to those in the training set, and the focus of trained models is on interpolation. Since links that are close to each other have similar PL values, shuffling the data impacts negatively on the generalizability of the model given the limited amount of PL measurements.

Motivated by the large street-by-street variation of measured PL observed from Manhattan measurements [8], we propose a new way of training–testing referred to as street-by-street training and testing, focusing on the extrapolation capabilities. We group the measurement links based on streets where they are collected and formulate 13 groups, one for each street. We then create 13 train–test combinations, in which for each combination, one street is selected for testing and the rest for training. A model is trained and tested 13 times, using the 13 train–test combinations independently, producing 13 RMSE values. We summarize the procedure briefly in Algorithm 2.

Remark 2: In Algorithm 2, the street-by-street variation in prediction is quantified by the standard deviation in RMSE of the 13 tested streets. It is the metric chosen to measure the generalizability to unseen streets. Thus, the lower the standard deviation, the better the generalizability.

Algorithm 2 Street-by-Street Training and Testing Procedure

Require: $\mathcal{F}_{(\cdot)}^{\{1,\dots,13\}}$ and $\mathbf{P}^{\{1,\dots,13\}}$ for each of the 13 streets.

1: **for** $k = 1$ **to** 13 **do**　　　　　　　　　　　　▷ Loop for each of the 13 streets as testing.
2:　　Testing Street := k^{th} street, Training streets := All the 13 streets **except** k^{th} street := $\{1,\dots,13\} - \{k\}$.
3:　　$\mathbf{x}_{\text{test}} := \mathcal{F}_{(\cdot)}^{k}$, $\mathbf{x}_{\text{train}} := \mathcal{F}_{(\cdot)}^{\{1,\dots,13\}-\{k\}}$, $\mathbf{y}_{\text{test}} := \mathbf{P}^k$ and $\mathbf{y}_{\text{train}} := \mathbf{P}^{\{1,\dots,13\}-\{k\}}$ ▷ Measurements of kth street becomes testing data, while all the other measurements form training data.
4:　　Train ML-based PL models (detailed in Section V-A), using $[\mathbf{x}_{\text{train}}, \mathbf{y}_{\text{train}}]$ to obtain trained model $\mathcal{M}^k$.
5:　　Test model $\mathcal{M}^k$ using $\mathbf{x}_{\text{test}}$ to predict PL $\mathbf{y}_{\text{pred}}$.
6:　　Calculate $RMSE^k$ in PL between ground-truth PL $\mathbf{y}_{\text{test}}$ and predicted PL $\mathbf{y}_{\text{pred}}$ for k^{th} testing street.
7: **end for**
8: Mean RMSE := $\mu(RMSE^{\{1,2,\dots,13\}})$ and std. deviation := $\sigma(RMSE^{\{1,2,\dots,13\}})$.

TABLE III

RMSE IN PL PREDICTION

Prediction method	Street-by-street training-testing (mean ± standard deviation) over 13 streets			Links-shuffle-split training-testing (mean ± standard deviation) over 25 shuffles		
3GPP UMi NLOS	18.0 ± 4.1 dB			18.1 ± 0.4 dB		
3GPP UMa LOS	10.6 ± 4.4 dB			11.3 ± 0.4 dB		
Slope-intercept	6.5 ± 2.0 dB			6.1 ± 0.3 dB		
ML Algorithm	**Clutter**	**Clutter + Building**		**Clutter**	**Clutter + Building**	
		Average over 25 AE runs	*Best* out of 25 AE runs	*Average* over 25 shuffles	*Average* over 25 AE runs	*Best* out of 25 AE runs
RF	6.6 ± 1.8 dB	6.9 ± 1.9 dB	5.8 ± 1.5 dB	**4.1 ± 0.2 dB**	**4.3 ± 0.2 dB**	**4.0 dB**
SVR	5.4 ± 1.4 dB	5.8 ± 1.4 dB	4.8 ± 1.1 dB	4.4 ± 0.2 dB	4.4 ± 0.2 dB	4.0 dB
Lasso	5.7 ± 1.5 dB	5.7 ± 1.4 dB	4.8 ± 1.0 dB	5.0 ± 0.2 dB	4.8 ± 0.2 dB	4.4 dB
Elastic-net	**5.4 ± 1.3 dB**	**5.7 ± 1.4 dB**	**4.8 ± 1.1 dB**	5.0 ± 0.2 dB	4.8 ± 0.2 dB	4.4 dB

D. Model Applicability and Generalizability

The measurement campaign [8] focused on rooftop-to-same street measurements in urban street canyons. Thus, the trained ML-based PL prediction models proposed in this work are applicable to similar urban street canyons, where BS is placed on the rooftop and UEs move along the same street. The direct path between them would have been in LOS if there were no street clutter/foliage.

The generalizability of the trained model to unseen streets depends on the similarity of street canyons and street clutter density. We have introduced three measures (see Section I-B) to enhance the "generalizability" of trained ML models, namely, street-by-street training and testing to force extrapolation, aggressive feature compression to improve robustness, and simple prediction models to reduce overfitting.

The applicability and generalizability of the trained ML models to new urban street canyons can be assessed from three aspects.

1) *Ranges of Features:* The trained ML model is expected to work best/well when the ranges of the features in unseen streets are within or close to the ranges of the training features [52], as summarized in Table II.

2) *Feature Importance and Sensitivity:* Not all features are equally important in PL prediction and not all features of unseen streets have a similar range as training data. Thus, it becomes pivotal to analyze feature importance

and feature sensitivity [53]. We defer the analysis to Section VI-D and Appendix C.

3) *Distribution of Measured PL:* When sparse PL measurements are available from target streets, we can check whether the measured PL distribution is similar to the distribution of training PL samples, as shown in Fig. 2(b). The trained ML model is expected to work best/well only if the distributions are similar.

Remark 3: If the feature ranges of future streets and/or the distribution of measured PL do not align well with the existing training datasets, we can employ transfer learning [54], [55] to fine-tune the existing ML model with very few measurements, instead of training the new ML model from scratch by leveraging the trained ML model.

