RadioUNet: Fast Radio Map Estimation with Convolutional Neural Networks

4 authors:

Ron Levie
Technische Universität Berlin
9 PUBLICATIONS   **89** CITATIONS

SEE PROFILE

Çağkan Yapar
Technische Universität Berlin
11 PUBLICATIONS   **13** CITATIONS

SEE PROFILE

Gitta Kutyniok
Technische Universität Berlin
224 PUBLICATIONS   **6,378** CITATIONS

SEE PROFILE

Giuseppe Caire
Technische Universität Berlin
649 PUBLICATIONS   **27,659** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project   Rate Splitting for MIMO Wireless Networks: A Promising PHY-Layer Strategy for 5G and beyond   View project

Project   Sparse Regularization of Inverse Problems and Anisotropic Structures   View project

RadioUNet: Fast Radio Map Estimation with Convolutional Neural Networks

Ron Levie[†][*]   Çağkan Yapar[‡][*]   Gitta Kutyniok[†][§]   Giuseppe Caire[‡]

[†] Institute of Mathematics, TU Berlin,
[‡] Institute of Telecommunication Systems, TU Berlin,
[§]Department of Physics and Technology, University of Tromsø

Abstract

In this paper we propose a highly efficient and very accurate method for estimating the propagation pathloss from a point x to all points y on the 2D plane. Our method, termed RadioUNet, is a deep neural network. For applications such as user-cell site association and device-to-device (D2D) link scheduling, an accurate knowledge of the pathloss function for all pairs of locations is very important. Commonly used statistical models approximate the pathloss as a decaying function of the distance between the points. However, in realistic propagation environments characterized by the presence of buildings, street canyons, and objects at different heights, such radial-symmetric functions yield very misleading results. In this paper we show that properly designed and trained deep neural networks are able to learn how to estimate the pathloss function, given an urban environment, very accurately and extremely quickly. Our proposed method generates pathloss estimations that are very close to estimations given by physical simulation, but much faster. Moreover, experimental results show that our method significantly outperforms previously proposed methods based on radial basis function interpolation and tensor completion.

1   Introduction

In wireless communication, pathloss is a quantity that measures the loss of signal strength (reduction in power, or attenuation) between a transmitter and receiver due to large scale effects. The reduction in power can be caused by different factors, like an obstacle in the line of sight between the transmitter and receiver, or simply a loss in power as the signal travels through free space. The pathloss function, also called path gain function or radio

---

[*]Çağkan Yapar and Ron Levie contributed equally to this work.

map, is a function that assigns to each transmitter and receiver pair of locations the signal attenuation. Alternatively, the pathloss function, or radio map, can also be defined for a fixed transmitter location, as the function that assigns to each spatial location the pathloss at this location from the fixed transmitter.

Many applications in wireless communication explicitly rely on the knowledge of the pathloss function, and thus, estimating pathloss is a tremendously important task. For example, in device to device (D2D) link scheduling, there is a set of wireless devices that transmit signals to each other in pairs. A pair of devices that communicate defines a transmitter-receiver link. The signal sent by a transmitter is not only picked up by its respective receiver, but also by receivers of other links. Generally speaking, if all transmitters transmit at maximal power at the same time, the receivers would mainly pick up interference. To avoid this problem, a link scheduler decides in each time slot which links are active (i.e., their corresponding transmitters are switched on or off) or, more in general, which power each transmitter should use [10, 34, 9]. In order to schedule effectively, it is useful to predict the signal strength of each transmitter at all the receivers, which in turns requires the knowledge of the radio map.

Another example is base station assignment, or user-cell site association, where the goal is to assign a set of wireless devices to a set of cellular base stations. In order to decide which device to assign to which station, it is important to know the radio map (see, e.g., [3] and references therein). Some additional applications that rely on the knowledge of the pathloss function are fingerprint based localization [23], physical-layer security [32], power control in emerging systems [6, 25] and activity detection [5].

