IEEE ComSoc IEEE Communications Society
IEEE Open Journal of the
Communications Society

Received 25 September 2024; revised 3 November 2024; accepted 20 November 2024. Date of publication 3 December 2024;
date of current version 13 December 2024.

Digital Object Identifier 10.1109/OJCOMS.2024.3509453

Environment Semantic Communication: Enabling Distributed Sensing Aided Networks

SHOAIB IMRAN, GOURANGA CHARAN, AND AHMED ALKHATEEB

School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85287, USA

CORRESPONDING AUTHOR: A. ALKHATEEB (e-mail: alkhateeb@asu.edu)

This work was supported in part by the National Science Foundation under Grant 2048021.

ABSTRACT Millimeter-wave (mmWave) and terahertz (THz) communication systems require large antenna arrays and use narrow directive beams to ensure sufficient receive signal power. However, selecting the optimal beams for these large antenna arrays incurs a significant beam training overhead, making it challenging to support applications involving high mobility. In recent years, machine learning (ML) solutions have shown promising results in reducing the beam training overhead by utilizing various sensing modalities such as GPS position and RGB images. However, the existing approaches are mainly limited to scenarios with only a single object of interest present in the wireless environment and focus only on co-located sensing, where all the sensors are installed at the communication terminal. This brings key challenges such as the limited sensing coverage compared to the coverage of the communication system and the difficulty in handling non-line-of-sight scenarios. To overcome these limitations, our paper proposes the deployment of multiple distributed sensing nodes, each equipped with an RGB camera. These nodes focus on extracting environmental semantics from the captured RGB images. The semantic data, rather than the raw images, are then transmitted to the basestation. This strategy significantly alleviates the overhead associated with the data storage and transmission of the raw images. Furthermore, semantic communication enhances the system's adaptability and responsiveness to dynamic environments, allowing for prioritization and transmission of contextually relevant information. Experimental results on the DeepSense 6G dataset demonstrate the effectiveness of the proposed solution in reducing the sensing data transmission overhead while accurately predicting the optimal beams in realistic communication environments.

INDEX TERMS Beamforming, camera, computer vision, deep learning, distributed sensing, environment semantics, millimeter-wave, semantic communications.

I. INTRODUCTION

UTILIZING higher frequency bands, such as mmWave in 5G and possibly sub-terahertz in 6G, is a key trend in current and future communication systems. These frequency ranges provide higher bandwidths, enabling the communication systems to efficiently meet the higher data rate demands of emerging applications such as augmented/virtual reality, autonomous vehicles, and smart cities [1], [2], [3]. However, these systems necessitate the deployment of large antenna arrays and the use of narrow beams at both the transmitter and receiver to ensure adequate receive signal power. Selecting the best beams for these large antenna arrays incurs a substantial training overhead, making it challenging to satisfy the low-latency and high-reliability requirements of these current and future applications. This emphasizes the need to explore innovative approaches that (i) reduce or mitigate the training overhead associated with beam selection and (ii) enable highly mobile wireless communication applications.

Several solutions have been proposed over the years to reduce the beam training and channel estimation overhead in mmWave communication systems [4], [5], [6], [7]. The focus of these solutions has been mainly on: (i) The development of beam training with adaptive/hierarchical beam codebooks [4], [5]. (ii) The utilization of compressive sensing tools [5] to estimate the full channel with a much smaller number of measurements. This is motivated by the sparsity nature of the mmWave channels, where only a few dominant paths typically exist between the transmitter and receiver. (iii) The design of beam tracking techniques [6] that leverages the user mobility information to predict the future beams and hence reduce the exhaustive search beam training overhead. These classical approaches, however, usually result in a training overhead reduction of only one order of magnitude, which is not sufficient for very large antenna array systems and applications that require very low-latency.

The challenges faced by classical solutions have led to the development of machine learning approaches that leverage prior observation and additional sensing information [8], [9], [10], [11], [12], [13], [14]. The additional sensing modalities include position (GPS location) [9], RGB images [10], [11], LiDAR [12], and radar [13], [14]. The additional sensing information provides a crucial environmental context, enabling an in-depth comprehension of the wireless environment and its influence on channel characteristics. These prior studies have demonstrated the potential of utilizing additional side information in minimizing the beam training overhead. However, these solutions have certain limitations. Firstly, they are primarily designed for scenarios with a single object of interest, which can be challenging when scaling them to real-world situations with multiple objects. Secondly, the additional sensors used in these solutions, such as cameras, LiDAR, and radar, are positioned exclusively at the basestation and have a limited range of approximately $60-80$ meters [15]. This range is significantly shorter than the typical range of the mmWave communication systems, which is around 300 meters. Consequently, this limited range of these additional sensing modalities significantly impacts the effectiveness of these solutions in real-world wireless communication tasks (such as beam prediction and proactive blockage prediction). Additionally, these sensors do not provide coverage for non-line-of-sight scenarios, further restricting their applicability in diverse environments.

