





Uncertainty Estimation via Monte Carlo Dropout in CNN-Based mmWave MIMO Localization

Mohammad Amin Maleki Sadr , João Gante, *Member, IEEE*, Benoit Champagne , *Senior Member, IEEE*, Gabriel Falcao , *Senior Member, IEEE*, and Leonel Sousa , *Senior Member, IEEE*

Abstract—Recently, there has been much interest in the use of convolutional neural networks (CNN) for mobile user localization in massive multiple-input multiple-output (MIMO) systems operating at millimeter wave (mmWave) frequencies. However, current CNN-based approaches cannot predict the confidence interval bounds for the localization accuracy. While the Bayesian neural network (BNN) method can be employed to estimate the model uncertainty, it entails a high computational cost. In this letter, the Monte Carlo (MC) dropout based method is proposed as a low-complexity approximation to BNN inference for capturing the uncertainty in a CNN-based mmWave MIMO outdoor localization system, without sacrificing accuracy. The proposed method is evaluated by means of simulations using a ray-tracing model of urban propagation at 28GHz. Results show that the localization uncertainty region can be properly determined and that their shape depends on the maximum power received at the user.

Index Terms—mmWave communications, massive MIMO, CNN, fingerprint localization.

I. INTRODUCTION

FUTURE generations of mobile wireless networks will combine transmissions at millimeter wave (mmWave) frequencies with massive MIMO technologies in order to boost user throughput, spectral/energy efficiency and overall capacity [1]. These networks will be required to provide enhanced localization accuracy of user devices within sub-meter accuracy depending on use cases [2]. Nonetheless, mmWave channels exhibit notable sparsity in the angular and time domains, and suffer from severe path loss and absorption [3], which pose significant challenges to the localization task.

Manuscript received October 6, 2021; revised November 13, 2021; accepted November 15, 2021. Date of publication November 24, 2021; date of current version January 27, 2022. This work was supported in part by Fundação para a Ciência e a Tecnologia (FCT) under Grant UIDB/50021/2020. The work of Gabriel Falcao was supported by Fundação para a Ciência e a Tecnologia (FCT) and Instituto de Telecomunicações under Grant UIDB/EEA/50008/2020. The associate editor coordinating the review of this manuscript and approving it for publication was Yeong-Luh Ueng. (*Corresponding author: Mohammad Amin Maleki Sadr.*)

Mohammad Amin Maleki Sadr and Benoit Champagne are with the Department of Electrical and Computer Engineering, McGill University 3480 University Street, Montréal QC H3A 0E9, ON, Canada (e-mail: aminmalekisadr@gmail.com; benoit.champagne@mcgill.ca).

João Gante and Leonel Sousa are with the INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, 1000-029 Lisbon, Portugal (e-mail: joaofranciscocardosogante@gmail.com; las@inesc-id.pt).

Gabriel Falcao is with the Department of Electrical and Computer Engineering, University of Coimbra and Instituto de Telecomunicações, Coimbra QC H3A 0E9, Portugal (e-mail: gff@deec.uc.pt).

Digital Object Identifier 10.1109/LSP.2021.3130504

Several studies have addressed the problem of radio localization (or positioning) in mmWave bands for both indoor [4], and outdoor scenarios [5], [6]. Localization in these bands can be approached from two different perspectives, i.e.: model-based with the aid of geometry and radio propagation models [7], [8], and data-based, which emphasizes the use of machine learning techniques [9]–[13]. In recent years, deep learning in particular has attracted considerable interest for data-based localization. In [14], the authors introduce a convolutional neural network (CNN) to infer the position of a mobile user from channel state information (CSI) data, and use transfer learning to extend the acquired knowledge towards other scenarios. In [15], the use of CNN along with sequence-based deep learning is proposed to perform user localization at mmWave frequencies. The CNN is trained using a pre-established codebook of transmit beamforming (BF) vectors at the base station (BS), along with power delay profiles collected from a known set of user locations. Although these methods can estimate mobile user locations, they do not provide crucial information about the uncertainty regions.

