3D Scene Based Beam Selection for mmWave Communications

Weihua Xu, Feifei Gao, Shi Jin, and Ahmed Alkhateeb

Abstract—In this paper, we present a novel framework of 3D scene based beam selection for mmWave communications that relies only on the environmental data and deep learning techniques. Different from other out-of-band side-information aided communication strategies, the proposed one fully utilizes the environmental information, e.g., the shape, the position, and even the materials of the surrounding buildings/cars/trees obtained from 3D scene reconstruction and presents a full picture that could determine the wireless channels. Specifically, we build the neural networks with the input as point cloud of the 3D scene and the output as the beam indices. Compared with the LIDAR aided technique, the proposed 3D scene reconstruction is achieved from multiple images taken offline from cameras and thus significantly lowers down the cost and makes itself applicable for small mobile terminals. Simulation results show that the proposed 3D scene based beam selection can outperform the LIDAR method in beam selection accuracy.

Index Terms—3D Scene Reconstruction, Point Cloud, Deep Learning, Beam Selection, 3D Scene Based Wireless Communications,

I. INTRODUCTION

Millimeter wave (mmWave) communications is an essential part of 5G due to its large bandwidth of radio spectrum. Generally, massive number of antennas are adopted at the base station (BS) and the narrow beamwidth is formulated to overcome the high path attenuation of mmWave band and to achieve ultra high transmission rate. However, narrow beamwidth makes the beam alignment a difficult task, especially in high mobility scenarios like vehicle-to-everything (V2X) communications. In this case, the traditional beamforming strategies, such as beam sweeping or beam computing, that rely on channel estimation would be repeated frequently and thus bring significant time and energy overhead. Hence, research on highly efficient beamforming schemes to avoid latency caused by beam misalignment or beam switching has attracted increasing attention in recent years [1]-[2].

Using out-of-band side-information currently becomes one major approach to reduce channel estimation or beam selection overhead. In [3], the out-of-band spatial information obtained from sub-6 GHz is used for mmwave beam selection. In [4]-[6], the fingerprinting-based database that is composed of the side-information and the corresponding beam information are used for beam selection, where the side-information can be vehicle positions obtained by GPS. In [7], the positions of multiple vehicles are adopted through the machine learning tools under the V2I scenario. As vehicles can equip auxiliary sensors [8], such as radar, LIDAR, camera, etc., a more recent work utilizes the point cloud scanned by LIDAR to improve the efficiency of beam selection [9]-[10], which is shown to have higher accuracy than the position-aided method [10]. However, using the expensive high precision LIDAR may not be always available and affordable, especially for portable terminals like mobile phones. Hence, some other affordable ways to get point clouds should be explored.

In this paper, we present a new framework of how to utilize the surrounding 3D scene of the cellular coverage to help the wireless communications and to reduce the training overhead. Specifically, we proposed a novel 3D scene based beamforming selection method by utilizing panoramic scene feature. The 3D scene reconstruction techniques is applied from ordinary camera that takes images for cellular coverage area from many perspectives. Then the panoramic point cloud can be built and is stored at the base station (BS) or at the mobile station (MS). Such an approach does not have the huge cost as compared to the LIDAR-based method and is much cheaper for mobile users or BSs. Moreover, the panoramic point cloud can be regarded as a base scene feature of surrounding environment and would remain unchanged if the buildings of this environment are not modified. With the information of MS’s position, we design a relative scene feature from the saved point cloud to represent the relative spatial relationship between BS and MS. Based on point cloud segmentation, the size of designed feature is also fully reduced compared with the original saved point cloud data for lower computational overhead. Then the deep neural network (DNN) can be trained with such relative scene feature as well as the optimal beam indices from the training set. For users at the other positions, i.e., the test positions, the trained DNN could predict the optimal transmit and receive beams, simultaneously.

Based on the proposed DNN training strategy, our target is to train a general DNN, which can be applied to various 3D scenes without retraining. More specifically, DNN can learn the distribution of the building characteristics, e.g., the number, the position, siz, etc., and thereby keeps stable performance for different new scenes whose buildings follow the same distribution. The proposed 3D scene based approach does not require huge overhead of transmitting point clouds from MS to BS compared with the LIDAR centralized method [9]. While compared with the LIDAR distributed method [10], the proposed 3D scene based approach can adapt to new environments in a better way and achieve a higher beam.
prediction accuracy, as will be seen in the simulation part.

