This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2019.2899571, IEEE Wireless Communications Letters



# LIDAR Data for Deep Learning-Based mmWave Beam-Selection

Aldebaro Klautau, Nuria González-Prelcic and Robert W. Heath Jr.

***Abstract*—Millimeter wave communication systems can leverage information from sensors to reduce the overhead associated with link configuration. LIDAR (light detection and ranging) is one sensor widely used in autonomous driving for high resolution mapping and positioning. This paper shows how LIDAR data can be used for line-of-sight detection and to reduce the overhead in millimeter wave beam-selection. In the proposed distributed architecture, the base station broadcasts its position. The connected vehicle leverages its LIDAR data to suggest a set of beams selected via a deep convolutional neural network. Co-simulation of communications and LIDAR in a vehicle-to-infrastructure (V2I) scenario confirm that LIDAR can help configuring mmWave V2I links.**

***Keywords*—LIDAR, mmWave, machine learning, deep learning, convolutional networks.**

## I. Introduction

Millimeter wave (*mmWave*) is a key technology for sharing high rate sensor data for connected and automated vehicles [1]. Prior work has shown that position information obtained from vehicles can be used to reduce the overhead required to establish mmWave links [1]–[5]. In this paper, we show how LIDAR provides an additional source of information to reduce communication overhead. The LIDAR uses a laser to scan an area and measure the time delay from the backscattered signal. This data is then converted into points in space and interpreted as three-dimensional (3D) images with pixels indicating relative positions from the sensor [6]. LIDAR data can be exploited without additional cost for improved communications when it is already used on a automated vehicle for mapping, positioning, or obstacle detection.

Reducing the beam-selection overhead is important in cellular and WiFi systems operating at mmWave frequencies [6]–[8]. Out-of-band measurements were used for improved beam-selection in mmWave communications in [9], [10]. The benefit of a radar located in infrastructure was investigated in [11]. The use of position information in V2I mmWave was studied in [1]–[5]. Some work using position, targeted only line-of-sight (LOS) situations [3], [12]. Non-LOS (NLOS) was investigated in [2], [4] with measurement *fingerprint* databases. Prior work has established that position information can reduce mmWave beam-selection overheads, and that machine learning (ML) is a good tool for this problem. But the performance of previously proposed systems is limited by the *penetration rate* of connected vehicles. The use of LIDAR, which is popular for automated cars, has not been considered, nor have decentralized architectures for applying ML to beam-selection problems.

In this paper, we develop a *distributed* architecture for reducing mmWave beam-selection overhead. We assume the BS broadcasts its position via a low-frequency control channel (CC), and all processing is performed by the connected vehicle. The vehicle uses its LIDAR data, its own position, and the broadcasted BS position, to estimate a set of $M$ candidate beam pairs that are informed to the BS through the CC. The recommended beam pairs are then trained by the BS, and the best one is chosen for data transmission. Our system uses only the LIDAR and position information for the prediction; fusion with other sensors and direct performance comparison with centralized architectures [1]–[5] is a topic of future work.

We use ML to solve two key problems in our LIDAR-aided mmWave system. First, we develop a predictor to assess whether the channel is in LOS or NLOS. LOS detection is useful because beam-selection is easier in the LOS setting. Second, we use deep learning (DL) [13] with a neural network trained to perform top-$M$ classification [14] conditioned on LOS and NLOS state estimates. We take this approach instead of alternatives such as *subset ranking* [15] because all $M$ selected beams are evaluated in the subsequent stage and, consequently, their local rank is irrelevant.