The goal of model uncertainty estimation (MUE) is to produce a measure of confidence for statistical point estimators, which is of great importance in applications. In the deep learning literature, the so-called Bayesian Neural Network (BNN) is generally employed to determine probability that neural network (NN) predictions fall within a given region [16]. However, obtaining the posterior distribution needed for application of BNN is challenging, while providing an analytical expression is intractable [17]. To circumvent this difficulty, various approximation techniques have been introduced, including: variational inference [18], expectation propagation [19], Kalman filter variants [20], and Monte Carlo (MC) dropout [21]. Besides, a non-Bayesian method to extract uncertainty information from NN was proposed in [22].

Among these techniques, MC dropout exhibits important advantages, including, low-complexity (compared to original BNN) and high accuracy. MC dropout relies on the fact that as the layer width grows larger, a BNN reduces to a deep Gaussian process (GP) that admits a closed-form compositional kernel. Under this condition, the dropout rate applied before every dense layer in a NN can be interpreted as providing an approximation to the posterior distribution for the limiting GP [21]. Hence, GPs allow predictions from BNNs to be obtained more efficiently, and provide an analytic framework to understand deep learning models. Note that in contrast to standard dropout [21] which is applied only at training time, MC dropout is applied at both training and testing times, which in effect makes the output of the NN non-deterministic.

In this letter, we use MC dropout to capture the uncertainty of CNN-based methods for mmWave MIMO localization. Specifically, we employ MC dropout to obtain the second-order

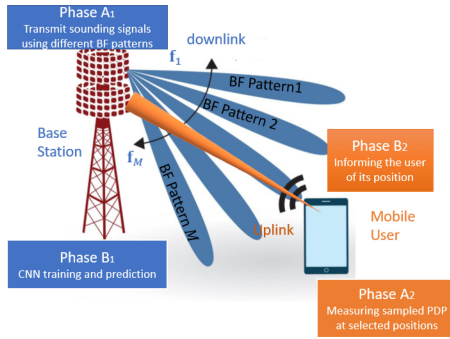


Fig. 1. Illustrating the different phases of CNN-based mmWave MIMO localization and corresponding exchange of information between the BS and mobile user.

statistics of the limiting GP for the CNN method in [15]. These statistics are used in turn to derive confidence intervals or uncertainty regions for the localization estimates. Thanks to its regularization effect, MC dropout contributes to reducing the inherent complexity while preserving a comparable accuracy to BNN. Employing a ray-tracing model of urban propagation at 28GHz, we show by simulations that the localization uncertainty regions can be properly determined via the proposed MC dropout method if sufficient data is available for training. Furthermore, the shape of such region depends on the maximum power received at the mobile user. While many studies have addressed wireless localization from the perspective of machine learning, none could be found that specifically investigate the related problem of uncertainty estimation in CNN-based localization, which is the main focus of this work.

The letter is organized as follows: In Section II, the system model of the underlying CNN-based localization method is reviewed. In Section III, the MC dropout method for MUE is introduced. In Section IV, simulation results are presented, followed by concluding remarks in Section V.

II. SYSTEM MODEL

The underlying CNN-based localization method consists of two phases [15]: A) data acquisition phase, and; B) learning phase. Fig. 1 illustrates the two phases and related sub-tasks.

A. Data Acquisition

Phase A, which consists of sub-tasks A₁ and A₂, performs the feature extraction. This phase is present for both of the offline and online modes, depending on the data collection being performed for training or test purposes, respectively. In A₁, a BS equipped with N_T antennas broadcasts channel-sounding signals over a given frequency band B using a fixed codebook of transmit beamforming (BF) vectors. The codebook is represented as $\mathcal{C} = \{\mathbf{f}_1, \dots, \mathbf{f}_M\}$, where $\mathbf{f}_i \in \mathbb{C}^{N_T \times 1}$ is the i th transmit BF vector (i.e., BF codebook index) and M is the codebook size. The signals broadcast by the BS are monitored by a user device equipped with N_R antennas whose location is known. The received signal at a given sounding frequency $f \in B$ corresponding to the i th BF vector can be written as:

$$r_i(f) = \mathbf{w}^H \mathbf{H} \mathbf{f}_i s + \mathbf{w}^H \mathbf{z}, \quad i \in \{1, 2, \dots, M\} \quad (1)$$

where $s \in \mathbb{C}$ is the amplitude of the sounding signal, $\mathbf{H} \in \mathbb{C}^{N_R \times N_T}$ is the transfer function of the MIMO wireless channel, $\mathbf{w} \in \mathbb{C}^{N_R \times 1}$ is the beamformer at the receiver and $\mathbf{z} \in \mathbb{C}^{N_R \times 1}$ is an additive noise term.