VI. PERFORMANCE EVALUATION AND ANALYSIS

In this section, we evaluate the performance of the proposed PL prediction models using the street-by-street training and testing. The key performance metric is the mean and standard deviation of the 13 RMSE values obtained in street-by-street PL prediction. Our benchmarks are the 3GPP UMa LOS prediction model ($P = 28.0 + 22 \log_{10} d_{\text{3D}} + 20 \log_{10} f_c$), 3GPP UMi NLOS prediction model ($P = 22.4 + 35.3 \log_{10} d_{\text{3D}} + 21.3 \log_{10} f_c$), where $f_c = 28$ denotes carrier frequency (in GHz), as well as the slope-intercept model described in (1) where the slope and intercept parameters are obtained using the same training data subsets as used by the ML-based methods. All of the three models only use the 3-D Tx–Rx Euclidean
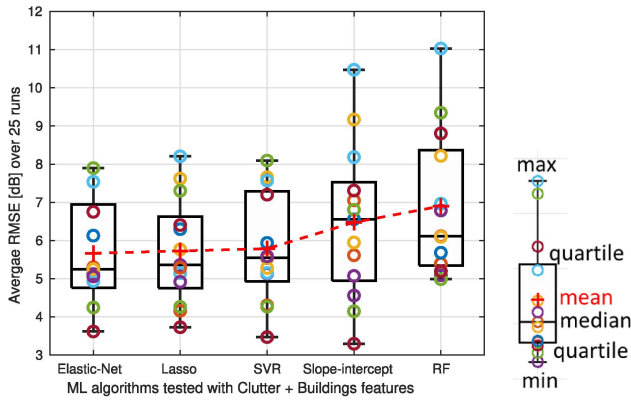
Fig. 9. Street-by-street variation of the average RMSE over 25 runs using Clutter + Building.

distance d_{3D} as the input feature and their performances are evaluated using the same testing data subsets as used by ML-based methods.

Simple ray-tracing-based PL prediction, which includes up to ten reflections from the wall and the ground but ignores street clutter, has predicted stronger signal power than in free space [4], [8], which itself is 10 dB hotter than the measured data in urban canyons [8]. Several works [32]–[34] have focused on (3-D) ray-tracing-based prediction for mm-wave in urban street canyons, where the accuracy of ray tracing is shown to be strongly dependent on modeled wave propagation mechanisms [33] and environmental feature description [34]. Therefore, an in-depth proper investigation of street clutter approximation and modeling is needed to bring ray-tracing prediction up to a reasonable level of accuracy. We leave improvements of ray-tracing-based PL prediction for future work.

A. PL Prediction Accuracy (RMSE)

We summarize in Table III the RMSE in PL prediction of linear and nonlinear ML algorithms proposed in Section V-A with street-by-street training–testing methodology (shown in Algorithm 2), where the standard deviation of RMSE represents robustness against street-by-street variation over all 13 training–testing combinations.

The 3GPP UMi NLOS and 3GPP UMa LOS channel models have the mean RMSE of 18.0 and 10.6 dB, respectively, not suitable for describing street canyon channels with clutters. The slope-intercept model produces a mean RMSE of 6.5 dB with a standard deviation of 2.0 dB across different testing streets. With the clutter feature set ($\mathcal{F}_{Clutter}$), both regularized linear Elastic-net model and the nonlinear SVR model simultaneously reduce the mean RMSE by about 1.1 dB and the street-by-street standard deviation by about 0.7 dB, creating a more generalizable model with better PL prediction accuracy.

The PL prediction performance can be further improved using the $\mathcal{F}_{Clutter\_Building}$ feature set. Unlike the clutter feature set that is deterministic, the $\mathcal{F}_{Clutter\_Building}$ feature set contains compressed building features extracted from a CNN-based AE, which is inherently a random process and the resulting performance can change significantly [51]. We run the AE 25 times and generate 25 unique $\mathcal{F}_{Clutter\_Building}$ feature sets.

For each of the 25 feature sets, we test the ML algorithms using the street-by-street testing, reporting both the best[4] and the average over all 25 runs. The street-by-street variation of the average RMSE over all 25 runs is shown in Fig. 9 using a box plot, where the average RMSE over 25 runs for each testing street is represented as a color-coded *o* symbol. The median and mean over 13 streets are given by line inside each box and the red + symbol, respectively, and the edges of the box mark the quartiles, with whiskers extending outside the box indicating the minimum and maximum over all 13 testing streets. By searching for a better AE out of 25 runs for each testing street, the mean RMSE can be further reduced by about 0.6 dB and street-by-street standard deviation by about 0.2 dB for both Elastic-net and SVR. The gain of RF-based prediction over the slope-intercept model is small, which is likely because it is not good at extrapolation when the statistics of the training and testing sets differ [50].

To verify and compare the capability of interpolation of various prediction models, we also run the classical links-shuffle-split training and testing approach detailed in Algorithm 1 with iterations = 25, i.e., randomly shuffle-and-split 25 times. The results are also shown in Table III, where the 0.2 dB standard deviation in RMSE for Clutter + Building comes from both the inherent randomness of shuffling and splitting the dataset and AE compression out of 25 independent runs. The best result out of 25 runs for all the proposed ML-based PL prediction achieved over 1.7 dB reduction in mean RMSE compared to the slope-intercept prediction and over 0.4 dB reduction compared to the street-by-street testing. However, caution has to be taken to differentiate extrapolation and interpolation for site-specific PL prediction given limited measurement datasets.

B. Robustness Against Street-by-Street Variation and Distance

We evaluate the robustness of PL prediction against street-by-street variation in Fig. 10 using Elastic-net-based prediction for both the clutter feature set and the best Clutter + Building ($\mathcal{F}_{Clutter\_Building}$) feature set out of the 25 options. Compared to the two 3GPP models and the classical slope-intercept model, the two ML-based PL prediction models reduce both the mean RMSE and the street-by-street variability. This clearly demonstrates the importance of incorporating street-specific features, such as street clutter and building into PL prediction models and the robustness of linear ML-based models in extrapolation to unseen streets.

The mean RMSE in PL prediction as a function of the 3-D distance is shown in Fig. 11. All the links are grouped into 100 m intervals, and within each interval, we calculate the mean RMSE for street-by-street testing. Here, the *X*-axis denotes the ending position of an interval (e.g., 200 m denotes the interval spanning from 100 to 200 m). Our proposed models outperform the slope-intercept model for all the distance ranges. The improvement in mean RMSE is about 1.2–2.7 dB

---

[4]Both the mean and standard deviation of the best RMSE converge within 25 runs as shown in Appendix B.
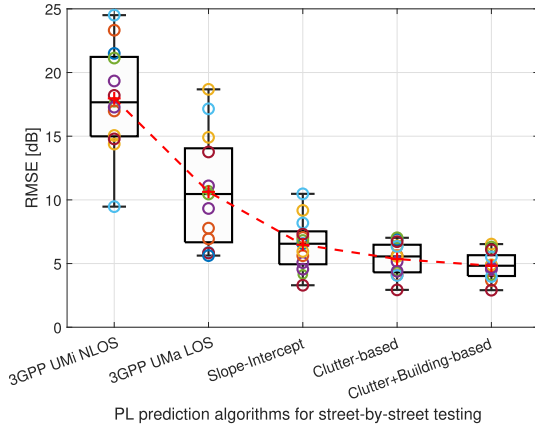
Fig. 10.   Street-by-street variation.

using Clutter + Building feature set compared to the classical slope-intercept model.