1.1 Pathloss Function Prediction

A multitude of approaches for estimating the pathloss function were proposed in the past. We roughly group these approaches in three categories. *Data driven interpolation methods* assume that some measurements of the pathloss function are given at certain locations. These methods estimate the pathloss function at non-measured locations via some signal processing approach and do not rely—or rely only lightly—on a model of the physical phenomenon. Some examples are radial basis function interpolation [4, Sect 5.1], tensor completion [29], support vector regression [31] and matrix completion [7].

Model based interpolation methods combine measurements of the pathloss function with a priori assumptions on the physical system to estimate the pathloss function at non-measured locations. For example, in tomography methods, the attenuation due to shadowing can be derived under some modeling assumptions from the so called spatial loss field (SLF), which in turn can be estimated from the measurements. Here, various assumptions on the underlying SLF can be imposed, e.g., low-rank structure [18], sparsity [1], and piecewise homogeneity [16, 17].

Last, *model based prediction* estimates the pathloss function based only on available prior knowledge, e.g., physical considerations, without taking any measurements from the area of interest. Some examples are ray-tracing [26], dominant path model [33], and empirical models, e.g. [36].

2

1.2 Our Contribution

In this paper we propose two versions of a pathloss estimation method based on deep learning, which we term *RadioUNet*. In our setting, we consider mobile devices/base stations in an urban environment. The first version of RadioUNet only uses the geometry of the urban environment, and no measurements, so it can be categorized as model based simulation. However, as opposed to classical model based simulation, our model is learned from training data, and does not have an explicit interpretable formulation. The second model combines measurements and a partial knowledge of the geometry of the urban environment and can thus be categorized as a model based interpolation method. Here, again, the model is implicit and learned from data. Our deep learning based methods are efficient, estimating the whole radio map in an order of 10^{-3}sec, with normalized mean square accuracy of order 10^{-2}.

We note that two recent papers proposed using deep learning methods for estimating radio maps or related notions. In [12] the network receives as input a city map and two 2D images of distances of each pixel from the transmitter and receiver, along with additional system parameters. The network then outputs the path gain between the transmitter and the receiver. In [28] spatial locations are classified as in the service zone or outside the service zone of a base station via a fully connected network. As opposed to these methods, which estimate pathloss between pairs of transmitter-receiver locations, our network computes the whole radio map at once.

2 Background and Preliminaries

2.1 Wireless Communication

The pathloss is defined as $P_{\mathrm{L}} = 10 \log_{10} \left(P_{\mathrm{t}}/P_{\mathrm{r}} \right)$ dB, where P_{t} and P_{r} denote the transmitted power and received power respectively. Here we consider isotropic antennas and loss of signal strength due to large scale effects.

The noise floor of a communication system is the inevitable background noise level, below which signals cannot be detected in general. In this paper we set the transmitter power to 0dBm and the noise floor to -80dB, truncating roughly the lowest quarter of pathloss values in our generated pathloss function simulations.

2.2 Convolutional Neural Networks

In the following, a *signal* is a function from a 2D grid to some $\mathbb{R}^n$, where n is called the number of channels of the signal. If $n = 1$, we call the signal a gray level image. A *Convolutional neural network* (CNN) is a parametric function/transform that maps signals to signals or signals to scalars [15], and is typically used in machine learning applications in imaging science. A CNN is defined by aggregating five basic computational steps, also called *layers*. A *convolution layer* is a step where an input signal (also called feature map) is convolved with a filter kernel and added to some pre-defined signal called the *bias*. The

number of input channels and output channels need not coincide. An *activation function* is any function applied on the entries of a signal, and a typical choice is ReLU. A *pooling layer* takes a signal and down-samples it, e.g., by assigning the maximal entry of each 2×2 patch to the corresponding entry of the down-sampled signal. An *up-sampling layer* up-samples lower resolution signals to higher resolution ones. Last, a *fully connected layer* is a general linear operator/matrix applied on the signal, and added to some pre-defined signal called the *bias*.