Subtask A₂ focuses on measuring the received power level at the user device. Specifically, given $r_i(f)$ for all $f \in B$, the power delay profile (PDP), $P_i(\tau)$ is extracted according to the procedure described in [23], where τ is the delay variable. By sampling $P_i(\tau)$ at the rate $F_s = \frac{1}{T_s}$, where T_s is the sampling period, we obtain $P_i[j] = P_i(jT_s), \forall j \in \{1, \dots, N_s\}$, where $N_s = \lfloor T/T_s \rfloor$ is the total number of samples and T is the maximum excess delay. Note that in order to distinguish signal from noise and also compressing the data, the PDP samples, $P_i[j]$, are represented by binary values defined as

$$x_{i,j} = \begin{cases} 1 & \text{if } P_i[j] \geq \eta \\ 0 & \text{else} \end{cases} \quad (2)$$

where η is the PDP threshold level. The use of binary PDP offers many advantages: it simplifies CNN implementation by reducing the complexity of calculations (for both training and testing) and allowing a smaller network size; it can also improve the localization performance under high noise levels [15]. The binary PDP values are used to define a feature matrix, i.e. $\mathbf{X} = [x_{i,j}]$ which is then transmitted to the BS.

For simplicity, we consider localization over a two-dimensional (2D) space, but generalization to 3D is straightforward. For the purpose of training, we assume that PDP data can be obtained in this way for the set of known mobile user positions distributed on a grid, and represented by vectors $\mathbf{y}_k \in \mathbb{R}^2 \forall k \in \mathcal{K} = \{1, \dots, K\}$, where K is the total number of positions. For the k th position, a feature matrix $\mathbf{X}_k \in \mathbb{R}^{M \times N_s}$ is obtained as explained above. Hence, for the k th position, \mathbf{X}_k and \mathbf{y}_k define the input and output data of the CNN, respectively. The set of all positions in the grid and associated feature matrices, denoted as $\mathbf{D} = \{(\mathbf{X}_k, \mathbf{y}_k) | k \in \mathcal{K}\}$, referred to as the BF fingerprint, is stored at the BS where it is used as a training data set.

B. Learning-Based Localization

The general purposes of phase B which consists of B₁ and B₂, are to train the CNN and use it to predict the position at BS and finally inform the mobile user of its position. The accuracy resulting from the proposed system ultimately depends on the learning capabilities of the inference block. Sub-task B₁ consists of both online and offline modes: in the offline mode the CNN is trained using collected training data, \mathbf{D} at the offline mode of phase A, and in the online mode, the CNN performs the prediction for the test data collected at the online mode of phase A. It is a supervised learning problem. Due to the nonlinear activation functions, a CNN is a good candidate to learn the complex phenomena commonly encountered in a mmWave transmission, such as reflections and diffraction [13]. To solve the localization, the network is trained using the well-known gradient-based algorithm for learning the CNN parameters. The device's position is relayed back to the user in phase B₂. An evaluation of the computational complexity of this CNN algorithm can be found in [15].

III. MC DROPOUT METHOD FOR MUE

We first discuss the approximation of the GP with MC dropout and then consider MUE based on this approximation.