One promising solution to overcome these challenges is deploying multiple nodes, each equipped with its own sensors, to capture information about the wireless environment in a coordinated manner. This distributed sensing approach enhances coverage, reliability, and adaptability by strategically distributing sensors throughout the network [16], [17]. Instead of relying solely on sensors at the basestation, data collected by these distributed nodes can be utilized by one or more basestations to make informed decisions. This scalable and robust approach leverages the collective sensing capabilities of multiple nodes, providing a comprehensive view of the wireless environment and optimizing tasks such as beam prediction and proactive blockage prediction. However, as the number of distributed nodes increases, challenges arise in managing the growing volume of captured data, including storage, processing, and transmission concerns. Furthermore, the heightened data rate resulting from the increased number of nodes necessitates robust data synchronization methods to maintain temporal coherence.

One approach to address these challenges is to process the data captured by the distributed nodes locally, either at the edge or in the cloud. This processing involves extracting essential information, referred to as semantics, from raw sensor data and transmit these semantics to basestation. Semantic communication systems are broadly classified into two categories: source-oriented semantic communications (SOSCs) and channel-oriented semantic communications (COSCs) [18]. An SOSC system typically comprises a semantic encoder at the transmitter and a semantic decoder at the receiver. The semantic encoder, implemented as a deep neural network, extracts semantic features based on the transmitter's background knowledge. The semantic decoder, also a deep neural network, interprets these semantic features and reconstructs the original source signals using the receiver's background knowledge. In SOSCs, the transmitted semantic information consists of feature vectors produced by the encoder's deep neural network. For instance, [19] and [20] utilize deep neural networks to extract feature vectors from text and speech data, respectively, and transmit these feature vectors as the semantic information of the source data. Conversely, COSC systems extract channel semantics from sensory information about the environment data such as images, LiDAR, or radar [18]. This semantic information is utilized for channel-related downstream tasks, including blockage prediction, beam prediction, and basestation handoff. In [21], the semantic information in channel-oriented semantic communication system is divided into two categories: (i) parameter semantics and (ii) environment semantics. Parameter semantics include channel parameters such as angle of arrival, angle of departure, number of paths, and Doppler frequency offset, which can be obtained from sensors like radar and GPS. Environment semantics, on the other hand, refer to information about the environment, such as the layout, shape, number, and category of objects present in it [22]. The most effective form of semantic information for channel-oriented semantic communication system would depend on the sensor data modality and the communication task to be performed at the basestation. In our work, which involves using cameras at distributed nodes for vehicle-to-infrastructure communication, we focus on extracting environment semantics. Moreover, to extract environment semantics from images, we can utilize powerful yet efficient pre-trained deep learning models such as YOLOv7 [23], eliminating the need to train models from scratch, as would be required for parameter semantics.

While exploring distributed learning solutions, it is notable that paradigms such as federated learning [24], [25] are gaining traction. This method entails training models across decentralized devices using local data, with the aggregated insights refining the overall model while maintaining data privacy and minimizing bandwidth use. In addition, distributed artificial intelligence (AI) and edge computing have shown potential in enhancing network functionalities. These technologies, particularly edge computing, have been effective in managing the data from distributed nodes, addressing significant challenges in data processing and storage [26], [27]. However, these advancements still need