C. PL Prediction Versus Measurement

We compare the measured PL values with the predicted PL values in Fig. 12. We analyze the street-by-street testing using Elastic-net prediction (regularized linear regression) with *Clutter + Building* features for the best case (smallest RMSE, Street 1) and the worst case (largest RMSE, Street 13) in Fig. 12(a) and (b), respectively. The linear model derived from the Elastic-net prediction (red solid line) is very close to the slope-intercept fit to measured data (black solid line), with marginally increased RMSE ranging between 0.2 and 0.7 dB. This indicates that our proposed Elastic-net is highly generalizable and effective in capturing street-by-street variation. While the trained model is good at capturing the overall PL trend per street, its capability of tracking link-by-link variation is modest, as shown by the marginal improvement (0.1–0.2 dB in RMSE) over the linear models derived from the corresponding prediction. In Fig. 12(c), we also analyze the links-shuffle-split testing over 20% of measured data from all 13 streets using the trained RF with clutter features. This shows that our proposed RF (nonlinear regression) model is highly effective in generalizing to nearby links.

D. Feature Importance for the Point Cloud-Based Street Clutter Features

The Lasso regression imposes an l_1-norm penalty that minimizes the weights of least relevant features to improve the accuracy, and therefore, the relative magnitude of the resulting weights can be interpreted as the feature importance in PL prediction. We perform the Lasso regression on the clutter feature set with street-by-street training–testing methodology and present the obtained weights in Fig. 13, where each bar represents the mean value of the weights corresponding to each feature. The error bar indicates the minimum and maximum weight values obtained for that specific feature when tested separately for 13 streets. The amplitude of each weight indicates the importance of that feature, and the opposite signs (in amplitude) of the weights help balance the clutter features in PL prediction. As expected, the 3-D distance has the highest weight, followed by the clutter per street, which quantifies how
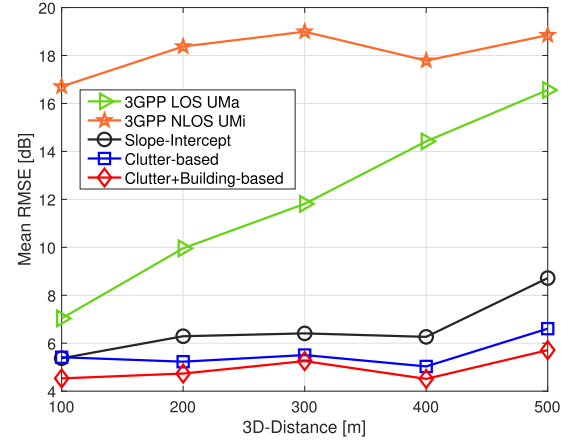

Fig. 11.   Mean RMSE versus distance.

TABLE IV
TRAINING AND TESTING COMPLEXITY OF PL PREDICTION ALGORITHMS

Algorithm	Training Complexity	Testing Complexity
3GPP	−	$\mathcal{O}(1)$
Slope-intercept	$\mathcal{O}(n)$	
Lasso	$\mathcal{O}\left(Tp^2n + p^3\right)$	$\mathcal{O}(p)$
Elastic-Net	$\mathcal{O}\left(Tp^2n + p^3\right)$	
Random forest	$\mathcal{O}\left(n\log(n)pn_tn_d\right)$	$\mathcal{O}(n_tn_d)$
Support Vector	$\mathcal{O}\left(n^2p + n^3\right)$	$\mathcal{O}(n_{sv}p)$

cluttered each street is based on the normalized point cloud densities of the whole street. The "building on both sides" binary indicator and the clutter per link, which quantify accumulated clutter density along the direct path, also have notable importance. This aligns with the understanding that reflection from buildings on both sides of the canyon increases signal strength, and that clutter intruded into the direct path has an adversarial effect on propagation.

Feature importance can also be observed by comparing the change of prediction accuracy when each clutter feature is excluded individually from prediction, as shown in Appendix C using Elastic-net regression, where the same top four most important features are identified (in descending order): clutter per street, 3-D distance, clutter per link, and building on both sides.

By only using the top four most influential features, the RMSE of 5.5 ± 1.1 dB can be achieved using Elastic-net regression. For almost all the ML-based prediction results using the reduced clutter feature set, both the mean RMSE and standard deviations are within 0.2 dB from the results shown in Table III obtained using all the seven clutter features (see Appendix C for details).

E. Computational Complexity

Let us denote the number of training samples by n, the number of features by p, the number of trees by n_t and depth of the tree by n_d, and the number of support vectors by n_{sv}; the training and testing complexity can be summarized in Table IV where T is the number of outer iterations used in the coordinate descent solver. Given the low dimensionality of the features (7 in clutter and 12 in building) used in our PL prediction models, the PL prediction is very fast. The prediction time per link is less than 3 μs for Lasso
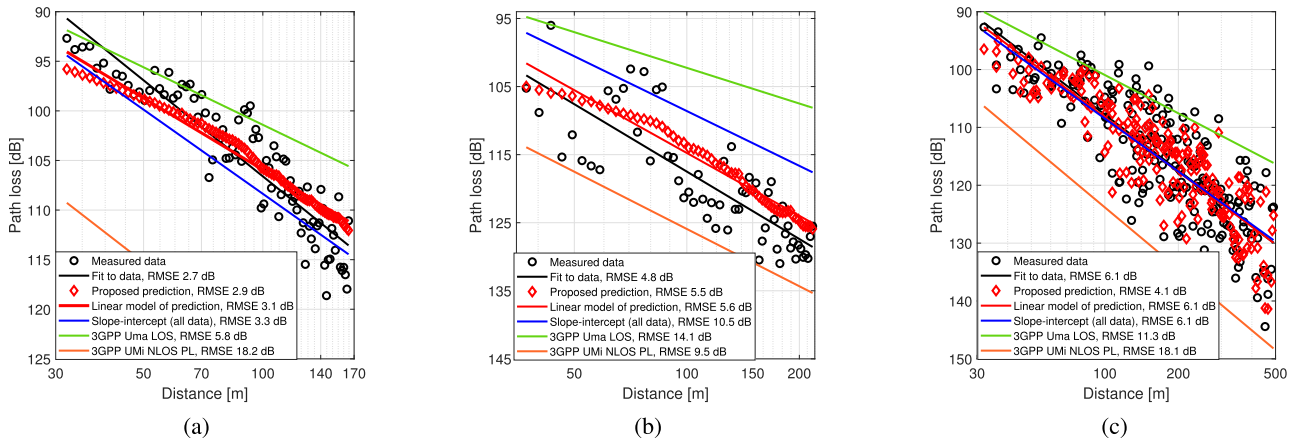
Fig. 12. Measured PL data versus prediction PL data. For the street-by-street testing in (a) and (b), we employ Elastic-net on Clutter + Building features. For the links-shuffle-split testing in (c), we employ RF on clutter features. (a) Street by street—best case (Street 1). (b) Street by street—worst case (Street 13). (c) Link shuffle split with 80%–20% split.
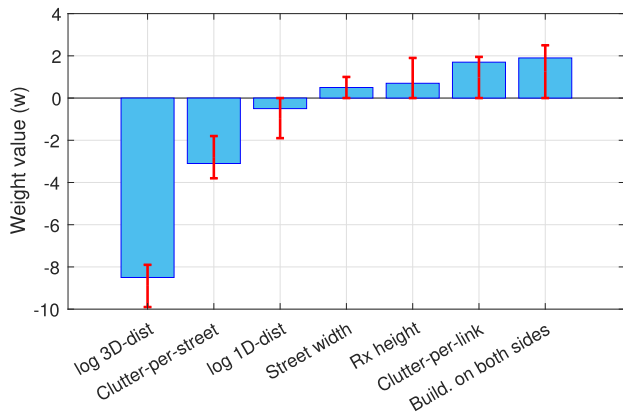