A typical CNN architecture applies a sequence of convolution, activation, and pooling/up-sampling layers many times, optionally followed by some fully connected layers. The overall number of layers is called the depth of the CNN. To specify a CNN requires to define the exact order of the layers, and to choose specific filter kernels and biases for the convolution layers as well as specific matrices an biases for the fully connected layers. CNNs are based on large numbers of free parameters (the coefficients of the filter kernels, biases, and fully connected matrices), typically in the order of millions. Thus, the parameters are generally not handpicked, but rather automatically generated. This is done by a procedure that optimizes the CNN parameters to achieve some desired target—or in the machine learning jargon—the CNN is trained.

In supervised learning, a dataset of sample input signals and corresponding desired output signals is provided. The CNN parameters are optimized to minimize the error between the signals generated by applying the CNN on the input signals, and the desired output signals. Optimizing—or training—the parameters is typically done via variants of stochastic gradient descent, e.g., Adam [14].

2.3   UNets

UNet is a special CNN architecture, introduced in [27], and used in a multitude of applications, including image segmentation [30, 2, 22, 8], video predicting [21], super resolution/ image inpainting [19], inverse problems in imaging [13], image-to-image translation [35], and medical image analysis [20] to name a few.

UNets consist of convolution, pooling, up-sampling, and activation function layers, without fully connected layers. The UNet architecture is divided into two paths. The first portion of the layers gradually contracts the image as the layers deepen, and gradually increases the number of channels. This path—also called the *encoder*—is interpreted as a procedure for extracting "concepts" which become more complex/high-level and less spatially localized along the layers. The second portion of the layers—also called the *decoder*—expands the image as the layers deepen and reduces the number of channels gradually. This path is interpreted as a procedure of combining/synthesizing the concepts, layer by layer, to lower-level concepts, and eventually to an output image. The decoder layers are derived by up-sampling lower resolution images, and thus lack high resolution information on their own. To provide high resolution information to the decoder layers, the feature map in the channels of the encoder layers are copied and concatenated to the corresponding channels of the decoder layers having the same resolution. This copying between non-neighboring layers is called *skip connection*.

4

3 Estimating Radio Maps via UNets

3.1 The RadioUNet Method

RadioUNet is a UNet that estimates radio maps. We define RadioUNet in two settings. In the first scenario, the RadioUNet receives as input two channels: the transmitter location and the urban environment, which are assumed to be known accurately. The RadioUNet returns one output channel, which is the estimated radio map. We call this version of the method RadioUNet$_\mathrm{C}$, with 'C' for "clean maps". In the second scenario, the RadioUNet receives as input three channels: a perturbed version of the urban environment, some measurements of the accurate radio map, and the transmitter location. The UNet returns here as well the radio map. To model perturbation, we consider the situation where some buildings are missing from the map. We call this version of our method RadioUNet$_\mathrm{S}$, with 'S' for "samples". The architectures of our proposed UNets are reported in Table 1.

The urban geometry is given as a morphological (black and white) 2D image, where the interior of the buildings is white (pixel value = 1), and the exterior of the buildings is black (pixel value = 0). The transmitter is given in a second input channel as a morphological image, where the pixel in which the transmitter is located is white and the rest is black. For the second scenario, we consider in addition a third channel, consisting of the radio map values at the pixels corresponding to the measurements and zero elsewhere. The estimated radio map is a 2D image of the pathloss at each spatial location, from the corresponding input transmitter. Pathloss is represented in *gray level*, as explained next.

The simulated radio map stores at each pixel the pathloss between the pixel location and the transmitter location in dB. We convert the pathloss values P_L to pixel values between 0 and 1 as follows. Consider the noise floor $\mathcal{N}$ which we assume in our setting to be -80dB. Denote by M_1 the maximal pathloss in all radio maps in the dataset, and define $f = \max\{\frac{P_\mathrm{L}-\mathcal{N}}{M_1-\mathcal{N}}, 0\}$. Here, $f = 0$ represents anything below the noise level, and $f = 1$ represents the maximal gain at the transmitter. We call the dimension of the resulting pixel values *gray level*.