A. MC Dropout as a GP Approximation

As mentioned earlier, solving the MUE problem for an arbitrary GP is intractable. Here, starting with the full GP, MC dropout can be used as an approximation for Bayesian uncertainty estimation. This view of dropout will allow us to

derive new probabilistic results in deep learning. Consider a NN model with L layers. Assume that at layer i th, $i \in 1, \dots, L$, the model applies a random weight matrix \mathbf{W}_i of dimension $K_i \times K_{i-1}$. Let $\mathbf{y} \in \mathbb{R}^{K_L}$ denote the output vector (or prediction) corresponding to an input feature vector $\mathbf{x} \in \mathbb{R}^{K_0}$. Each GP layer may also apply a bias vector $\mathbf{m}_i \in \mathbb{R}^{K_i}$. Let $\boldsymbol{\omega} = [\mathbf{W}_1, \mathbf{m}_1, \dots, \mathbf{W}_L, \mathbf{m}_L]$ represent the collection of weight parameters, which is a multi-dimensional random variable with prior distribution $p(\boldsymbol{\omega})$. For an arbitrary GP, we have

$$p(\mathbf{y} | \mathbf{x}, \boldsymbol{\omega}) = \mathcal{N}(\mathbf{E}(\mathbf{y} | \mathbf{x}, \boldsymbol{\omega}), \tau^{-1} \mathbf{I}) \quad (3)$$

$$\mathbf{E}(\mathbf{y} | \mathbf{x}, \boldsymbol{\omega}) = \sqrt{\frac{1}{K_L}} \mathbf{W}_L \sigma\left(\dots \sqrt{\frac{1}{K_2}} \mathbf{W}_2 \sigma(\mathbf{W}_1 \mathbf{x} + \mathbf{m}_1) \dots\right) \quad (4)$$

where σ is NN activation function and $\tau > 0$ is the model precision hyper-parameter. The predictive distribution for an input data \mathbf{x} is equivalent to using an ensemble of the infinite number of neural networks with various configuration of the weights, which is:

$$p(\mathbf{y} | \mathbf{x}, \mathbf{D}) = \int p(\mathbf{y} | \mathbf{x}, \boldsymbol{\omega}) p(\boldsymbol{\omega} | \mathbf{D}) d\boldsymbol{\omega} \quad (5)$$

where $p(\boldsymbol{\omega} | \mathbf{D})$ is the conditional or posterior distribution on weights. It can be inferred from (5) that for a given data, in order to find the prediction value, we must examine the proposed model in all the existing possible models.

The posterior distribution $p(\boldsymbol{\omega} | \mathbf{D})$ is intractable. In this case, the only possible solution is approximating $p(\boldsymbol{\omega} | \mathbf{D})$. Instead of finding $p(\boldsymbol{\omega} | \mathbf{D})$, we define an approximated variational distribution $q(\boldsymbol{\omega})$, whose structure is easy to evaluate. We would like the approximated distribution to be as close as possible to the posterior distribution obtained from the full GP. Thus, we minimize the Kullback–Leibler (KL) divergence $\text{KL}(q(\boldsymbol{\omega}), p(\boldsymbol{\omega} | \mathbf{D}))$, which is intuitively a measure of similarity between two distributions [21]. One of the solutions for approximating $p(\boldsymbol{\omega} | \mathbf{D})$, is called MC dropout. In this solution, we use $q(\boldsymbol{\omega})$ to approximate the intractable posterior. In fact, $q(\boldsymbol{\omega})$ can be defined in terms of the random quantities:

$$\begin{aligned} \mathbf{W}_i &= \text{diag}([z_{i,j}]_{j=1}^{K_i}) \mathbf{M}_i \\ z_{i,j} &\sim \mathcal{B}(p_i) \quad \forall i \in \{1, \dots, L\}, j \in \{1, \dots, K_{i-1}\} \\ q(\boldsymbol{\omega}) &\sim \mathcal{N}\left(\mathbf{E}(\mathbf{y} | \mathbf{x}, \boldsymbol{\omega}), \frac{1}{\tau} \mathbf{I}\right) \text{diag}(\mathbf{z}_1, \dots, \mathbf{z}_L) \end{aligned} \quad (6)$$

where p_i and matrices \mathbf{M}_i of dimensions $K_i \times K_{i-1}$ are variational parameters and \mathcal{B} stands for Bernoulli distribution. The binary variable $z_{i,j} = 0$ corresponds to unit j in layer $i - 1$ being dropped out as an input to layer i and also $\mathbf{z}_i = [z_{i,1}, \dots, z_{i,K_i}]^T$. The $\text{KL}(q(\boldsymbol{\omega}), p(\boldsymbol{\omega} | \mathbf{D}))$, which is our minimization objective, is shown to be [21]:

$$\mathcal{L}_{\text{MC}} = - \int q(\boldsymbol{\omega}) \log p(\mathbf{y} | \mathbf{X}, \boldsymbol{\omega}) d\boldsymbol{\omega} + \text{KL}(q(\boldsymbol{\omega}), p(\boldsymbol{\omega})) \quad (7)$$

where the first term is the loss function and the second term represents the regularization. Minimizing (7) over $\boldsymbol{\omega}$ will result in a variational distribution $q(\boldsymbol{\omega})$ that explains the data well (as obtained from the first term) while still being close to the prior and preventing the model from over-fitting (regularization term). We rewrite the loss function term of (7) with Monte Carlo sampling over $\boldsymbol{\omega}$ with a single sample as:

$$\mathcal{L}_{\text{MC}}^{(1)} = \frac{1}{N} \sum_{n=1}^N \log p(\mathbf{y}_n | \mathbf{x}_n, \hat{\boldsymbol{\omega}}_n) \quad (8)$$

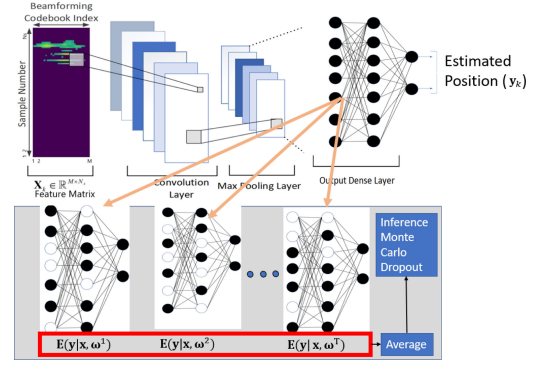


Fig. 2. CNN-based and MC dropout method for user position estimation and uncertainty using beamformed fingerprint data.

TABLE I
NUMBER OF LEARNABLE PARAMETERS, TRAINING TIME, AND INFERENCE THROUGHPUT

| Architecture | Parameters | Training Time | Throughput |
|--------------|--------------------|---------------|--------------------------------|
| CNN [15] | 3.37×10^6 | 287 min | 19.15×10^3 predict./s |
| MC dropout | 3.37×10^6 | 287 min | 47.8×10^4 predict./s |

Note that $\hat{\boldsymbol{\omega}}_n$ is not maximum a posteriori estimate, but random variable realizations from the Bernoulli distribution, $\hat{\boldsymbol{\omega}}_n \sim q(\boldsymbol{\omega})$, which is identical to applying MC dropout to the weights of the network. In [21], it is shown that the second term in (7) is equivalent to $\sum_{i=1}^L \|\boldsymbol{\omega}_i\|_2^2$ with some modifications. Thus, the objective function can be written as:

$$\mathcal{L}_{\text{MC}} = -\lambda_1 \mathcal{L}_{\text{MC}}^{(1)} + \lambda_2 \left(\sum_{i=1}^L p_i \|\mathbf{M}_i\|_2^2 + \|\mathbf{m}_i\|_2^2 \right) \quad (9)$$

where $\lambda_1 = \frac{1}{N\tau}$, and $\lambda_2 = \frac{l^2}{2\tau N}$. The precision hyper-parameters τ and length-scale l are tuned by grid search. As seen in Fig. 2, the convolution layers and maxpooling operations remain untouched during the MC dropout method. In the dense layer, MC dropout is performed at prediction time. The random neurons in each layer are dropped out, according to the associated probability, from the base neural network model to create another model. The theoretical results in [24] reveal that for a NN with one dense layer, dropout can reduce the Rademacher complexity by a factor of p^2 . Table I shows the computational savings resulting from our proposed approach compared with the original CNN method in [15].