Fig. 13. Average feature importance for clutter feature set using the Lasso regression in street-by-street testing, where the error bars indicate the minimum–maximum range over 13 streets.

and Elastic-net and less than 40 μs for RF and SVR when implemented using scikit-learn [48] running with Intel Core i7–6700 CPU with 64 GB RAM and Ubuntu 18.04.4 LTS OS.

VII. CONCLUSION

We have proposed an ML-based PL prediction model for urban street canyon using the 28 GHz measurement data collected from Manhattan. The feature set contains street clutter obtained from LiDAR point cloud and buildings from a 3-D mesh grid. The PL dataset has 1028 PL measurement links from 13 streets. Although the PL dataset is massive for the classical slope-intercept PL modeling, it is small for ML-based approaches when compared to massive point cloud and 3-D building feature sets and parameters in AE and learning algorithms. To mitigate the risk of overfitting, we defined seven expert features with physical meaning from the point cloud. We also compressed 3-D building features to a length-12 vector for each link using CNN-based AE. Instead of interpolation to nearby links, we focused on the extrapolation by introducing a street-by-street training and testing approach. Using linear ML algorithms for PL prediction, we achieved the RMSE of 4.8 ± 1.1 dB compared to 10.6 ± 4.4 and 6.5 ± 2.0 dB for 3GPP LOS and slope-intercept prediction, respectively, which demonstrates the superior capability of our model in extrapolation.

Intuitive interpretation of feature importance was obtained using the Lasso regression-based analysis and feature exclusion analysis. By only using the top four most influential features, namely, distance, street clutter density (clutter per street and clutter per link), and street canyon indication (building on both sides), the RMSE of 5.5 ± 1.1 dB can be achieved using Elastic-net regression.

Among the four learning algorithms used in this article, the nonlinear RF regression has achieved the worst prediction performance under street-by-street testing but is the best under links-shuffle-split testing. This may be attributed to the reduced similarity in statistics between training and testing sets in street-by-street testing compared to the links-shuffle-split testing, and the lack of extrapolation capabilities of RF regressions. Regularized linear Elastic-net regression has the best performance, which is in line with the intuition that regularization on linear algorithms is more robust against overfitting with limited training dataset. The nonlinear SVR with RBF kernel performs well (second best) for both links-shuffle-split testing and street-by-street testing, at the cost of higher complexity.

APPENDIX A
CNN-BASED AE DESIGN TO COMPRESS BUILDING FEATURES

We design CNN-based AE for feature extraction from the building dataset, as shown in Fig. 7, where an encoder $e(\cdot)$ compresses the input 2-D building collapse $\mathbf{I} \in \mathbb{R}^{(500,40)}$ to an representation $\mathbf{X} \in \mathbb{R}^{(12)}$, which is then fed to the decoder $d(\cdot)$ to reconstruct original input 2-D building collapse $\mathbf{I}$. We briefly describe next the network layers shown in Fig. 8.

1) *1-D Convolutional Layer (Conv 1-D):* It employs various kernels to convolve the 2-D image, preserving the spatial characteristics of the input image while extracting relevant features.
2) *1-D Max-Pooling Layer (Max-Pooling 1-D):* Pooling is a sample-based discretization process utilized to downsample the input image by making assumptions in the binned subregion. In max pooling, we take the maximum value in the binned subregion.

3) *1-D Upsampling Layer (Up-Sampling 1-D):* Upsampling layer has no weights, which helps increase the input dimensions when followed by a convolutional layer.

4) *Fully Connected Layer (Dense):* The nonlinear processing is performed via dense layers in which each neuron is fully connected to all the neurons in the previous layer.

Please note that 1-D in the above layers means that we have kernels (in Conv layer) and factor (for upsampling and downsampling) [45] in only one dimension.

A. Designing the Encoder

The input to our network is 2-D collapse of buildings given by $\mathbf{I} \in \mathbb{R}^{(500,40)}$, which is normalized between 0 and 1. Let $e(\mathbf{I}|\Theta_e)$ be the mapping from the input buildings to compressed representation when the parametric transformation of encoder is given by Θ_e, which denotes the weight, filters, and bias terms. Thus, the encoder can be denoted as

$$\mathbf{X} = e(\mathbf{I}|\Theta_e) = e_{L_e}\big(\ldots\big(e_2\big(e_1\big(\mathbf{I}|\Theta_{e_1}\big)|\Theta_{e_2}\big)\ldots|\Theta_{e_{L_e}}\big) \quad (6)$$

where L_e denotes the number of layers in the encoder.

For the convolutional layers, the operation of the lth layer can be represented as follows:

$$\mathbf{X}_l = e_l\big(\mathbf{I}_l|\Theta_{e_l}\big) = h(\mathbf{W}_l \otimes \mathbf{I}_l + b_l), \quad \Theta_{e_l} = [\mathbf{W}_l, b_l] \quad (7)$$

where $\otimes$ indicates the convolutional process, $\mathbf{W}_l$ represents the 1-D kernels used for feature extraction, b_l denotes the bias vector, $h(\cdot)$ is the activation function, $\mathbf{I}_l = \mathbf{X}_{l-1}$ comes from layer concatenation, and $\mathbf{I}_1$ equals the 2-D building matrix $\mathbf{I}$.