We present simulation results obtained with a methodology that combines a traffic simulator to model realistic mobility scenarios with integrated ("paired") data from ray-tracing (for estimating mmWave channels) and LIDAR simulators. Our results indicate that the beam-selection overhead can be reduced by factors of 12x in LOS and 2x in NLOS, without reduction of throughput or by larger factors if some reduction is acceptable. Compared with prior work [1]–[5], we consider LIDAR on the vehicle as an additional sensor. We also use DL because of its promising results for position-based beam-selection [16] and many other domains [13], [14]. An advantage of our distributed architecture versus [1]–[5] is that it does not depend on the penetration rate of connected vehicles, as it only uses the LIDAR of the connecting vehicle. Because unconnected cars do not report their positions to the BS, the beam-selection performance in a centralized architecture decreases with the number of unconnected vehicles [5].

## II. System Model

We consider a downlink OFDM mmWave system with analog beamforming [10]. Both transmitter and receiver have

This material is based upon work supported in part by the National Science Foundation under Grant No. ECCS-1711702 and Grant No. CNS-1731658, as well as gifts from Nokia Bell Labs and Toyota ITC. The work of A. Klautau was supported in part by CNPq, Brazil (201493/2017-9/PDE).

A. Klautau (aldebaro@ufpa.br) is with UFPA, Belem, PA, 66075, Brazil.

N. González-Prelcic (ngprelcic@utexas.edu) and R. Heath (rheath@utexas.edu) are with the Wireless Networking and Communications Group, The University of Texas at Austin, Austin, TX 78712, USA.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2019.2899571, IEEE Wireless Communications Letters



antenna arrays with only one radio frequency (RF) chain and fixed beam codebooks. To simulate the channel, we use ray-tracing data and combine the ray-tracing output with a *wideband* mmWave *geometric channel model* as, e. g., in [10]. Assuming $R_c$ multipath components (MPC) per transmitter / receiver pair, the information collected from the outputs for the $r$-th MPC of a given pair is: complex path gain $\alpha_r$, time delay $\tau_r$ and angles $\phi_r^D$, $\theta_r^D$, $\phi_r^A$, $\theta_r^A$, corresponding respectively to azimuth and elevation for departure and arrival. The frequency-selective channel model at the time instant corresponding to the $n$-th symbol vector is described in detail in [Section III] [10], which also includes the definition of the model in the frequency domain $\mathbf{H}[k]$, where $k$ is the subcarrier index.

We assume beam codebooks $\mathcal{C}_t = \{\mathbf{f}_1, \cdots, \mathbf{f}_{|\mathcal{C}_t|}\}$ and $\mathcal{C}_r = \{\mathbf{w}_1, \cdots, \mathbf{w}_{|\mathcal{C}_r|}\}$ at the transmitter and the receiver sides, with no restriction on the codebook size (e. g., they do not have to be DFT codebooks). For a given pair $(p, q)$ of vectors, representing precoder $\mathbf{f}_p$ and combiner $\mathbf{w}_q$, the received signal at subcarrier $k$ is $\mathbf{s}[k] = \mathbf{w}_q^H \mathbf{H}[k] \mathbf{f}_p$, where $H$ denotes conjugate transpose. The beam-selection is guided by the normalized signal power

$$y_{(p,q)} = \sum_{k=0}^{K-1} |\mathbf{w}_q^H \mathbf{H}[k] \mathbf{f}_p|^2 \qquad (1)$$

and the *optimum* beam pair is $\widehat{(p,q)} = \arg\max_{(p,q)} y_{(p,q)}$. In this paper, the goal of beam-selection is to recommend a set $\mathcal{B} = \{(p_i, q_i)\}_{i=1}^M$ such that $\widehat{(p,q)} \in \mathcal{B}$.

## III. Machine Learning using LIDAR Data

### A. Information exchange protocol

We develop a ML-based *beam-selection* strategy for V2I mmWave cellular communication system, assuming that the connected vehicle is equipped with a LIDAR. The proposed ML-based protocol is illustrated in Fig. 1. It is assumed that the BS can broadcast its absolute position $P_b = (x_b, y_b, z_b)$ for mmWave V2I beam alignment of incoming vehicles using a CC provided by, for instance, DSRC signals or as part of the BS CC [6]. A vehicle estimates its position $P_v = (x_v, y_v, z_v)$ using for example, *Global Positioning System* (GPS) or a simultaneous localization and mapping (SLAM) algorithm [17]. The BS also broadcasts its *coverage zone* $Z = (x_z^b, y_z^b, x_z^e, y_z^e, h)$, which is a cuboid specified by its height $h$, and points $(x_z^b, y_z^b)$ and $(x_z^e, y_z^e)$ denoting the base.