B. Uncertainty Estimation

As we discussed, the model uncertainty can be obtained from the MC dropout method. Approximated predictive distribution is given by [21]:

$$q(\mathbf{y} | \mathbf{x}) = \int p(\mathbf{y} | \mathbf{x}, \boldsymbol{\omega}) q(\boldsymbol{\omega}) d\boldsymbol{\omega} \quad (10)$$

To obtain the first moment for the GPs, we use T sets of MC dropout realizations from the Bernoulli distribution as

$$\mathbf{E}_{q(\mathbf{y} | \mathbf{x})}(\mathbf{y}) \approx \frac{1}{T} \sum_{t=1}^T \mathbf{E}(\mathbf{y} | \mathbf{x}, \boldsymbol{\omega}^t) \quad (11)$$

In practice, this is equivalent to performing T stochastic forward passes through the network and averaging the results (Fig. 2). Note that the mean of the dropout realizations is interpreted as the network prediction. We can estimate the second moment in

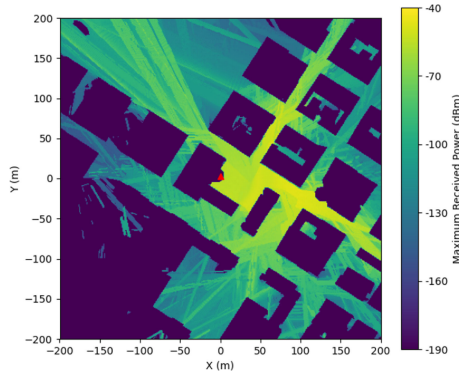


Fig. 3. Maximum received power from the ray-tracing simulation in the NYU area.

the same way ($\phi = q(\mathbf{y}|\mathbf{x})$):

$$\mathbb{E}_{\phi}(\mathbf{y}^T \mathbf{y}) \approx \tau^{-1} \mathbf{I} + \frac{1}{T} \sum_{t=1}^T \mathbb{E}(\mathbf{y}|\mathbf{x}, \omega^t)^T \mathbb{E}(\mathbf{y}|\mathbf{x}, \omega^t) \quad (12)$$

The model's predictive variance is

$$\text{var}_{\phi}(\mathbf{y}) = \mathbb{E}_{\phi}(\mathbf{y}^T \mathbf{y}) - \mathbb{E}_{\phi}(\mathbf{y})^T \mathbb{E}_{\phi}(\mathbf{y}) \quad (13)$$

IV. NUMERICAL RESULTS¹

We use the mmWave data generated by the Wireless InSite ray-tracing simulator [25] and a high precision open-source 3D map of the NYU area [26], containing BF fingerprint data from 401×401 positions. The specific propagation parameters are inherited from the experimental measurements in [27] while in [28], it is shown that these ray-tracing simulations adequately match experimental observations. Fig. 3, shows the ray-tracing measurements with a transmit power of 30dBm, which corresponds to the maximum received power for all possible transmit BF vectors. In this simulation, the carrier frequency is 28GHz, BF fingerprint codebook size is $M = 32$, and the transmitter is at position (0,0). In order to account for possible mismatch in the data set (measurement uncertainty), noise is added to the ray tracing data following a log-normal distribution with 6dB standard deviation. The robustness of the CNN-based localization method with respect to measurement noise in the PDP data is further discussed in [15]. The following configuration is assumed: one convolution layer with 8, 3×3 filters, followed by 2×1 max-pooling, 7 hidden layers (1024 neurons each), 2 outputs, and the MC dropout rate set to 0.2.

For each one of 10 selected positions, Fig. 4 illustrates the uncertainty estimation using 1000 MC dropout points (10^4 MC Dropout points in total). Let $0 \leq e \leq 1$ denote the eccentricity of the shape of the level contour curves of the GPs. At positions with low received power, e tends to one, and by increasing the received power, e is decreased. Also, at high transmitted power, the mean of MC dropout inference values is close to the true value, which was also expected. A confidence ellipse (with corresponding main axis lengths) is shown in the bottom left corner of Fig. 4, where λ_x, λ_y are the eigenvalues of the covariance matrix and $\eta = 5.99$ is the 95% percentile of the chi-square distribution with two degrees of freedom [29]. Fig. 5 depicts the Pearson correlations [30] ($-1 \leq PC \leq 1$) for all

¹The Code and Data Can Be Found At <https://github.com/gante/mmWave-Localization-Learning>.