We apply several Max-Pooling 1-D layers in between for improving the region covered by the following receptive fields. Moreover, as shown in Fig. 8(a), we introduce grouped convolutions in the encoder, in which we take the output of the first Max-Pooling 1-D and make two branches of it, with separate Conv 1-D and Max-Pooling 1-D layers, and then add the output of both branches (inspired by AlexNet [45]). The convolutional layer's output is then flattened to $\mathbf{K}$ and used as input of several stacked dense layers, where the first dense layer in the encoder can be given by

$$\mathbf{X}_l = e_l\big(\mathbf{I}_l|\Theta_{e_l}\big) = h(\mathbf{W}_l \mathbf{K} + b_l), \quad \Theta_{e_l} = [\mathbf{W}_l, b_l]. \quad (8)$$

B. Designing the Decoder

The input to our decoder is the output of the encoder given by $\mathbf{X} \in \mathbb{R}^{(12)}$. Let $d(\mathbf{X}|\Theta_d)$ be the mapping from the compressed representation to input buildings when the parametric transformation of decoder is given by Θ_d, which denotes its weight, filters, and bias terms. The decoder can be represented as follows:

$$\mathbf{Y} = d(\mathbf{X}|\Theta_d) = d_{L_d}\big(\ldots\big(d_2\big(d_1\big(\mathbf{X}|\Theta_{d_1}\big)|\Theta_{d_2}\big)|\Theta_{d_{L_d}}\big) \quad (9)$$

where L_d denotes the number of layers in the decoder. It performs a reverse operation of encoder here to generate the output $\mathbf{Y} \in \mathbb{R}^{(500,40)}$ of the same size as the input $\mathbf{I}$.

As shown in Fig. 8, we use the Tanh activation function $tanh(x) = (e^x - e^{-x}/e^x + e^{-x})$ for all the layers because Tanh activation function performed the best compared to the other nonlinear activation functions, except for the last layer, where
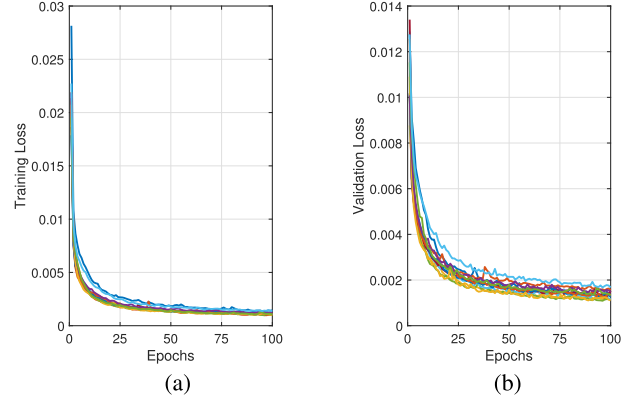

Fig. 14. Convergence of CNN AE for 13 testing streets. (a) Training loss. (b) Validation loss.

we used the ReLU activation function $ReLU(x) = \max(0, x)$ to ensure a positive real-value output. Moreover, $a \times b$ on each Conv 1-D layer indicates the filters and kernel size. The value on each Max-Pooling 1-D and Up-Sampling 1-D denotes the factor by which downsampling and upsampling are performed on the first dimension. Also, value on each dense layer indicates the number of neurons considered in that layer. The symbol $+$ indicates the addition of the outputs of two previous layers.

C. Designing the Loss Function

We use log-cosh loss, which is the logarithm of the prediction error's hyperbolic cosine. Also, we have $\mathbf{I}$ as the input to the encoder in the AE as well as the ground truth to be predicted from the decoder and $\mathbf{Y}$ as the predicted output of the AE. Thus, the difference between the input and the output of the AE can be given by $\theta_{i,j} = \mathbf{Y}_{i,j} - \mathbf{I}_{i,j}, \forall (i, j)$, where $i = \{1, \ldots, 500\}$ and $j = \{1, \ldots, 40\}$ denote the length and width of the streets (with building facades). We choose the log-cosh loss to help stabilize the training performance with fewer epochs (iterations) because the outliers minimally impact the log-cosh loss compared to the MSE loss [45]. Also, since we have appended zeros in the input $\mathbf{I}$, there have many zeros appended for shorter distances, which makes it difficult for the AE network to learn nonzeros values in closer distance ranges. Thus, we introduce a matrix $\hat{\mathbf{Y}} \in \mathbb{R}^{(500,40)}$, where for the nth training sample, given by

$$\hat{\mathbf{Y}}_{i,j}^n = \begin{cases} 0, & \text{if } \mathbf{I}_{i,j}^n = 0 \\ \mathbf{Y}_{i,j}^n, & \text{otherwise} \end{cases}, \quad \forall (i, j). \quad (10)$$

Then, the combined loss function for the nth training sample can be given by

$$\mathcal{L} = \mu\big(\log\big(\cosh\big(\hat{\mathbf{Y}}^n - \mathbf{I}^n\big)\big)\big) \\ + 0.1 \times \mu\big(\log\big(\cosh\big(\mathbf{Y}^n - \mathbf{I}^n\big)\big)\big) \quad (11)$$

where $\mu(\cdot)$ is the mean. The loss function has two parts. The first part focuses on the reconstruction error of the nonzero values. The second part focuses on the reconstruction error of all the values, whereas the weight 0.1 helps us in reducing the impact of the appended zeros.

TABLE V
RMSE IN PL PREDICTION USING THE FOUR MOST IMPORTANT CLUTTER FEATURES (CLUTTER$_4$)

| ML Algorithm	Street-by-street training-testing			Links-shuffle-split training-testing		
	Clutter$_4$	Clutter$_4$ + Building		Clutter$_4$	Clutter$_4$ + Building	
		Average over 25 AE runs	*Best* out of 25 AE runs	*Average* over 25 shuffles	*Average* over 25 AE runs	*Best* out of 25 AE runs
RF	6.2 ± 1.4 dB	6.9 ± 1.9 dB	5.8 ± 1.5 dB	**4.1 ± 0.2 dB**	**4.3 ± 0.2 dB**	**3.9 dB**
SVR	5.6 ± 1.2 dB	5.7 ± 1.3 dB	4.8 ± 0.9 dB	4.3 ± 0.2 dB	4.3 ± 0.2 dB	4.0 dB
Lasso	5.7 ± 1.5 dB	5.7 ± 1.3 dB	4.8 ± 1.0 dB	5.1 ± 0.2 dB	4.8 ± 0.2 dB	4.5 dB
Elastic-net	**5.5 ± 1.1 dB**	**5.5 ± 1.2 dB**	**4.7 ± 1.0 dB**	5.1 ± 0.2 dB	4.8 ± 0.2 dB	4.5 dB
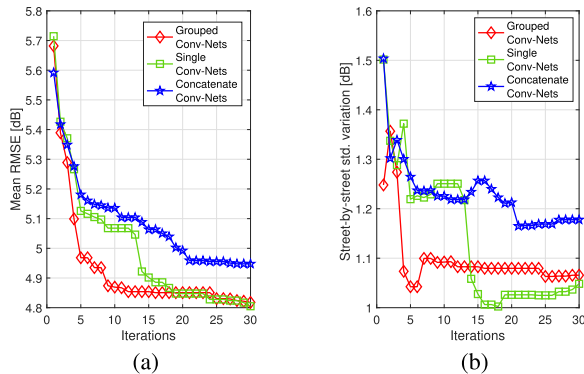
Fig. 15. Evaluation of grouped CNNs and iterations needed for reproducibility. (a) μ(RMSE) versus runs. (b) σ(RMSE) versus runs.