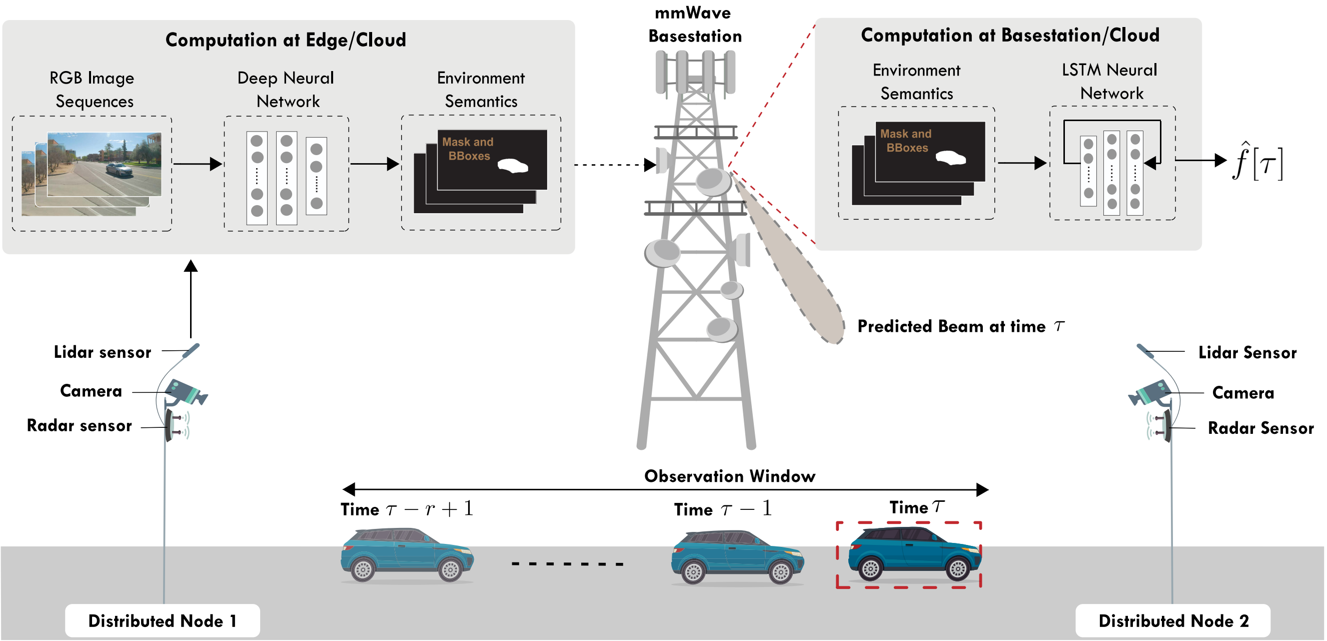


FIGURE 1. Overall system model of the proposed setup. The distributed nodes extract environment semantic information from the RGB images, which is subsequently transmitted to the basestation. This semantic information is then utilized for beam prediction at the basestation.

to be explored in the context of distributed sensing-aided vehicle-to-infrastructure (V2I) beam prediction and tracking. To bridge this gap, we aim to investigate the utilization of environment semantics in enabling distributed sensing-aided wireless communication in real-world scenarios. Specifically, we propose a novel approach that leverages environment semantics in a distributed sensing scenario to predict optimal beams in a real-world wireless communication setting accurately. The main contributions of this paper can be summarized as follows:

- Formulating the sensing-aided beam prediction problem for vehicle-to-infrastructure (V2I) communication scenario with multiple distributed nodes, each equipped with an RGB camera to capture the wireless environment.
- Developing a novel deep learning-based solution that leverages images captured by cameras installed at distributed nodes to accurately predict the optimal beam index at the basestation in a V2I communication scenario.
- Investigating various environment semantics that can be extracted from images, such as object bounding boxes and masks, with the aim of enabling distributed sensing-aided wireless communication. We further perform a comprehensive comparative study, evaluating the performance, complexity, and practical feasibility of these environment semantics for the specific task of distributed sensing-aided beam prediction.
- Providing the first real-world evaluation of distributed environment semantic-aided beam prediction based on a new scenario in the DeepSense 6G dataset [28]. This scenario focuses explicitly on the distributed aspect, capturing co-existing multi-modal data from the basestation and two distributed units, enabling the study of distributed sensing-aided wireless communication.

The paper is organized as follows: Section II provides the system model and problem formulation for the proposed solution. In Sections III and IV, we delve into the key idea and the proposed solution, respectively. The testbed and the DeepSense 6G dataset utilized in our experiments are described in Section V. Finally, in Section VI we present a detailed evaluation of the proposed solution.

II. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we present the adopted system model and formulate the distributed environment semantic-aided beam prediction problem.