The ML algorithm is executed at the vehicle and outputs a set $\mathcal{B}$ of beam pairs. After the initialization stage (see Fig. 1), the BS transmits using the beams specified by $\{p_i\}_{i=1}^M$ and the $M$ pairs of beams are evaluated at the vehicle according to (1). The best pair is then fed back to the BS. If *beam correspondence* can be assumed, the same beam pair can be used for uplink. Once mmWave communication links are established, the overhead information required by beam tracking can rely on the high data rates of mmWave links.

### B. LIDAR-based feature extraction and deep learning

We use ML to tackle two distinct problems. The first is the use of only LIDAR data for LOS versus NLOS binary classification. The second problem is the selection of the top-$M$ beam pairs based on (1) for decreasing the beam-selection overhead, which is associated with the protocol explained in the previous subsection. The raw input data to solve both problems is composed of the LIDAR point cloud $\mathbf{C}$ collected by the vehicle, the BS coverage zone $Z$ and positions $P_v$ and $P_b$. The LIDAR cloud $\mathbf{C}$ is an array of dimension $D \times 3$, composed of 3D points indicating the presence of obstacles. Typical values of $D$ are relatively large and using an alternative representation helps to control the computational cost.

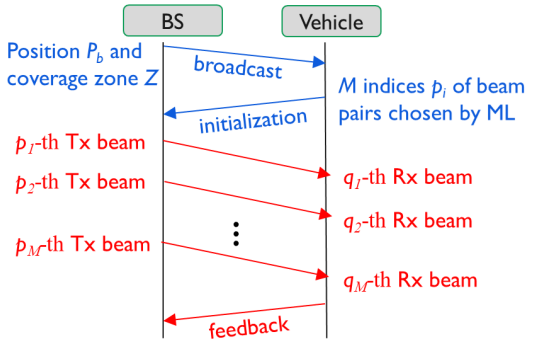


Fig. 1. Timing diagram for the distributed LIDAR-based beam-selection method. The first phase (broadcast and initialization) uses a low-frequency CC while the second corresponds to mmWave communication.

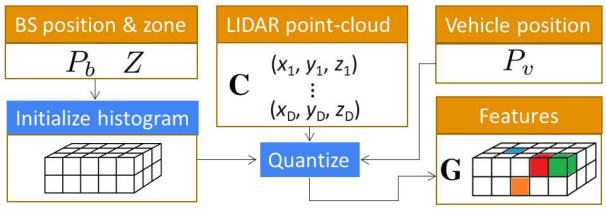


Fig. 2. Feature extraction of 3D histogram $\mathbf{G}$ from LIDAR data.

Given the absolute positions of $Z$, we create a 3D histogram $\mathbf{G}$ in which a bin corresponds to a fixed region of $Z$ (Fig. 2). Each element of $\mathbf{G}$ stores the number of elements of $\mathbf{C}$ within the corresponding bin. This histogram calculation was implemented as the uniform quantization of elements of $\mathbf{C}$ using $(b_x, b_y, b_z)$ bits, with $Z$ providing the quantizers' dynamic ranges. Before quantization, the relative distances of obstacles provided by $\mathbf{C}$ are converted into absolute positions $\mathbf{C} + P_v$. For example, considering only the $x$-axis for simplicity, if $x_z^b = 10$ m, $x_z^e = 22$ m and $b_x = 2$ bits, the four bins are represented by values $10, 14, 18$ and $22$. Assuming $P_v = (5, y', z')$, an element $(16, y'', z'')$ in $\mathbf{C}$ is mapped to $(21, y' + y'', z' + z'')$. The value 21 is then quantized to 22, and this point accounted by incrementing a bin of $\mathbf{G}$ that corresponds to the last (4th) position in the $x$-axis. The element of $\mathbf{G}$ corresponding to $P_b$ is the same in all examples, given the grid uses absolute positions. We discard points that are farther from $P_v$ by more than a certain distance $d_{\max}$. The ML input feature is then a histogram $\mathbf{G}$ with dimension $2^{b_x} \times 2^{b_y} \times 2^{b_z}$.