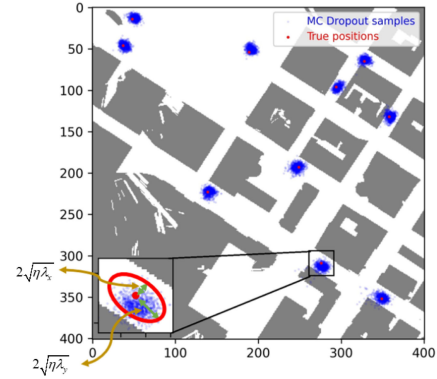


Fig. 4. Uncertainty estimation using 1000 MC dropout points for 10 true positions.

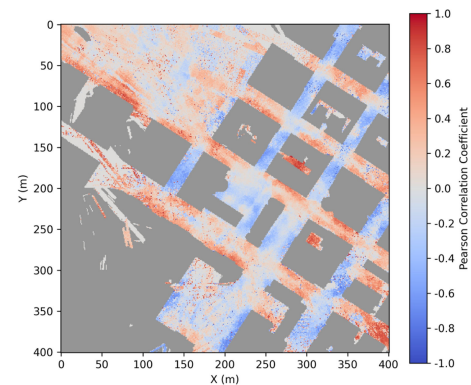


Fig. 5. PC using 1000 MC dropout points for each possible position in the grid.

possible positions ($401 \times 401 = 160801$) in the grid. For each such position in Fig. 5, we train the network, test it to produce 1000 MC dropout points, and then calculate PC for that position. We note that the shapes of the uncertainty regions learned from the data for each true position in Fig. 4 are consistent with the corresponding PC values in Fig. 5. For example, the displayed ellipse for true position (280,310) has major axis pointing along south-east, which is the corresponding value of 0.65 for the PC.

V. CONCLUSION

In this letter, we proposed a method for capturing the uncertainty in mobile user localization for the MIMO mmWave outdoor localization system. Using MC dropout algorithm, we estimated the uncertainty of the position. We showed that MC dropout can be represented as an approximation to GPs to predict the MUE. Simulation results show that the shape of the uncertainty region depends on the maximum power received at the mobile user. The trade-off between uncertainty region accuracy and computational complexity of MC-dropout and other BNN approximations for mmWave localization will be explored in future work.

REFERENCES

- [1] T. Bai and R. W. Heath, "Coverage and rate analysis for millimeter-wave cellular networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 2, pp. 1100–1114, Feb. 2015.