A. SYSTEM MODEL

Fig. 1 illustrates the proposed distributed sensing-aided communication setup. N distributed nodes sense the environment. Each distributed node is equipped with an RGB camera. Furthermore, the basestation is also equipped with an RGB camera and 3 M-element uniform linear arrays (ULAs), with each ULA having a field of view of around 90°. The three ULAs are positioned 90° apart from each other and oriented towards the front, left, and right of the basestation. Furthermore, the area served by the basestation is divided into $N + 1$ subregions. The basestation camera provides sensing information for the region directly in front of the basestation while each distributed node provides sensing information for one of the remaining N regions. We strategically position the distributed nodes to enhance the combined camera coverage over the range of the mmWave communication system.

The user is equipped with a single-antenna transmitter. The basestation, for each ULA, uses (i) OFDM transmission with K subcarriers and a cyclic prefix of length D, and (ii) a pre-defined beam steering codebook $\boldsymbol{\mathcal{F}} = \{\mathbf{f}_q\}_{q=1}^{Q}$, where $\mathbf{f}_q \in \mathbb{C}^{M\times 1}$ is the q^{th} beamforming vector and Q is the total number of beamforming vectors. The beam steering beams are uniformly spaced and jointly cover the ULA's 90° field of view. In the downlink, the received signal at the user from the ULA that has the user in its field of view at the k^{th} sub-carrier and time t can be represented as

$$y_k[t] = \mathbf{h}_k^T[t]\mathbf{f}_q[t]x + v_k[t], \tag{1}$$

where $\mathbf{h}_k[t] \in \mathbb{C}^{M\times 1}$ denotes the channel between the basestation and the mobile user, $\mathbf{f}_q \in \boldsymbol{\mathcal{F}}$ is the beamforming vector, and $v_k[t]$ represents noise sampled from a complex Gaussian distribution $\mathcal{N}_\mathbb{C}(0, \sigma^2)$. The transmitted complex symbol $x \in \mathbb{C}$ satisfies the power constraint $\mathbb{E}[|x|^2] = P$, where P is the average symbol power. Moreover, the beamforming vector $\mathbf{f}_q[t]$, at each time step t is selected from the beam steering codebook $\boldsymbol{\mathcal{F}}$ to maximize the average receive SNR as follows

$$\underset{\mathbf{f}_q[t]\in\boldsymbol{\mathcal{F}}}{\operatorname{argmax}} \frac{1}{K}\sum_{k=1}^{K} \mathsf{SNR}\left|\mathbf{h}_k^T[t]\mathbf{f}_q[t]\right|^2, \tag{2}$$

where SNR is the transmit signal-to-noise ratio, $\mathsf{SNR} = \frac{P}{\sigma^2}$. At any time instant t, the receive power vector of effective channel gain with codebook elements from the ULA that has the user in its field of view can therefore be expressed as $\mathbf{p}[t] = [p_1[t], \ldots, p_Q[t]]$, where $\mathbf{p}[t] \in \mathbb{R}^{Q\times 1}$ and $p_q[t]$ is defined as

$$p_q[t] = \left|\mathbf{h}_k^T[t]\mathbf{f}_q[t]\right|^2. \quad q \in 1, \ldots.., Q \tag{3}$$

In the next subsection, we formulate the distributed environment semantic-aided beam prediction problem.

B. PROBLEM FORMULATION

Given the system model presented above, the goal is to select the optimal beam index (at the basestation for any given time t) that maximizes the receive power using camera images captured by the distributed node. Attaining this goal involves a few key tasks. Firstly, we need to determine the ULA that encompasses the user within its field of view. Secondly, we have to identify the sub-region where the user is located. Lastly, we must also discern the transmitter vehicle from other vehicles present in the RGB images. One potential solution to this problem is to leverage the receive power vector derived from previous time instances. Consequently, this work aims to develop a beam prediction model that utilizes a sequence of available RGB images and the ground truth receive power vector corresponding to the time instant of the first image capture. The receive power vector corresponding to the first image capture in the sequence serves a dual role in our proposed solution. Initially, it is used to identify the ULA and the sub-region where the user is located. Subsequently, this receive power vector is employed to facilitate the identification of the transmitter in the scene. Let $\mathbf{X}_n[t] \in \mathbb{R}^{W\times H\times C}$ represent the RGB image captured at time t by the camera installed at the n^{th} node, where W, H, and C are the width, height, and the number of color channels for the image, respectively. Further, let $\mathbf{p}[t] \in \mathbb{R}^{1\times Q}$ denote the mmWave receive power vector from the ULA that has the user in its field of view at time t. At any given time instant t, the distributed node n, captures a sequence of r RGB images, and the basestation collects the mmWave receive power vector corresponding to the time instant of the first image capture, $\mathbf{S}[t]$, defined as