For both problems (LOS decision and beam-selection), we adopted neural networks with 13 layers from which 7 are 2D *convolutional* layers with decreasing kernel sizes, from $13 \times 13$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2019.2899571, IEEE Wireless Communications Letters



to $1 \times 1$, trained with Kera's *Adadelta* optimizer [14]. We used *pooling* layers and, to mitigate overfitting, *regularization* and *dropout*. For beam-selection, the values in (1) below 6 dB from the maximum were zeroed and normalized to have unitary sum. For top-$M$ classification, the output layer had a *softmax* activation function and a *categorical cross-entropy* as loss function [14]. For binary classification, the output layer and loss were *sigmoid* and *binary cross-entropy*, respectively [14]. The number of parameters per network is approximately $10^5$. To improve reproducibility, we share code and data at [18].

As a baseline for comparing with DL applied to the LOS decision problem, we also evaluated a simple geometric approach: given $P_b$ and $P_v$, we calculate the line $\mathcal{L}$ connecting them. We denote by $\hat{d}$ the *minimum distance* between any point in $\mathbf{C}$ to $\mathcal{L}$. A *decision stump* classifier [14] uses a threshold $\gamma$ to decide for NLOS if $\hat{d} < \gamma$ or LOS otherwise. The intuition is that if $\mathcal{L}$ is far from all obstacles identified by the LIDAR in $\mathbf{C}$, the link is potentially LOS.

## IV. Numerical Results

### A. Simulation methodology

Aiming at realistic datasets, we adopted a simulation methodology using traffic, ray-tracing and LIDAR simulators in V2I mmWave communications [16]. We paired the simulations of the mmWave communication system and the LIDAR data acquisition integrating three softwares: the *Blender Sensor Simulation* (BlenSor) [19], the *Simulation of Urban MObility* (SUMO) traffic simulator [20], both open source, and Remcom's Wireless InSite for ray-tracing. In the configuration stage, the user provides information about the objects in the 3D scenario, lanes coordinates, eletromagnetic parameters, etc. The software execution is based on a Python *orchestrator* code that invokes SUMO and converts its ouputs (vehicles positions, orientations, etc.) to formats that can be interpreted by distinct simulators. The orchestrator then invokes the simulators (LIDAR and ray-tracing in this case) to obtain paired results.

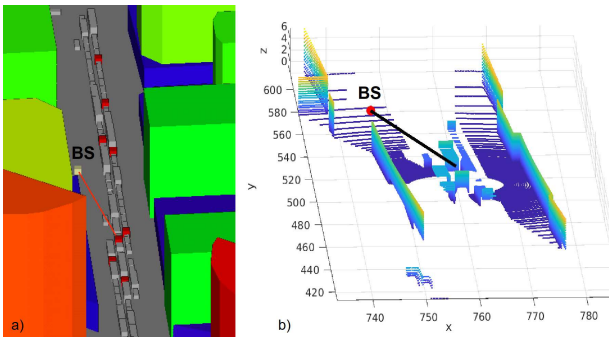


Fig. 3. a) Urban canyon 3D scenario with vehicles of distinct sizes randomly positioned. The building color indicates height and corresponds to a range from 0 (blue) to 101 meters (red). b) Corresponding LIDAR point cloud. The LOS rays between the BS antenna at $z = 4$ m and vehicle are shown.