Let us explain the importance of our gray level conversion when evaluating the performance of any pathloss estimation. We evaluate performance of any approximation $\tilde{f} : \mathcal{D} \to \mathbb{R}$ of a signal $f : \mathcal{D} \to \mathbb{R}$, where $\mathcal{D} = \{x_n\}_n$ is some finite grid in $\mathbb{R}^2$, via the normalized mean square error (NMSE)

$$E = \frac{\sum_n |\tilde{f}(x_n) - f(x_n)|^2}{\sum_n |f(x_n)|^2}. \tag{1}$$

The numerator in (1) represents the absolute error, and the denominator represents the global magnitude of f. The coefficients $|\tilde{f}(x_n) - f(x_n)|^2$ and $|f(x_n)|^2$ having larger values affect the outcome of E the most, and small values are negligible. It is thus crucial to express the signal f in a representation in which the important parts of the signal obtain large values.

In our case, the representation of the radio map should be constructed in such a way that small powers contribute small values to E. Indeed, locations of small power represent

| | | | | | | | RadioUNet | | | | | | | | | | | |
Layer	In	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	out
Resolution	256	256	128	64	64	32	32	16	8	16	32	32	64	64	128	256	256	256
Channels	2/3	6	40	60	80	100	120	200	400	400	240	200	160	120	80	29	32	1
Filter	3	5	5	5	5	3	5	5	4	4	3	6	5	6	6	5	2	

Table 1: RadioUNet architecture. *Resolution* is the number of pixels of the image in each channel along the x and y axes. *Filter* is the number of pixels of each filter kernel along the x and y axes.

a weak signal. If we represent the radio map as standard pathloss, in dB, the smaller the power in a certain location, the higher the magnitude of the pathloss, with negative sign. When the power goes to zero, the pathloss diverges to $-\infty$. In this representation, locations of a weak signal dominate the global magnitude of the radio map, and in general define a misleading concept of the "size" of the radio map. A similar situation occurs for the absolute error (the numerator of (1)).

Our gray level conversion resolves this issue. Indeed, anything below the noise floor is deemed "to small to be interesting", and set to zero. The values of higher power, which are most important, are transformed to 1, which is the maximal gray level. We note that papers like [29, 7, 18] suffer from the aforementioned shortcoming, and it is thus difficult to interpret their reported performance.

3.2 Training

We perform supervised learning on a dataset of urban maps and their corresponding simulated radio maps, converted to gray level as described above.

Maps are taken from OpenStreetMap [24] and converted to morphological images of 256×256 pixels, where each pixel represents one square meter. The simulated radio maps were generated using the Dominant Path Model, with the radio network planning software *WinProp* [11] and converted the obtained results to gray level. The data consists of 495 maps, and 100 random transmitter locations for each map. The 495 maps are split into a training (250 maps), test (140 maps), and validation (105 maps) sets. Accuracy is measured by NMSE. Training of all methods was performed with Adam [14], for 20 epochs, with learning rate of 10^{-4}, no regularization, and batch size 15.

The physical simulation is roughly estimated to have accuracy of order 10^{-2} NMSE [33], which is comparable to the error between the RadioUNet and the simulation. Thus, by the triangle inequality, the estimated error between RadioUNet and real life radio maps is estimated to be of the same order, 10^{-2} NMSE.