$$\mathbf{S}[t] = \left\{\{\mathbf{X}_n[t]\}_{t=\tau-r+1}^{t=\tau}, \mathbf{p}[\tau - r + 1]\right\}, \tag{4}$$

where $r \in \mathbb{Z}$ is the length of the input sequence or the observation window to predict the optimal beam index. In particular, at any given time instant t, the goal in this work is to find a mapping function f_Θ that utilizes the available sensory data samples $\mathbf{S}[t]$ to predict (estimate) the optimal beam index $\hat{\mathbf{f}}[t] \in \boldsymbol{\mathcal{F}}$ with high fidelity. The mapping function can be formally expressed as

$$f_\Theta : \mathbf{S}[t] \to \hat{\mathbf{f}}[t]. \tag{5}$$

Let $\mathcal{D} = \{(\mathbf{S}_l, \mathbf{f}_l^\star)\}_{l=1}^{l=\varkappa_1}$ represent the available dataset collected from the real-world wireless environment. The total number of samples in the dataset is denoted by $\varkappa_1$. The goal is to maximize the number of correct predictions over all the samples in the dataset $\mathcal{D}$. This can be formally written as

$$f_{\Theta^\star}^\star = \underset{\Theta}{\operatorname{argmax}} \prod_{l=1}^{\varkappa_1} \mathbb{P}\big(f_\Theta(\mathbf{S}_l) = \mathbf{f}_l^\star\big), \tag{6}$$

where the joint probability distribution in (6) is due to the implicit assumption that the samples in $\mathcal{D}$ are drawn from an independent and identical distribution. The objective is to find the optimal set of parameters $\Theta^\star$ that maximizes the product of the probabilities of correct predictions. The set of parameters $\Theta^\star$ can be learned from the dataset $\mathcal{D}$ by utilizing a machine learning solution. During training, f_Θ can be considered random as the parameters change during training based on the stochastic updates. The model f_Θ is deterministic once the parameters Θ are learned. The next section presents the proposed deep learning-based solution for the proposed distributed sensing-aided beam prediction.

III. KEY IDEA

In this section, we present the key idea behind setting up distributed nodes and utilizing the environment semantics from these distributed nodes for beam prediction at the basestation.

Recent works [9], [10], [11], [12], [13], [14] demonstrate the potential of using various sensing modalities including position [9], LiDAR [12], radar [13], and RGB images [10] for beam prediction. These works primarily focus on co-located sensing and communication, where sensors are installed at the basestation. This approach introduces various challenges that need to be addressed. First, practical sensors have limited range capabilities. When machine learning models process this sensor data to detect distant objects, the resulting bounding boxes and masks appear disproportionately small, giving the impression of minimal movement even if the objects are actually moving rapidly. Second, the sensors predominantly rely on line-of-sight (LOS) conditions for accurate data capture [29]. Consequently, non-line-of-sight (NLOS) scenarios, which are inherently challenging, significantly impact the usability of these sensors. Achieving reliable mmWave communication necessitates the development of techniques that can effectively handle both LOS and NLOS cases.

To unlock the full potential of additional sensing modalities, **adopting a distributed sensing approach is essential**. This approach entails deploying multiple distributed nodes, each equipped with sensors such as camera, LiDAR, and radar, to overcome limitations in range and expand the scope of data collection to cover a broader area. Furthermore, distributed sensing enables us to address NLOS scenarios effectively by enhancing sensing coverage and capturing diverse perspectives. By capitalizing on the synergistic capabilities of multiple sensors deployed across the network, we can elevate overall system performance and realize advanced functionalities.