Fig. 3a depicts the adopted urban canyon 3D scenario, which is part of Wireless InSite's examples and represents a region of Rosslyn, Virginia. The *study area* is a rectangle of approximately $337 \times 202$ m$^2$ and the BS antenna array height is $z = 4$ m. We placed receivers and LIDARs on top of all connected vehicles (identified in red) in each scene snapshot. Fig. 3b illustrates an example of the corresponding LIDAR *point cloud*. Lines between BS and vehicle are also shown, and suggest a LOS channel.

The ray-tracing simulations used a maximum of $L = 25$ MPCs per transmitter / receiver pair, isotropic antennas, 60 GHz carrier frequency, $B = 100$ MHz, $K = 64$ subcarriers, and enabled ray-tracing diffuse scattering. Other parameters followed the ones in [16].

The downlink mmWave massive MIMO relied on a BS with a $16 \times 16$ uniform planar array (UPA) and vehicles with $4 \times 4$ UPAs. When designing $\mathcal{C}_t$ and $\mathcal{C}_r$, we first augmented the conventional DFT codebook with steered codevectors, linear combinations of codevectors and random vectors from Grassmannian codebooks. From this large initial set, we kept only the codevectors that were chosen as $\widehat{(p, q)}$ more than 100 times in the training set. This procedure led to $|\mathcal{C}_t| = 20$ and $|\mathcal{C}_r| = 12$, respectively. Hence, the number of classes for top-$M$ classification is 240.

The LIDAR simulations assumed a Velodyne model HDL-64E2 scanner positioned at a height $z = 1$ m from the top-center of the vehicle. The angle resolution was 0.1728 degrees and the rotation speed 10 Hz. The experiments adopted $b_x = b_y = 6$ and $b_z = 3$ bits. We eliminated from $\mathbf{C}$ the points with small values in the $z$-axis ($< 0.1$ m), which correspond to ground reflections (see Fig. 3b), and also the points with a distance from the LIDAR larger than $d_{\max} = 25$ m.

A preliminary investigation indicated consistent beam-selection results for distinct signal-to-noise ratios and, for simplicity, the results here are for a noise-free mmWave channel. But we considered two conditions with respect to positioning accuracy: *noise-free* and *noisy*. The LIDAR noise [19] is assumed to have independent components distributed according to a zero-mean Gaussian $\mathcal{N}(0, \sigma_L^2/3)$ with variance $\sigma_L^2/3$. For the *noisy* condition, we adopted the HDL-64E2 default value of $\sigma_L = 0.1$ m. Similarly, the accuracy of the *Global Navigation Satellite System* (GNSS) technology is modeled assuming the elements of the position error vector are independent and identically distributed according to $\mathcal{N}(0, \sigma_G^2/3)$ (no bias). Conventional GPS may lead to errors of 3 to 5 m, while sophisticated SLAMs can help to keep the error below 50 cm in the horizontal plane [17]. For the *noisy* condition, we assumed $\sigma_G = 3$ m and $\sigma_L = 0.1$ m.

Beam-selection is harder in NLOS because the predictability decreases considerably when compared to LOS cases. If an experiment considers both LOS and NLOS channels, the accuracy of ML will depend on the blockage probability, which is heavily influenced by traffic statistics, large vehicles (potential blockers) and antenna height. Numerical results of distinct experiments that used mixed LOS and NLOS are harder to compare and the ML models may be biased by the easier LOS cases. To avoid this situation, we present separate evaluations of beam-selection for each case. The mmWave data is composed of $N_L = 6,482$ LOS and $N_N = 4,712$ NLOS channel examples. The beam-selection experiments used $N_L$ and $N_N$ examples in the LOS and NLOS evaluations, respectively, while LOS detection used all $N_L + N_N$ examples. For all experiments we created disjoint test and training sets

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2019.2899571, IEEE Wireless Communications Letters



with 20% and 80% of the examples, respectively.