- [2] A. Ghosh, A. Maeder, M. Baker, and D. Chandramouli, "5G evolution: A view on 5G cellular technology beyond 3GPP release 15," *IEEE Access*, vol. 7, pp. 127 639–127 651, 2019.
- [3] S. Sun, T. S. Rappaport, R. W. Heath, A. Nix, and S. Rangan, "MIMO for millimeter-wave wireless communications: Beamforming, spatial multiplexing, or both," *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 110–121, Dec. 2014.
- [4] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 763–776, Jan. 2017.
- [5] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems," *IEEE Access*, vol. 6, pp. 37 328–37 348, 2018.
- [6] O. Kanhere and T. S. Rappaport, "Position locationing for millimeter wave systems," in *Proc. IEEE Global Commun. Conf.*, Dec. 2018, pp. 206–212.
- [7] M. Koivisto *et al.*, "Joint device positioning and clock synchronization in 5G ultra-dense networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 2866–2881, May. 2017.
- [8] M. Koivisto, A. Hakkarainen, M. Costa, P. Kela, K. Leppanen, and M. Valkama, "High-efficiency device positioning and location-aware communications in dense 5G networks," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 188–195, Aug. 2017.
- [9] S. De Bast and S. Pollin, "MaMIMO CSI-based positioning using CNNs: Peeking inside the black box," in *Proc. IEEE Int. Conf. Commun. Workshops*, 2020, pp. 1–6.
- [10] J. Vieira, E. Leitingner, M. Sarajlic, X. Li, and F. Tufvesson, "Deep convolutional neural networks for massive MIMO fingerprint-based positioning," in *Proc. IEEE Int. Symp. Pers. Indoor Mobile Radio Commun.*, 2017, pp. 1–6.
- [11] M. Arnold, S. Dorner, S. Cammerer, and S. Brink, "On deep learning-based massive MIMO indoor user localization," in *Proc. IEEE Workshop Signal Process. Adv. Wireless Commun.*, 2018, pp. 1–5.
- [12] Z. Wei, Y. Zhao, X. Liu, and Z. Feng, "DoA-LF: A location fingerprint positioning algorithm with millimeter-wave," *IEEE Access*, vol. 5, pp. 22 678–22 688, 2017.
- [13] J. Gante, L. Sousa, and G. Falcao, "Dethroning GPS: Low-power accurate 5G positioning systems using machine learning," *IEEE Trans. Emerg. Sel. Topics Circuits Syst.*, vol. 10, no. 2, pp. 240–252, Jun. 2020.
- [14] S. De Bast, A. P. Guevara, and S. Pollin, "CSI-based positioning in massive MIMO systems using CNN," in *Proc. IEEE Veh. Technol. Conf.*, 2020, pp. 1–5.
- [15] J. Gante, G. Falcão, and L. Sousa, "Deep learning architectures for accurate millimeter wave positioning in 5G," *Neural Process. Lett.*, vol. 51, no. 1, pp. 487–514, 2020.
- [16] A. Loquercio, M. Segu, and D. Scaramuzza, "A general framework for uncertainty estimation in deep learning," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3153–3160, Apr. 2020.
- [17] L. V. Jospin, W. Buntine, F. Boussaid, H. Laga, and M. Bennamoun, "Hands-on Bayesian neural networks—A tutorial for deep learning users," *ACM Comput. Surv.*, Jul. 14 2020, *arXiv:2007.06823*.
- [18] A. Graves, "Practical variational inference for neural networks," in *Proc. Adv. Neural Inform. Process. Syst.*, 2011, vol. 24, pp. 2348–2356.
- [19] T. Bui, D. Hernández-Lobato, J. Hernandez-Lobato, Y. Li, and R. Turner, "Deep Gaussian processes for regression using approximate expectation propagation," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1472–1481.
- [20] D. T. Mirikitani and N. Nikolaeov, "Dynamic modeling with ensemble Kalman filter trained recurrent neural networks," in *Proc. IEEE 7th Int. Conf. Mach. Learn. App.*, 2008, pp. 843–848.
- [21] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2016, vol. 48, pp. 1050–1059.
- [22] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6405–6416.
- [23] K. Guan *et al.*, "Channel sounding and ray tracing for intrawagon scenario at mmWave and sub-mmWave bands," *IEEE Trans. Antennas Propag.*, vol. 69, no. 2, pp. 1007–1019, Feb. 2021.
- [24] W. Gao and Z.-H. Zhou, "Dropout Rademacher complexity of deep neural networks," *Sci. China Inf. Sci.*, vol. 59, no. 7, pp. 1–12, Jul. 2016.
- [25] Remcom, "Wireless insite," Accessed: Jun. 2018. [Online]. Available: <http://www.remcom.com/wireless-insite>
- [26] "Open source New York 3D building model." Accessed: Apr. 9, 2021. [Online]. Available: <http://www1.nyc.gov/site/doitt/initiatives/3d-building>
- [27] Y. Azar *et al.*, "28GHz propagation measurements for outdoor cellular communications using steerable beam antennas in New York City," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2013, pp. 5143–5147.
- [28] J. Gante, G. Falciao, and L. Sousa, "Data-aided fast beamforming selection for 5G," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 1183–1187.
- [29] O. Erten and C. V. Deutsch, "Combination of multivariate Gaussian distributions through error ellipses," *Geostatistics Lessons*, 2020. [Online]. Available: <http://www.geostatisticslessons.com/lessons/errorellipses>
- [30] J. Benesty, J. Chen, and Y. Huang, "On the importance of the Pearson correlation coefficient in noise reduction," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 4, pp. 757–765, May. 2008.