As the number of distributed nodes increases, there is a **corresponding increase in the volume of captured data**, which brings forth several challenges. Firstly, the substantial size of the accumulated data presents significant obstacles in terms of storage, processing, and transmission. Secondly, the heightened data rate resulting from the increased number of nodes calls for robust data synchronization mechanisms to maintain temporal coherence and mitigate potential data inconsistencies. Addressing these challenges associated with the growing scale of data capture and synchronization overhead is paramount to facilitate seamless operation and enable the effective utilization of distributed sensing techniques in mmWave communication systems. One promising approach to overcome these challenges is reducing the data traffic volume between the basestation and the distributed nodes by selectively **transferring only critical information**. For instance, in the case of a distributed node equipped with a camera, rather than transmitting the entire image, a more efficient strategy is to extract the environment semantics locally. These environment semantics comprise relevant information about the wireless environment, such as the presence of different vehicles in the scene and their relative locations. By focusing on transmitting only this critical information, the data traffic between the distributed nodes and the basestation can be significantly reduced, alleviating the storage, processing, and transmission burdens.

To address the challenges mentioned above, this paper focuses on designing efficient strategies for extracting **environment semantics from RGB images (such as the masks and bounding boxes of the objects of interest)** to facilitate the beam prediction process in a realistic V2I communication scenario with two distributed nodes. In this scenario, we consider a more complex environment with multiple probable objects, requiring advanced techniques for accurate prediction. Given the multi-candidate nature of the scenario, our solution is designed to efficiently extract environment semantics, identify the transmitter in the scene, and predict the optimal beam in real time. To accomplish this, we leverage a sequence of RGB images captured by the distributed nodes. However, one key challenge that still remains is the latency associated with transferring these environment semantics to the basestation for beam prediction. It is important to note here that our solution can also be extended to predict future beams, enhancing the proactive nature of the system. By adopting a proactive approach and predicting future beams, we can effectively overcome the latency issue, ensuring timely and accurate beam prediction in distributed sensing-based mmWave communication systems. Next, we present the proposed solution in detail.

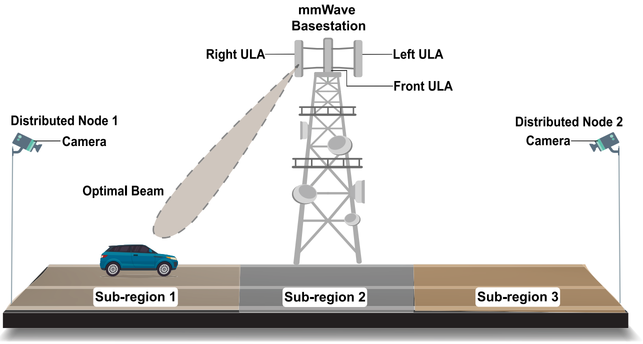


FIGURE 2. The selection process of the ULA, the sub-region, and the corresponding distributed node.

IV. PROPOSED SOLUTION

This section provides a comprehensive overview of the proposed solution for distributed environment semantic-aided communication. Note that our focus in this work is primarily on mobile vehicles within the context of vehicle-to-infrastructure communication. We divide the region served by the basestation into three sub-regions, where each sub-region corresponds to one of the phased arrays of the basestation as shown in Fig. 2. Furthermore, we include two distributed nodes in the system with one node located to the left of the basestation and the other to the right. At any given time t, the selection of sensing data for further processing and beam prediction depends on the user's location in the wireless environment. For instance, if the user is situated in the sub-region to the right of the basestation, the RGB images captured by the right distributed node (distributed node 1) are utilized for beam prediction. It is important to note here that the selection of the distributed node for further processing and beam prediction does not rely on the user's position