II. SYSTEM MODEL

A. Signal Model

Let us consider a downlink mmWave MIMO system with a single MS, while this 3D scene based framework can be straightforwardly adapted to multi-user scenario. BS is equipped with a uniform planar array (UPA) of $N_B = N_B^a \times N_B^b$ antennas, and MS is equipped with a UPA of $N_M = N_M^a \times N_M^b$ antennas. The antenna spacing of transmit and receive UPA is half carrier wavelength. We assume BS and MS both have a single radio frequency (RF) chain at mmWave band to save the cost. The downlink signal received at the user can be expressed as

$$y = f_M H f_B s + f_M n,$$  \hspace{1cm} (1)

where $H \in \mathbb{C}^{N_M \times N_B}$ is the channel matrix, $f_M \in \mathbb{C}^{N_M \times 1}$ is the receive beamforming vector, $f_B \in \mathbb{C}^{N_B \times 1}$ is the transmit beamforming vector, and $n \in \mathbb{C}^{N_B \times 1}$ is the Gaussian noise signal. We assume the transmit power is $P$, i.e., $E\{sn\} = P$.

The codebook-based beamforming strategy is generally adopted in practice, where the pair of beamforming vectors $(f_M, f_B)$ should be selected from the preset receive beam codebook $\mathcal{F}_B = \{f_{B,1}, f_{B,2}, \cdots, f_{B,n}\}$ and the transmit beam codebook $\mathcal{F}_M = \{f_{M,1}, f_{M,2}, \cdots, f_{M,m}\}$. The target is to maximize the receive SNR and select the optimal beamforming vector pair from

$$(f_M^{opt}, f_B^{opt}) = \arg \max_{f_M \in \mathcal{F}_M, f_B \in \mathcal{F}_B} |f_M^H H f_B|^2 / \|f_M\|^2.$$  \hspace{1cm} (2)

B. Channel Model

For mmWave transmission, the geometric channel model is usually adopted and the channel matrix $H$ is expressed as

$$H = \sum_{l=1}^{N_L} \alpha_l a_r(\theta_l^e, \phi_l^e) a_l^H(\theta_l^t, \phi_l^t),$$  \hspace{1cm} (3)

where $\alpha_l$ is the complex gain of the $l$th path, $\theta_l^e$ and $\phi_l^e$ are the elevation and azimuth of the $l$th path’s angle of arrive (AOA), $\theta_l^t$ and $\phi_l^t$ are the elevation and azimuth of the $l$th path’s angle of departure (AOD), $a_r(\theta, \phi) \in \mathbb{C}^{N_M \times 1}$ and $a_l(\theta, \phi) \in \mathbb{C}^{N_B \times 1}$ are the complex steering vectors of receive and transmit antenna. In addition, the steering vector of a UPA with $N = N_a \times N_b$ antennas is given by

$$a(N_a, N_b, \theta, \phi) = \frac{1}{\sqrt{N_a N_b}} a_{az} \otimes a_{ele},$$  \hspace{1cm} (4)

where

$$a_{ele} = [e^{j\pi \cos(\theta)}, \cdots, e^{j(N_a-1)\pi \cos(\theta)}]^T,$$

$$a_{az} = [e^{j\pi \sin(\phi) \cos(\theta)}, \cdots, e^{j(N_b-1)\pi \sin(\phi) \sin(\theta)}]^T,$$

and $\otimes$ represents the kronecker product. Then, we have $a_r(\theta, \phi) = a(N_a^b, N_M^a, \theta, \phi)$ and $a_l(\theta, \phi) = a(N_B^a, N_M^b, \theta, \phi)$.

III. THE 3D SCENE BASED BEAMFORMING

To maximize $\mathbb{E}\{y^H H y\}$, the conventional communications approach needs first obtain $H$, which then needs BS to send large amount the training signals. Such training overhead costs much spectral resource, especially when BS has multiple antennas, and would be repeated frequently when the user is moving. Albeit everyone knows that the reasons of channel changing is due to the change of the scene between BS and MS, i.e., buildings, cars, trees, people, etc, there is no explicit function to describe such relationship. Nevertheless, with the recently developed deep learning, it would be very much possible to build the relationship from the 3D scene to the channel with the help of a properly trained DNN.