3.3 Results

First, we compare our method with a model based simulation. Specifically we compare run-time with the efficient dominant pathloss method [33]. RadioUNet estimates radio maps in roughy three orders of magnitudes faster. In our experiments, WinProp completes a simulation in roughly an order of 1sec on a Intel Core i7-8750H CPU, and RadioUNet roughly an order of 10^{-3}sec on an Nvidia Quadro GR100.
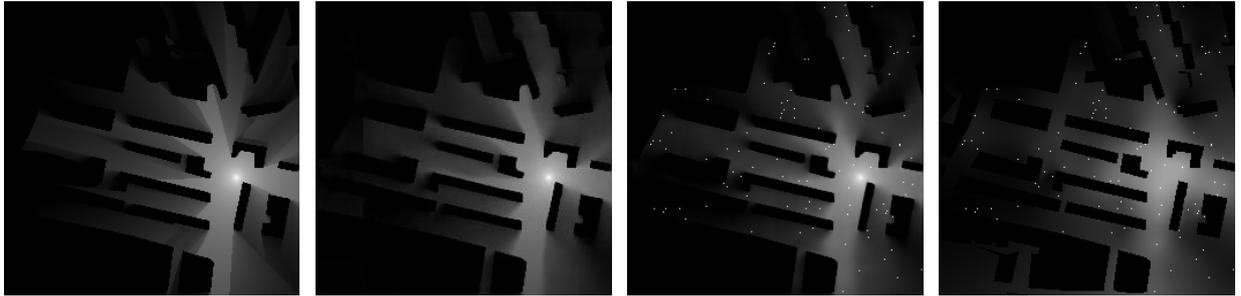
Figure 1: From left to right. 1:Ground truth radio map. 2:RadioUNet$_c$. 3:RadioUNet$_S$. 4:RBF. The measurement locations for both RadioUNetS and RBF are marked in white.

Next, we compare RadioUNet$_C$ and RadioUNet$_S$ with data driven interpolation methods: radial basis function (RBF) interpolation using multiquadric function [4, Sect 5.1] and tensor completion [29]. For the data driven methods we set to zero the gray level values inside the buildings post-processing, thus using the urban geometry data. Without this step, data driven interpolation methods obtain a very poor NMSE close to 1 since they are not able to recover the sharp building edges. We evaluate the gray level NMSE of RadioUNet$_C$ and plot it as a horizontal line. We moreover plot the NMSE of RadioUNet$_S$ as a function of the number of samples. We plot the NMSE of the two data driven interpolation methods as a function of the number of samples as well. The results are reported in Figure 2. Both versions of RadioUNet clearly outperform state-of-the-art. Aside from that, RadioUNet is roughly three orders of magnitude faster than RBF interpolation, and five orders of magnitude faster than tensor completion interpolation.

In Figure 1 we show an example of a ground truth radio map generated by simulation, and the estimated radio map computed by the RadioUNet$_C$ and RadioUNet$_S$. We also show the estimation by RBF interpolation. Aside from the low quantitative error, RadioUNet seems to synthesize radio maps from the urban geometry which qualitatively captures the correct shadow patterns. Note that the results in Figure 1 are representative of the general quality of RadioUNet. One might naively interpret the success of the RadioUNet by postulating that it learns to mimic a physical model, like ray-tracing or some differential equation like Maxwell's equations. However, we believe that this is a misleading viewpoint. A more reasonable interpretation follows from the encoder-decoder description of general UNets. In the encoder path, the RadioUNet extracts complicated concepts about the geometry of the urban environment and the mutual relationship between the different geometric features, their location, and the location of the transmitter. Then, in the decoder path, the RadioUNet uses these concepts to synthesize the radio map. Thus, RadioUNet is based on extracting and analyzing *global* information about the urban environment, as opposed to classical physical models that are based on *local* information, like collisions with the geometry in ray-tracing and derivatives in differential equations. In this viewpoint, it is more fitting to compare RadioUNet to a highly skilled artist that draws radio maps from his/her perception of the urban environment as a whole, rather than comparing to a classical local physical model.
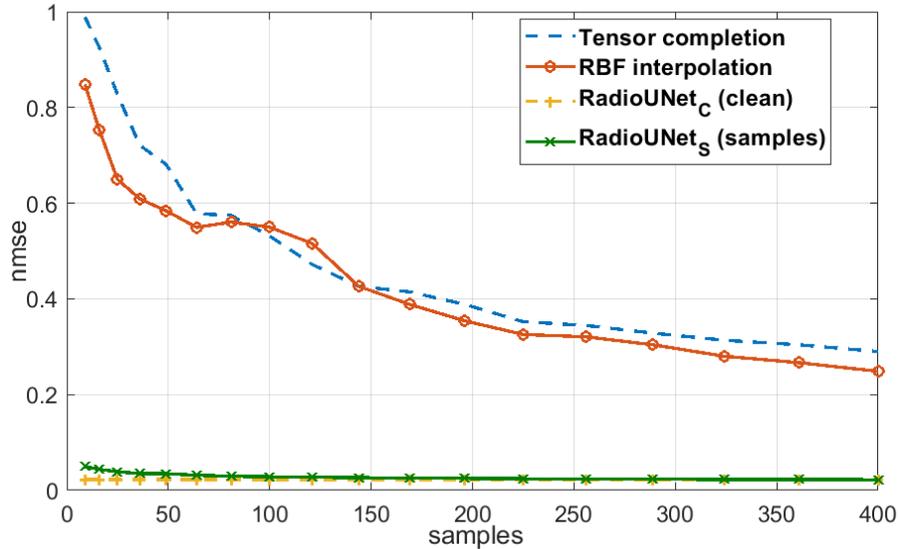
Figure 2: Estimation error of the radio map reconstruction methods as a function of the number of measurements. RadioUNet$_C$ has zero samples, and is given as a baseline.