APPENDIX B
LOSS CONVERGENCE AND REPRODUCIBILITY OF THE PROPOSED CNN-BASED AE

We utilize the designed AE to extract the compressed feature vector $\mathbf{X} \in \mathbb{R}^{(12)}$ from the 2-D collapse $\mathbf{I} \in \mathbb{R}^{(500,40)}$ of the 3-D building dataset. Then, we train the AE in an end-to-end manner by minimizing the designed loss in (11). The convergence of the training and validation losses for 13 models created for street-by-street training and testing is shown in Fig. 14, in which the loss converges within 50 epochs (iterations utilized by the Adam optimizer to converge).

In Fig. 15, we evaluate the compressed features ($\mathbf{X}$) by making three types of AEs, in which only the grouped CNNs in the encoder as proposed in Fig. 8(a) is replaced by: 1) grouped Conv-Nets—as proposed; 2) single Conv-Net—remove Conv-Net-2 from the encoder; and 3) concatenate Conv-Nets—concatenate Conv-Net-1 and Conv-Net-2 serially, to obtain their respective feature set $\mathcal{F}_{\text{Clutter\_Building}}$. Furthermore, we show the best RMSE performance achieved by the Elastic-net regression to predict the PL, with a varying number of iterations. Grouped CNNs perform the best with smoother convergence compared to others. Furthermore, Fig. 15 also shows that with 25 iterations, we can achieve the reproducibility for the best PL prediction RMSE performance.

APPENDIX C
ANALYZING THE IMPORTANCE OF DESIGNED CLUTTER IN PL PREDICTIONS

In Fig. 13, we utilized the l_1-norm based Lasso regression to determine the importance of the individual clutter feature if all seven clutter features are provided for the PL prediction.
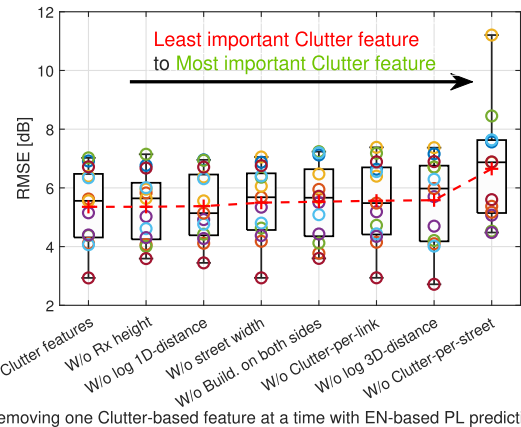

Fig. 16. Importance of each feature in the Elastic-net-based PL prediction model.

The importance of features, as quantified by their Lasso weights, in the descending order is given as: log 3-D distance, clutter per street, building on both sides, clutter per link, Rx height, street width, and log 1-D distance.

Feature importance can also be observed by comparing the change of prediction accuracy when each clutter feature is excluded individually from prediction. This approach works for all ML-based prediction algorithms and thus provides an alternative way of assessing feature importance.

In Fig. 16, we remove one feature at a time from the seven clutter features and determine the RMSE in Elastic-net-based PL prediction using street-by-street testing in Algorithm 2. Removing the clutter per street feature has the strongest consequence, degrading the mean RMSE by as much as 1.3 dB. Thus, based on the degradation of mean RMSE, the importance of the features[5] in descending order can be given: clutter per street, log 3-D distance, clutter per link, building on both sides, street width, log 1-D distance, and Rx height.

Based on observations from Figs. 13 and 16, we conclude that the most important features are the log 3-D distance, street clutter information given by clutter per street and clutter per link, and canyon status (buildings on both sides), referred to as Clutter$_4$. We analyze the RMSE in PL prediction using Clutter$_4$ for both street-by-street and shuffle-split testing, as shown in Table V. By only using the top four most influential features, the RMSE of 5.5 ± 1.1 dB can be achieved. When using Clutter$_4$, both the mean RMSE and

---

[5]The potential correlation among multiple features may underestimate the importance of a feature if it has high correlation with others.

standard deviations of almost all the ML-based predictions are within 0.2 dB from the results where all seven clutter features are used.

REFERENCES

[1] C. Phillips, D. Sicker, and D. Grunwald, "A survey of wireless path loss prediction and coverage mapping methods," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 255–270, 1st Quart., 2013.

[2] V. Erceg, "Channel models for broadband fixed wireless systems," IEEE 802.16 Broadband Wireless Access Working Group, San Jose, CA, USA, Tech. Rep. 3c-00, 2000. [Online]. Available: https://www.ieee802.org/16/tg3/contrib/802163c-01_29r4.pdf

[3] T. S. Rappaport *et al.*, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, 2013.

[4] A. Karttunen, A. F. Molisch, S. Hur, J. Park, and C. J. Zhang, "Spatially consistent street-by-street path loss model for 28-GHz channels in micro cell urban environments," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7538–7550, Nov. 2017.

[5] T. S. Rappaport, G. R. MacCartney, S. Sun, H. Yan, and S. Deng, "Small-scale, local area, and transitional millimeter wave propagation for 5G communications," *IEEE Trans. Antennas Propag.*, vol. 65, no. 12, pp. 6474–6490, Dec. 2017.

[6] J. Du, D. Chizhik, R. Feick, M. Rodriguez, G. Castro, and R. A. Valenzuela, "Suburban fixed wireless access channel measurements and models at 28 GHz for 90% outdoor coverage," *IEEE Trans. Antennas Propag.*, vol. 68, no. 1, pp. 411–420, Jan. 2020.

[7] (Mar. 2016). *5G Channel Model for Bands Up to 100 GHz, Annex A: Summary of Channel Sounding, Simulations and Measurement Data, 5GCM White Paper*. [Online]. Available: http://www.5gworkshops.com/5GCM.html

[8] J. Du *et al.*, "Directional measurements in urban street canyons from macro rooftop sites at 28 GHz for 90% outdoor coverage," *IEEE Trans. Antennas Propag.*, vol. 69, no. 6, pp. 3459–3469, Jun. 2021.