In this work, we propose to reconstruct the 3D scene with the images obtained from the camera at BS and MS or from any other means, and then relate it with parameters in wireless communications, e.g., beam index. Interestingly, 3D scene reconstruction is a core problem in the field of computer vision and has been involved in many applications, such as autonomous driving, augmented reality, etc. We here explore the roles of 3D reconstruction techniques for wireless communication, since it represents the full spatial feature between BS and MS, and we expect 3D scene based approach to outperform the conventional candidates that only leverage partial information from sensors.

Fig. 1 illustrates the proposed framework with an example of BS storing the point cloud. We first take enough photos offline for the cellular coverage, with lens facing towards BS. Next we apply the 3D reconstruction methods to rebuild the 3D scene of the whole cell and then convert it to the panoramic point cloud. During the training stage, the user could report its position, and then BS can construct a relative panoramic scene feature by combining user’s relative position and the panoramic point cloud. Such relative panoramic feature can be used as the input to train DNN and is used for beam selection later during the testing stage. In cases that the user’s position is private, the full panoramic point cloud of the cell can be downloaded at MS in advance, and user could train the DNN itself and then feedback BS the indices of the selected beams during the testing stage.

An important novelty of the proposed framework is that, we may train the same neural network under many different environments, i.e., different surrounding buildings distributions, such that trained DNN would be adaptable to the new environments with similar types of buildings. In other words, if two neighboring cells have similar buildings, then the DNN trained from one cell would also be used for beam selection in another cell. The above framework is illustrated in details as follows:

A. Panoramic Point Cloud Generation

With no practical data in hand, we resort to the computer generated data to verify the proposed framework. Specifically, we use Belender\footnote{The photos can be obtained from MSs, UAV, or any other approaches.} a the 3D modeling software, to create several building models and combine them randomly into
different environments. An example is shown in Fig. 2 (a), where we choose simple brick texture to overspread the surface of all the buildings to ease 3D reconstruction.

To reconstruct the point cloud as accurate as possible, we require the camera to revolve around the environment at different heights, and keep the camera lens facing the geometric center of the environment all the time. Hence, the angle interval and the height interval of revolving become the critical parameters to determine the quality of reconstruction, i.e., they would affect the overlapping degree of adjacent images. It needs to be mentioned that, even if we require for a dense angle and height interval to get high-quality 3D reconstruction, the cost of taking photos offline by ordinary camera are still not very large, especially when compared with the LiDAR data.

After taking photos at each specific angle and height, we obtain an image set of the corresponding environment. The reconstruction is achieved by inputting the obtained image set to COLMAP [12], which is an open-source 3D reconstruction tools based on Structure-from-Motion [13] and Multi-View Stereo [14] algorithms. It is worth noting that the order of input images is irrelevant. Fig. 2(b) shows the reconstructed panoramic point cloud of the environment example in Fig. 2(a). Moreover, it is also possible to recognize the materials of the surrounding buildings, etc., bricks, woods, plastics, glass, metals, etc., from photos [15] and then add these materials information as the input to DNN. By doing so, the DNN can better represent the relationship between the 3D scene and the transceiver beams.

**B. Relative Panoramic Scene Feature Extraction**

A drawback of the 3D reconstruction from images is that the coordinates of all points in the reproduced point cloud would be the rotated, translated and scaled version of the absolute coordinates of the points in Blender. We then calculate such transformation by finding some flag matching points. More specifically, we use the vertexes of buildings that are simple to be identified both in the point cloud and actual environment. Afterwards, the absolute coordinates can be obtained by the inverse transformation and is expressed as a set $P = \{p_1, p_2, \cdots, p_K\}$, where $K$ denotes the number of all points in the point cloud and each element $p_k \in \mathbb{R}^{3 \times 1}$ represents the three-dimensional coordinates of one point. We assume the BS already knows its absolute coordinate $p_B \in \mathbb{R}^{3 \times 1}$ (also known to MS). Hence, when the MS gets its own absolute coordinates $p_M \in \mathbb{R}^{3 \times 1}$ and feedbacks to BS, BS can construct a relative panoramic scene feature contains both MS location and surrounding environment using $P$, $p_B$ and $p_M$ (note that this can also be done at MS if the privacy of user’s position should be maintained).