### B. Results

The *accuracy* of both binary and top-$M$ classifiers improve considerably when the elevation angle of the LIDAR is adjusted for communications (points to the BS antenna). We did not perform this adjustment and used the HDL-64E2 default elevation. This increases the chances that the LIDAR does not detect a LOS blocker because it is obstructed by a neighbor vehicle. For the LOS detection in noise-free condition, the minimum achieved misclassification error with the geometry-based stump was 24% while DL leads to 10%. For other conditions, DL outperforms the baseline by a larger margin.

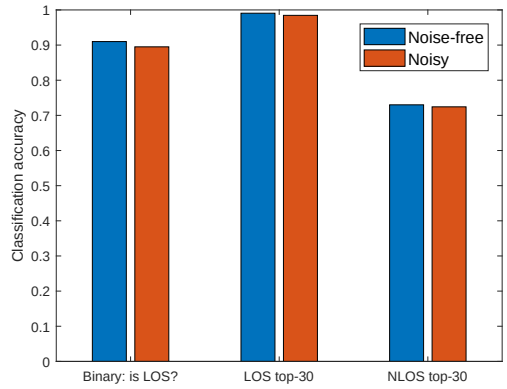


Fig. 4. Accuracy for LOS detection (binary problem) and beam-selection using top-$M$ classification with $M = 30$ for LOS and NLOS examples. The performance for both noise-free and noisy conditions are shown.

Fig. 4 presents the results using DL for LOS detection and the two cases of top-$M$ beam-selection for both (positioning) noise scenarios. It can be seen that the adopted noisy condition did not lead to significant loss of accuracy. As expected, the performance in NLOS is considerably lower than for LOS. Due to the difficulty of dealing with NLOS, the binary problem has worse performance than top-30 LOS classification.

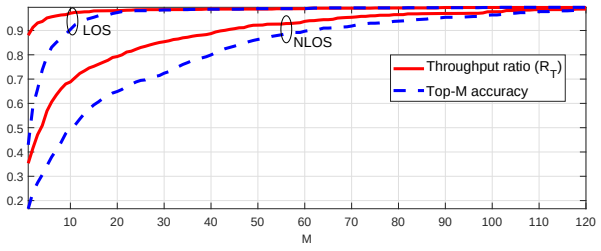


Fig. 5. LOS and NLOS top-$M$ classification for beam-selection with 240 beam pairs in noisy condition for $M = 1, \ldots, 120$.

While Fig. 4 shows results for $M = 30$ only, Fig. 5 presents the top-$M$ accuracy for $M = 1, \ldots, 120$. Fig. 5 also depicts the corresponding *achieved throughput ratio* $R_T = \sum_{i=1}^{N} \log_2(1 + y_{\widetilde{(p,q)}}) / \sum_{i=1}^{N} \log_2(1 + y_{\widehat{(p,q)}})$, where $N$ is the number of test examples and $\widetilde{(p,q)}$ is the best beam pair in $\mathcal{B}$. For $M = 10$, $R_T = 0.97$ and 0.69 for LOS and NLOS, respectively. In this case, while the overhead for beam-selection decreases by a factor of 24, the corresponding $R_T$ indicates a reduction to 69% of the achievable throughput for NLOS. For NLOS, $R_T$ reaches e. g. 94% for $M = 60$.

## V. Conclusions

LIDAR can be used for LOS detection and to reduce the mmWave beam-selection overhead in V2I scenarios. The results are promising in spite of the relatively simple adopted features. Future work includes exploring alternative features, using a larger amount of data, better tuning the many ML parameters for improved NLOS performance and fusing data from LIDAR and others sensors to maximize throughput with a reasonable overhead and computational cost.

## References

[1] N. González-Prelcic, A. Ali, V. Va, and R. W. Heath, "Millimeter-wave communication with out-of-band information," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 140–146, Dec. 2017.