References

[1] M. Angel Gutierrez-Estevez, R. L. G. Cavalcante, and S. Stanczak. Nonparametric radio maps reconstruction via elastic net regularization with multi-kernels. In *Proc. IEEE Int. Workshop Sig. Proc. Advances Wireless Comm. (SPAWC)*, pages 1–5, Kalamata, Greece, June 2018.

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Analysis and Machine Intel.*, 39(12):2481–2495, Dec 2017.

[3] D. Bethanabhotla, O. Y. Bursalioglu, H. C. Papadopoulos, and G. Caire. Optimal user-cell association for massive MIMO wireless networks. *IEEE Trans. Wireless Comm.*, 15(3):1835–1850, March 2016.

[4] C. M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, Inc., New York, NY, USA, 1995.

[5] Z. Chen, F. Sohrabi, and W. Yu. Sparse activity detection for massive connectivity. *IEEE Trans. Sig. Proc.*, 66(7):1890–1904, April 2018.

[6] T. V. Chien, T. N. Canh, E. Björnson, and E. G. Larsson. Power control in cellular massive MIMO with varying user activity: A deep learning solution. *CoRR*, abs/1901.03620, 2019.

[7] S. Chouvardas, S. Valentin, M. Draief, and M. Leconte. A method to reconstruct coverage loss maps based on matrix completion and adaptive sampling. In *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP)*, pages 6390–6394, Shanghai, China, March 2016.

[8] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. 3D U-Net: Learning dense volumetric segmentation from sparse annotation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, pages 424–432. Springer International Publishing, 2016.

[9] W. Cui, K. Shen, and W. Yu. Spatial deep learning for wireless scheduling. *IEEE Jour. Sel. Areas in Comm.*, 37(6):1248–1261, June 2019.

[10] C. Geng, N. Naderializadeh, A. S. Avestimehr, and S. A. Jafar. On the optimality of treating interference as noise. *IEEE Trans. Inf. Theory*, 61(4):1753–1767, April 2015.

[11] R. Hoppe, G. Wölfle, and U. Jakobus. Wave propagation and radio network planning software winprop added to the electromagnetic solver package feko. In *Proc. Int. Appl. Computational Electromagnetics Society Symp. - Italy (ACES)*, pages 1–2, Florence, Italy, March 2017.

[12] T. Imai, K. Kitao, and M. Inomata. Radio propagation prediction model using convolutional neural networks by deep learning. In *Proc. European Conf. Antennas and Propagation (EuCAP)*, pages 1–5, Krakow, Poland, March 2019.

[13] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Trans. Image Proc.*, 26(9):4509–4522, Sep. 2017.

[14] D. P. Kingma and J. Ba. Adam: a method for stochastic optimization. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, May 2015.

[15] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11):2278–2324, Nov 1998.

[16] D. Lee, D. Berberidis, and G. B. Giannakis. Adaptive Bayesian radio tomography. *IEEE Trans. Sig. Proc.*, 67(8):1964–1977, April 2019.