The core idea of constructing the relative features is to create a relative coordinate system with origin $p_M$ and making this coordinate system face towards $p_B$, i.e., making its one coordinate axis pass through $p_B$, as Fig. 3 shows. Thus, we choose the line pass $p_M$ and $p_B$ as the X-axis, without loss.
have played important roles in wireless communication. Fig. 4 shows the proposed deep learning network with the fully-connected layers, where the inputs are the relative panoramic scene features and the outputs are the optimal beam indices.

We train two DNNs to predict the optimal receive and transmit beam respectively (they can also be trained jointly), where the two DNN have the same structure as well as the same inputs as relative point cloud. Our target is to achieve a general DNN that can adapt to different environments, and the detailed training and testing phases are explained as follows:

Training: We generate numerous different environments in a random way and select the positions of MS for each generated environment. For each environment and each MS position, the corresponding channel is produced by the ray tracing software Wireless Insite [16]. Thus, the relative panoramic scene feature and the corresponding optimal transmit/receive beam indices can be acquired at each position of MS to build a training sample. The total training samples built in each generated environment formulate the training set. Then we train the transmit and receive DNN based on their respective training set by utilizing the RMSProp optimizer and the cross entropy loss function.

Testing: We generate new environments in the same random way as the training phase, but ensure these environments are different from previous environments used for training. Then we apply the same way to build the testing samples in all new environments and formulate the testing set to validate the networks’ generalization performance.

IV. SIMULATION RESULTS

A. Simulation Setup

We design three types of cube-like building models and summarise their sizes in TABLE. I. The random way of generating environments for simulation is explained in Fig. 5 that is the top view of BS coverage area. We focus on the of generality. The relative coordinates of all points in point cloud inside this relative coordinate system can be expressed as a new set \( \tilde{P} = \{ \tilde{p}_1, \tilde{p}_2, \ldots, \tilde{p}_K \} \) with

\[
\tilde{p}_k = G_{1,3}(\frac{\pi}{2} - \hat{\theta})G_{1,2}(-\hat{\phi})(p_k - p_M) \tag{7}
\]

\[
\hat{\theta} = \text{Arg}(\tilde{z} + j\sqrt{\tilde{x}^2 + \tilde{y}^2}) \quad \hat{\phi} = \text{Arg}(\tilde{x} + j\tilde{y}) \tag{8}
\]

where \( 1 \leq k \leq K \), \((\tilde{x}, \tilde{y}, \tilde{z}) = p_B - p_M\), \(G_{1,2}(\theta)\) and \(G_{1,3}(\theta)\) are the third-order Givens rotation matrices representing the rotation in X-Y plane and Z-X plane, respectively.

It can be easily known that \( \tilde{P} \) is related to \( P \) and \( p_M \), thereby serves as usable relative feature. In addition, the original point cloud is generally very dense, and hence many neighboring points in \( P \) are not necessary and can be regarded as redundant data. Moreover, large dimensions of features would also increase the scale of neural networks.

Therefore, we design the following two steps to reduce the data redundancy: (1) We first adopt random extraction with equal possibility to select \( K \ll K \) points from the original point cloud; (2) We use a cube area \( C \) to cover \( K \) points, as shown in Fig. 3. As BS has limited coverage, we can find a suitable specific cube area \( C \) that moves with MS, remains fixed under the MS’s relative coordinate system, and keeps on containing all or most of the extracted \( K \) points. More concretely, \( C \) would represent a set of all points’ relative coordinates within the suitable cube and should be fixed. Define \( P \) as the set of the extracted \( K \) points’ relative coordinates, also a function of \( p_M \), and define

\[
\alpha_C = \frac{\text{Card}(C \cap \tilde{P})}{K}, \tag{9}
\]

where Card represents cardinality. Then, \( C \) should satisfy \( \alpha_C = 1 \) for any possible MS position \( p_M \).