[9] T. K. Sarkar, Z. Ji, K. Kim, A. Medouri, and M. Salazar-Palma, "A survey of various propagation models for mobile communication," *IEEE Antennas Propag. Mag.*, vol. 45, no. 3, pp. 51–82, Jun. 2003.

[10] E. Montiel, A. S. Aguado, and F. X. Sillion, "A radiance model for predicting radio wave propagation in irregular dense urban areas," *IEEE Trans. Antennas Propag.*, vol. 51, no. 11, pp. 3097–3108, Nov. 2003.

[11] I. Popescu, D. Nikitopoulos, P. Constantinou, and I. Nafornita, "Comparison of ANN based models for path loss prediction in indoor environment," in *Proc. IEEE Veh. Technol. Conf.*, Montreal, QC, Canada, Sep. 2006, pp. 1–5.

[12] E. Ostlin, H.-J. Zepernick, and H. Suzuki, "Macrocell path-loss prediction using artificial neural networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 6, pp. 2735–2747, Jul. 2010.

[13] M. Ribero, R. W. Heath, Jr., H. Vikalo, D. Chizhik, and R. A. Valenzuela, "Deep learning propagation models over irregular terrain," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Brighton, U.K., May 2019, pp. 4519–4523.

[14] D. Wu, G. Zhu, and B. Ai, "Application of artificial neural networks for path loss prediction in railway environments," in *Proc. 5th Int. ICST Conf. Commun. Netw. China*, Beijing, China, 2010, pp. 1–5.

[15] S. P. Sotiroudis, S. K. Goudos, and K. Siakavara, "Neural networks and random forests: A comparison regarding prediction of propagation path loss for NB-IoT networks," in *Proc. 8th Int. Conf. Modern Circuits Syst. Technol. (MOCAST)*, Thessaloniki, Greece, May 2019, pp. 1–4.

[16] 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.

[17] Y. Egi and C. E. Otero, "Machine-learning and 3D point-cloud based signal power path loss model for the deployment of wireless communication systems," *IEEE Access*, vol. 7, pp. 42507–42517, 2019.

[18] U. Masood, H. Farooq, and A. Imran, "A machine learning based 3D propagation model for intelligent future cellular networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Waikoloa, HI, USA, Dec. 2019, pp. 1–6.

[19] J. Isabona and V. M. Srivastava, "Hybrid neural network approach for predicting signal propagation loss in urban microcells," in *Proc. IEEE Region 10 Hum. Technol. Conf. (R10-HTC)*, Dec. 2016, pp. 1–5.

[20] C. Park, D. K. Tettey, and H.-S. Jo, "Artificial neural network modeling for path loss prediction in urban environments," Apr. 2019, *arXiv:1904.02383*.

[21] 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, p. 1908, May 2019.

[22] M. E. Morocho-Cayamcela, M. Maier, and W. Lim, "Breaking wireless propagation environmental uncertainty with deep learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5075–5087, Aug. 2020.

[23] C. A. Oroza, Z. Zhang, T. Watteyne, and S. D. Glaser, "A machine-learning-based connectivity model for complex terrain large-scale low-power wireless deployments," *IEEE Trans. Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 576–584, Dec. 2017.

[24] 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, pp. 1–20, Feb. 2018.

[25] Y. Wang, S. R. Iyer, D. Chizhik, J. Du, and R. A. Valenzuela, "Channel prediction with terrains: Deep autoencoder with random forest," Nokia Bell Labs, Murray Hill, NJ, USA, Bell Labs Tech. Rep. ITD-19-59529F, Sep. 2019.

[26] N. Kuno and Y. Takatori, "Prediction method by deep-learning for path loss characteristics in an open-square environment," in *Proc. Int. Symp. Antennas Propag. (ISAP)*, Busan, South Korea, 2018, pp. 1–2.

[27] 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.

[28] R. Levie, C. Yapar, G. Kutyniok, and G. Caire, "RadioUNet: Fast radio map estimation with convolutional neural networks," Nov. 2019, *arXiv:1911.09002*.

[29] J.-Y. Lee, M. Y. Kang, and S.-C. Kim, "Path loss exponent prediction for outdoor millimeter wave channels through deep learning," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Marrakesh, Morocco, Apr. 2019, pp. 1–5.

[30] V. V. Ratnam *et al.*, "FadeNet: Deep learning-based mm-wave large-scale channel fading prediction and its applications," *IEEE Access*, vol. 9, pp. 3278–3290, 2021.

[31] A. Seretis and C. D. Sarris, "An overview of machine learning techniques for radiowave propagation modeling," Jan. 2021, *arXiv:2101.11760*.

[32] J.-H. Lee, J.-S. Choi, and S.-C. Kim, "Cell coverage analysis of 28 GHz millimeter wave in urban microcell environment using 3-D ray tracing," *IEEE Trans. Antennas Propag.*, vol. 66, no. 3, pp. 1479–1487, Mar. 2018.

[33] R. Charbonnier *et al.*, "Calibration of ray-tracing with diffuse scattering against 28-GHz directional urban channel measurements," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 14264–14276, Dec. 2020.

[34] J.-Y. Lee, J.-H. Lee, and S.-C. Kim, "Improving the accuracy of millimeter-wave ray-tracing simulations by modeling roadside trees," *IEEE Antennas Wireless Propag. Lett.*, vol. 18, no. 1, pp. 162–166, Jan. 2019.

[35] United States Geological Survey (USGS). (Oct. 2014). *Dataset of 2013–2014 USGS CMGP LiDAR: Post Sandy (New York City)*. [Online]. Available: https://coast.noaa.gov/htdata/lidar1_z/geoid12b/data/4920/

[36] CADMapper. *New York City Building CAD Files Dataset*. Accessed: Nov. 2019. [Online]. Available: https://cadmapper.com/

[37] V. N. Vapnik and A. Y. Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities," *Theory Probab. Appl.*, vol. 16, no. 2, pp. 264–280, 1971.

[38] *Study on Channel Model for Frequencies From 0.5 to 100 GHz*, document TR 38.901, Version 14.3.0, 3GPP, Dec. 2017.

[39] M. Marcus and B. Pattan, "Millimeter wave propagation: Spectrum management implications," *IEEE Microw. Mag.*, vol. 6, no. 2, pp. 54–62, Jun. 2005.

[40] S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter-wave cellular wireless networks: Potentials and challenges," *Proc. IEEE*, vol. 102, no. 3, pp. 366–385, Mar. 2014.

[41] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz, "Towards 3D point cloud based object maps for household environments," *Robot. Auton. Syst.*, vol. 56, no. 11, pp. 927–941, Nov. 2008.

[42] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," 2016, *arXiv:1612.00593*.

[43] M. Hall, "Correlation based feature selection for machine learning," Ph.D. dissertation, Dept. Comput. Sci., Univ. Waikato, Hamilton, New Zealand, 1999.