[17] D. Lee and G. B. Giannakis. A variational Bayes approach to adaptive channel-gain cartography. In *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc. (ICASSP)*, pages 8434–8438, Brighton, United Kingdom, May 2019.

[18] D. Lee, S. Kim, and G. B. Giannakis. Channel gain cartography for cognitive radios leveraging low rank and sparsity. *IEEE Trans. Wireless Comm.*, 16(9):5953–5966, Sep. 2017.

[19] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee. Enhanced deep residual networks for single image super-resolution. In *Proc. Conf. Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1132–1140, Honolulu, HI, USA, July 2017.

[20] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sanchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60 – 88, December 2017.

[21] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. In *Int. Conf. Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[22] F. Milletari, N. Navab, and S. Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In *Proc. Int. Conf. 3D Vision (3DV)*, pages 565–571, Stanford, CA, USA, Oct 2016.

[23] S. Nikitaki, G. Tsagkatakis, and P. Tsakalides. Efficient training for fingerprint based positioning using matrix completion. In *Proc. European Sig. Proc. Conf. (EUSIPCO)*, pages 195–199, Bucharest, Romania, Aug 2012.

[24] OpenStreetMap contributors. Planet dump retrieved from https://planet.osm.org . `https://www.openstreetmap.org`, 2017.

[25] S. Park, A. Q. Truong, and T. H. Nguyen. Power control for sum spectral efficiency optimization in MIMO-NOMA systems with linear beamforming. *IEEE Access*, 7:10593–10605, 2019.

[26] K. Rizk, J. . Wagen, and F. Gardiol. Two-dimensional ray-tracing modeling for propagation prediction in microcellular environments. *IEEE Trans. Vehic. Tech.*, 46(2):508–518, May 1997.

[27] O. Ronneberger, O. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Munich, Germany, 2015. Springer International Publishing.

[28] K. Saito, Y. Jin, C. Kang, J. Takada, and J.-S. Leu. Two-step path loss prediction by artificial neural network for wireless service area planning. *IEICE Communications Express*, 2019.

[29] D. Schäufele, R. L. G. Cavalcante, and S. Stanczak. Tensor completion for radio map reconstruction using low rank and smoothness. In *Proc. IEEE Int. Workshop Sig. Proc. Advances Wireless Comm. (SPAWC)*, pages 1–5, Cannes, France, July 2019.

[30] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Analysis and Machine Intel.*, 39(4):640–651, April 2017.

[31] R. Timoteo, D. Cunha, and G. Cavalcanti. A proposal for path loss prediction in urban environments using support vector regression. In *Proc. Adv. Int. Conf. Telecomm., (AICT)*, volume 2014, pages 119–124, Paris, France, Jul 2014.

[32] Z. Utkovski, P. Agostini, M. Frey, I. Bjelakovic, and S. Stanczak. Learning radio maps for physical-layer security in the radio access. In *Proc. IEEE Int. Workshop Sig. Proc. Advances Wireless Comm. (SPAWC)*, pages 1–5, Cannes, France, July 2019.

[33] R. Wahl, G. Wölfle, P. Wildbolz, and F. Landstorfer. Dominant path prediction model for urban scenarios. In *Proc. IST Mobile and Wireless Communications*, 2005.

[34] X. Yi and G. Caire. Optimality of treating interference as noise: A combinatorial perspective. *IEEE Trans. Inf. Theory*, 62(8):4654–4673, Aug 2016.

[35] Z. Yi, H. Zhang, P. Tan, and M. Gong. DualGAN: Unsupervised dual learning for image-to-image translation. In *Proc.IEEE Int. Conf. Comp. Vision (ICCV)*, pages 2868–2876, Oct 2017.

[36] T. Zugno, M. Drago, M. Giordani, M. Polese, and M. Zorzi. Towards standardization of millimeter wave vehicle-to-vehicle networks: Open challenges and performance evaluation. *arXiv preprint arXiv:1910.00300*, 2019.