Since missing few points would bring little impact, it would be possible to further reduce the volume of \( C \) compared with the target that \( C \) must contain all \( K \) points. Therefore a reasonably sized \( C \) should be determined by reducing its volume as much as possible while containing all extracted points.

After obtaining a reasonably designed cube area \( C \), we next evenly divide the length, the width and the height of \( C \) into \( a, b \) and \( c \) parts respectively. Then, \( C \) is further divided into \( a* b* c \) small cubic blocks, and we can partition the \( K \) extracted points into \( a* b* c \) spatial blocks to construct the relative panoramic scene feature, named as \( g \in \mathbb{R}^{abc*3} \), where the \( i \)th row of \( g \) is the mean value of coordinates of those points only appearing in the \( i \)th spatial block. We can set one row of \( g \) be 0 if there are no points in the corresponding spatial block. Compared with the original feature \( P \), the dimension of \( g \) is significantly reduced while such relative panoramic scene feature still works efficiently, as will be seen in the later simulations.

C. Deep Learning Model for Beam Selection

Deep learning has been applied in various fields, such as image processing, natural language processing, etc., and also have played important roles in wireless communication.
The main parameters of Wireless Insite to generate channels are described in TABLE. II. We choose the number of the extracted points as \( \hat{N} = 10 \). The antenna numbers of arrays in BS and MS are \( N^a_B, N^a_M = 8 \) and \( N^b_B, N^b_M = 72 \), respectively. We adopt the beam codebook with equal azimuth angle interval. We let BS and MS both have the same codebook with \( N = 30 \) and \( N = 50 \) beams i.e., \( f_{B,i} = f_{M,i} = a(N^a_B, N^a_M, \alpha_i, 2f_i - 2N_i) \), \( i = 1, 2, \cdots, N \), where \( \alpha \) is fixed to 95° according to the horizontal line constraint for MS position and the BS’s/MS’s heights.

We set \( T = 6 \) and \( T' = 4 \) for our deep learning network. For the transform network, the node number of each layer is set to be 3, 20, 10, 1 respectively in index order. For the fully-connected network after the concatenate layer, the node number of each layer is set to be 2700, 1200, 600, 300, 200, 100, \( N \) respectively in index order.

### B. Results and Discussions

As shown in Fig. 6, we analyze the top-5 selection accuracy of the proposed method under different sizes of the training set. With more training samples as input, the selection accuracy of proposed method increases. When more than 8000 samples are used as input, the increasing of the selection accuracy starts to slow down, i.e., few improvements can be achieved by inputting more samples. In this case, one may try to find other ways to further increase the accuracy, such as modeling based approach or a few pilot signals. Hence, there is a trade-off between the selection accuracy and the effort paid for training.

It is also seen that the receive beam selection accuracy is higher than the transmit beam selection accuracy. This may be because that the MS coordinate based input feature of DNN is more favor for MS’s beam selection. In addition, the accuracy decreases with the increase of \( N \), as expected.

We next compared the proposed method with the LIDAR distributed method, which performs the scanning point cloud and predicts the optimal beam pair in the MS. The observational distance of LIDAR is set as 60m. Fig. 7 shows that the selection accuracy of the proposed method and LIDAR distributed method increases with the values of \( M \). It is seen that the proposed method can outperform the LIDAR distributed method by 20% on average and up to nearly 30% in top-\( M \) selection accuracy. This result indicates the panoramic

---

**TABLE II**

| Parameter                  | Value   |
|---------------------------|---------|
| Carrier Frequency         | 60 GHz  |
| Propagation Model         | X3D     |
| Building Material         | Concrete|
| Diffuse Scattering Reflections | 2      |
| Diffuse Scattering Diffractions | 1      |
| Maximum Paths Per Receiver Point | 25     |
| Transmitter Power         | 0 dBm   |

Fig. 5. The top view of BS coverage area for simulate generating environments

Fig. 6. Top-5 beam selection accuracy of the proposed method with different input training set sizes
point clouds constructed by the proposed method are more appropriate for accurate beam selection when compared with the local point cloud scanned by single MS’s LIDAR. The results can be explained by the richer spatial information of our panoramic point cloud than the part spatial information of single LIDAR scanned point cloud. Another reason is that the proposed method have a better generalization performance for new scenes, whereas the LIDAR distributed method only works well for single environment [10]. It needs to emphasized that the cost of the proposed 3D scene reconstruction method is much lower than LIDAR distributed method in terms of the point cloud acquisition and user corporation.