[44] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[45] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[46] N. Ketkar, "Introduction to keras," in *Deep Learning With Python*. Berlin, Germany: Springer, 2017, pp. 97–111.

[47] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," Dec. 2014, *arXiv:1412.6980*.

[48] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[49] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdisc. Top.*, vol. 69, no. 6, Jun. 2004, Art. no. 066138.

[50] T. Hengl, M. Nussbaum, M. N. Wright, G. B. M. Heuvelink, and B. Gräler, "Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables," *PeerJ*, vol. 6, pp. 1–47, Aug. 2018.

[51] M. Crane, "Questionable answers in question answering research: Reproducibility and variability of published results," *Trans. Assoc. Comput. Linguistics*, vol. 6, pp. 241–256, Apr. 2018.

[52] D. Sculley *et al.*, "Machine learning: The high interest credit card of technical debt," in *Proc. Softw. Eng. Mach. Learn. (NIPS Workshop, SE4ML)*, 2014, pp. 1–9.

[53] E. Breck, S. Cai, E. Nielsen, M. Salib, and D. Sculley. (2016). *Whats Your ML Test Score? A Rubric for ML Production Systems*. [Online]. Available: https://storage.googleapis.com/pub-tools-public-publication-data/pdf/45742.pdf

[54] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of Research on Machine Learning Applications and Trends*. Hershey, PA, USA: IGI Global, 2009.

[55] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, Dec. 2014, pp. 3320–3328.

Ankit Gupta (Student Member, IEEE) received the B.Tech. degree in electronics and communication engineering from Guru Gobind Singh Indraprastha University, New Delhi, India, in 2015. He is currently pursuing the Ph.D. degree in signal processing and communication engineering with Heriot-Watt University, Edinburgh, U.K.

He was with Aricent Technologies, Ltd., (Holdings), Gurugram, India, until 2017. His current research interests include 5G, cooperative communications, multiple-input–multiple-output (MIMO) networks, nonorthogonal multiple access techniques, deep learning, autoencoders, and optimization methods in signal processing and communications.

Jinfeng Du (Member, IEEE) received the B.Eng. degree in electronic information engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2004, and the M.Sc., Tekn.Lic., and Ph.D. degrees from the Royal Institute of Technology (KTH), Stockholm, Sweden, in 2006, 2008, and 2012, respectively.

He was a Post-Doctoral Researcher with the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, from 2013 to 2015. He then joined Bell Labs at Crawford Hill, Holmdel, NJ, USA. His research interests are in the general area of wireless communications, especially in communication theory, information theory, wireless networks, millimeter-wave propagation, and channel modeling.

Dr. Du received the Best Paper Award from IC-WCSP in October 2010. His paper was elected as one of the best 50 papers in IEEE GLOBECOM 2014. He received the prestigious Hans Werthen Grant from the Royal Swedish Academy of Engineering Science (IVA) in 2011, the Chinese Government Award for Outstanding Self-Financed Students Abroad in 2012, the International PostDoc Grant from the Swedish Research Council in 2013, and three grants from the Ericsson Research Foundation.

Dmitry Chizhik (Fellow, IEEE) received the Ph.D. degree in electrophysics from New York University (NYU), Brooklyn, NY, USA, in 1991. His thesis work was in ultrasonics and nondestructive evaluation.

He joined the Naval Undersea Warfare Center, New London, CT, USA, where he did his research in scattering from ocean floor, geoacoustic modeling of porous media, and shallow water acoustic propagation. In 1996, he joined Bell Laboratories, Holmdel, NJ, USA, with a focus on radio propagation modeling and measurements, using deterministic and statistical techniques. He has been involved in measurement, modeling, and channel estimation of multiple-input–multiple-output (MIMO) channels. The results are used both for determination of channel-imposed bounds on channel capacity and system performance and for optimal antenna array design. His recent work has included system and link simulations of satellite and femtocell radio communications and millimeter-wave propagation that included all aspects of the physical layer. His research interests are in acoustic and electromagnetic wave propagation and scattering, signal processing, communications, radar, sonar, and medical imaging.

Dr. Chizhik was a recipient of the Bell Labs President's Award and a Distinguished Member of Technical Staff.

Reinaldo A. Valenzuela (Fellow, IEEE) received the B.Sc. degree from the University of Chile, Santiago, Chile, in 1975, and the Ph.D. degree from Imperial College London, London, U.K., in 1982.

He is currently the Director of the Communication Theory Department and a Distinguished Member of Technical Staff at Bell Laboratories, Murray Hill, NJ, USA. He has published 190 articles and 44 patents. He has over 32 900 Google Scholar citations and is a "Highly Cited Author" in Thomson ISI and a Fulbright Senior Specialist. He is engaged in propagation measurements and models, multiple-input–multiple-output (MIMO)/space–time systems achieving high capacities using transmit and receive antenna arrays, heterogeneous network (HetNet), small cells, and next-generation air interface techniques and architectures.

Dr. Valenzuela is a member of the National Academy of Engineering. He is a fellow of Bell Labs and Wireless World Research Forum (WWRF). He received the IEEE Eric E. Sumner Award, the 2014 IEEE CTTC Technical Achievement Award in 2015, and the IEEE VTS Avant Garde Award.

Mathini Sellathurai (Senior Member, IEEE) held visiting positions with Bell Laboratories, Holmdel, NJ, USA, and the Canadian Communications Research Centre, Ottawa, ON, Canada. She is currently a Professor of signal processing and wireless communications and the Dean of science and engineering with Heriot-Watt University, Edinburgh, U.K. She has been active in signal processing research for the past 20 years and has a strong international track record in multiple-input–multiple-output (MIMO) signal processing with applications in radar and wireless communications. She has authored or coauthored over 200 peer-reviewed articles in leading international journals and IEEE conferences, has given invited talks, and has written several book chapters as well as a research monograph as a lead author. Her current research interests include full-duplex systems, passive radar topography, localization, massive-MIMO, nonorthogonal multiple access, waveform designs, caching technologies, assisted care technologies, the Internet of Things (IoT), hearing aids, optimal coded modulation designs using autoencoders, channel prediction, and millimeter-wave imaging and communications.

Prof. Sellathurai was a recipient of the IEEE Communication Society Fred W. Ellersick Best Paper Award in 2005, the Industry Canada Public Service Awards for contributions to science and technology in 2005 and technology transfers to industry in 2004, and the Natural Sciences and Engineering Research Council of Canada (NSERC) Doctoral Award for her Ph.D. dissertation in 2002. From 2015 to 2018, she was the General Co-Chair of IEEE SPAWC2016 in Edinburgh and a member of the IEEE SPCOM Technical Strategy Committee from 2014 to 2019. She was an Editor of IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2009 to 2014.