[2] J. C. Aviles and A. Kouki, "Position-Aided mm-Wave Beam Training Under NLOS Conditions," *IEEE Access*, vol. 4, pp. 8703–8714, 2016.

[3] A. Loch, A. Asadi, G. H. Sim, J. Widmer, and M. Hollick, "Mm-Wave on wheels: Practical 60 GHz vehicular communication without beam training," in *9th COMSNETS*, Jan. 2017, pp. 1–8.

[4] V. Va, J. Choi, T. Shimizu, G. Bansal, and R. W. Heath, "Inverse Multipath Fingerprinting for Millimeter Wave V2I Beam Alignment," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4042–4058, May 2018.

[5] Y. Wang, A. Klautau, M. Ribero, M. Narasimha, and R. Heath, "Mmwave vehicular beam training with situational awareness by machine learning," in *Proc. of IEEE GLOBECOM*, Dec. 2018.

[6] J. Choi, V. Va, N. González-Prelcic, R. Daniels, C. R. Bhat, and R. W. Heath, "Millimeter-Wave Vehicular Communication to Support Massive Automotive Sensing," *IEEE Commun. Mag.*, vol. 54, no. 12, pp. 160–167, Dec. 2016.

[7] J. Kim and A. F. Molisch, "Fast millimeter-wave beam training with receive beamforming," *Journal of Comm. and Networks*, vol. 16, no. 5, pp. 512–522, Oct. 2014.

[8] P. Zhou, X. Fang, Y. Fang, Y. Long, R. He, and X. Han, "Enhanced Random Access and Beam Training for Millimeter Wave Wireless Local Networks With High User Density," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 7760–7773, Dec. 2017.

[9] T. Nitsche, A. B. Flores, E. W. Knightly, and J. Widmer, "Steering with eyes closed: Mm-Wave beam steering without in-band measurement," in *Proc. INFOCOM*, Apr. 2015, pp. 2416–2424.

[10] A. Ali, N. González-Prelcic, and R. W. Heath, "Millimeter Wave Beam-Selection Using Out-of-Band Spatial Information," *IEEE Trans. Wireless Commun.*, vol. 17, no. 2, pp. 1038–1052, 2017.

[11] N. González-Prelcic, R. Méndez-Rial, and R. W. Heath, "Radar aided beam alignment in MmWave V2I communications supporting antenna diversity," in *Proc. Inf. The. App. Workshop (ITA)*, Jan. 2016.

[12] W. B. Abbas and M. Zorzi, "Context information based initial cell search for millimeter wave 5G cellular networks," in *2016 European Conf. on Networks and Comm. (EuCNC)*, Jun. 2016, pp. 111–116.

[13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[14] A. Géron, *Hands-On Machine Learning with Scikit-Learn and TensorFlow*. O'Reilly Media, 2017.

[15] D. Cossock and T. Zhang, "Statistical analysis of Bayes optimal subset ranking," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 5140–5154, Nov. 2008.

[16] A. Klautau, P. Batista, N. González-Prelcic, Y. Wang, and R. W. Heath, "5G MIMO data for machine learning: Application to beam-selection using deep learning," in *Proc. Inf. Theory Appl. Workshop (ITA)*, 2018.

[17] L. Narula, J. M. Wooten, M. J. Murrian, D. M. LaChapelle, and T. E. Humphreys, "Accurate collaborative globally-referenced digital mapping with standard GNSS," *Sensors*, vol. 18, no. 8, 2018.

[18] https://github.com/lasseufpa/5gm-lidar.

[19] M. Gschwandtner, R. Kwitt, A. Uhl, and W. Pree, "BlenSor: Blender sensor simulation toolbox," in *Proc. 7th Int. Symp. Advances in Visual Computing*, 2011, pp. 199–208.

[20] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO - Simulation of Urban MObility," *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, pp. 128–138, Dec. 2012.