V. CONCLUSIONS

We have proposed a brand new beam selection method through image-based 3D reconstruction in order to avoid the severe overhead of using expensive auxiliary devices, such as Radar and LIDAR, etc. With proper training, the proposed approach can predict the optimal beam for any points in the current cell and it can even work for a new environment with the same type of buildings distributions. Simulations show that the proposed approach has higher beam selection accuracy and lower overhead compared to the LIDAR distributed method, which indicates the advantage of exploring image processing techniques. It is also one of the first attempts to relate the 3D scene and the wireless communications. More applications benefited from 3D reconstruction or other image techniques within wireless communications are of interest and deserve further exploration.

REFERENCES

[1] V. D. P. Souto, R. D. Souza, B. F. Uchó-Filho, and Y. Li, “A novel efficient initial access method for 5G millimeter wave communications using genetic algorithm,” IEEE Trans. Veh. Technol., vol. 68, no. 10, pp. 9908–9919, Oct. 2019.
[2] M. Giordani, M. Mezzavilla, C. N. Barati, S. Rangan, and M. Zorzi, “Comparative analysis of initial access techniques in 5G mmWave cellular networks,” in Proc. Annu. Conf. Inf. Sci. Syst. (CISS), Mar. 2016, pp. 268–273.
[3] A. Ali, N. Gonzalez-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, Feb. 2018.
[4] J. C. Aviles, and A. Kouki, “Position-aided mm-wave beam training under NLOS conditions,” IEEE Access, vol. 4, pp. 8703–8714, 2016.
[5] Z. Wei, Y. Zhao, X. Liu, and Z. Feng, “DoA-LF: A location fingerprint positioning algorithm with millimeter-wave,” IEEE Access, vol. 5, pp. 22678–22688, 2017.
[6] K. Satyanarayana, M. El-Hajjar, A. A. M. Mourad, and L. Hanzo, “Deep learning aided fingerprint-based beam alignment for mmWave vehicular communication,” IEEE Trans. Veh. Technol., vol. 68, no. 11, pp. 10858–10871, Nov. 2019.
[7] Y. Wang, A. Klautau, M. Ribero, A. C. K. Soong, and R. W. Heath, “MmWave vehicular beam selection with situational awareness using machine learning,” IEEE Access, vol. 7, pp. 87479–87493, 2019.
[8] L. Liang, H. Peng, G. Y. Li, and X. Shen, “Vehicular communications: a physical layer perspective,” IEEE Trans. Veh. Technol., vol. 66, no. 12, pp. 10647–10659, Dec. 2017.
[9] A. Klautau, N. Gonzalez-Prelcic, and R. W. Heath, “LIDAR data for deep learning-based mmWave beam-selection,” IEEE Wireless Commun. Lett., vol. 8, no. 3, pp. 909–912, June 2019.
[10] M. Dias, A. Klautau, N. Gonzalez-Prelcic, and R. W. Heath, “Position and LIDAR-aided mmWave beam selection using deep learning,” in Proc. IEEE Int. Workshop on Signal Processing Adv. in Wireless Commun. (SPAWC), Cannes, France, 2019, pp. 1–5.
[11] https://www.blender.org
[12] https://colmap.github.io
[13] J. L. Schönberger and J. Frahm, “Structure-from-motion revisited,” in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 4104–4113.
[14] J. L. Schönberger, E. Zheng, J. Frahm, and M. Pollefeys, “Pixelwise view selection for unstructured multi-view stereo,” in Proc. Euro. Conf. on Computer Vision (ECCV), Amsterdam, OC, 2016, pp. 501–518.
[15] M. Varma and A. Zisserman, “A statistical approach to material classification using image patch exemplars,” IEEE Trans. Pattern Anal. Machine Intell., vol. 31, no. 11, pp. 2032–2047, Nov. 2009.
[16] https://www.remcom.com/wireless-insite-em-propagation-software