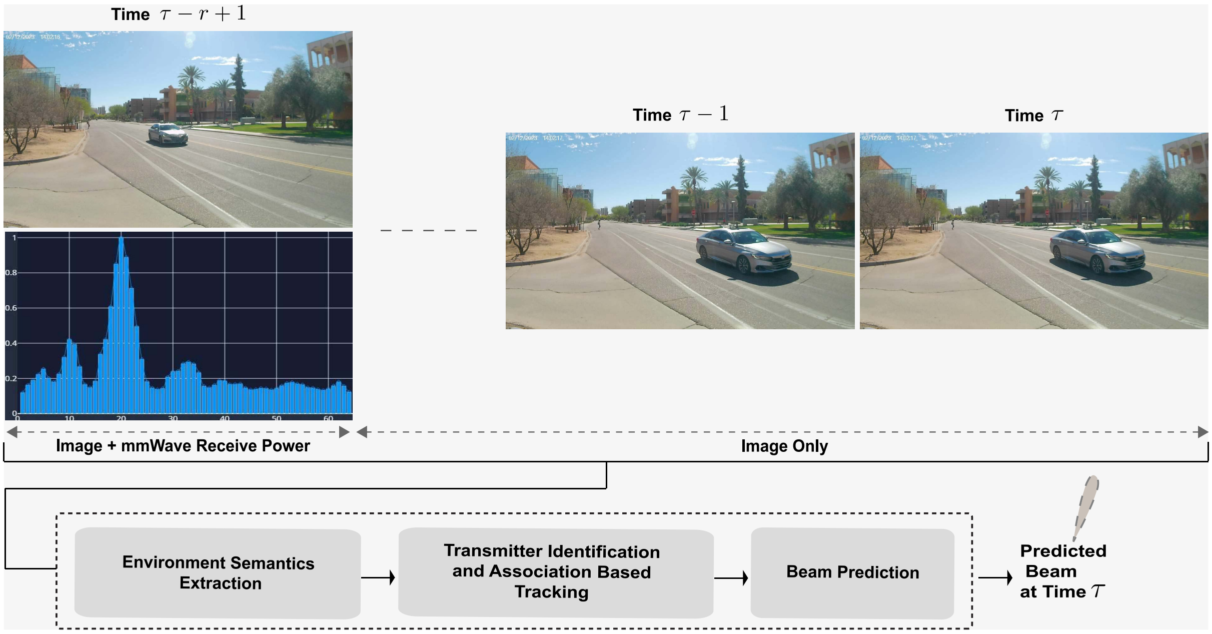

Time $\tau - r + 1$
Time $\tau - 1$
Time τ
1
0.8
0.6
0.4
0.2
0
0
10
20
30
40
50
60
Image + mmWave Receive Power
Image Only
Environment Semantics Extraction
Transmitter Identification and Association Based Tracking
Beam Prediction
Predicted Beam at Time τ

FIGURE 3. The different stages of the proposed solution. In the first stage, environment semantics are extracted from the raw RGB images at the distributed nodes. The second stage, performed at the basestation, involves identifying the transmitter in the initial frame and tracking it across the subsequent frames. Finally, in the third stage, the semantic information gathered in the second stage is used for beam prediction at the basestation.

data (GPS position). Instead, we utilize the receive power vector, which provides valuable directional information that aids in determining the optimal beam index. By utilizing the optimal beam index, we can select one of the ULAs. Next, depending on the selected ULA, we approximate the sub-region where the user is located. This approximation, in turn, helps identify the specific distributed node from which to utilize the sensing data. This approach ensures efficient utilization of the sensing data based on the user's location within the coverage area.

The proposed solution comprises three stages as depicted in Fig. 3. The first stage would be carried out at the distributed nodes, while the second and third stages would take place at the basestation. In the first stage, object masks and bounding boxes of potential users would be extracted at the distributed nodes. In the second stage, executed at the basestation, the received power vector would be used to identify the transmitter from multiple candidates. The transmitter would then be tracked over the subsequent $r - 1$ frames using the nearest neighbor algorithm. Lastly, in the third stage, the basestation would leverage the transmitter's semantic information from the current and past r frames to predict the current optimal beam index. Since the semantic information is very lightweight, the transmission of semantic information from the distributed nodes to the basestation is assumed to be instantaneous. A proposed real-time communication system could be designed with each distributed node equipped with a single antenna for uplink transmission, while the basestation uses its ULAs to form a beam for efficient reception of data from these nodes.

A. STAGE 1: ENVIRONMENT SEMANTICS EXTRACTION

The first stage of the proposed solution aims to extract environment semantics from RGB images, as shown in Fig. 4. The primary objective of this stage is to accurately and efficiently capture information that represents the objects of interest in the wireless environment while also minimizing the required data storage compared to the original sensing modality (i.e., the images themselves). To achieve this, we utilize the state-of-the-art COCO [30] pre-trained object detection and image segmentation model, YOLOv7 [23]. In particular, we aim to extract two crucial types of environment semantics: bounding boxes and binary masks. Bounding boxes, denoted as $\mathbf{X}_{\text{BBox}}[t] \in \mathbb{R}^{U\times 4}$, serve as representations for potential users within the wireless environment, where U is the total number of detected objects in the RGB image. Each row of $\mathbf{X}_{\text{BBox}}[t]$ contains a bounding box vector $[x_c, y_c, w, h]$, where x_c, y_c, w, and h denote the x-center, y-center, width, and height of the detected object, respectively. These bounding boxes provide essential spatial information about the potential users. Additionally, we generate binary masks, represented as $\mathbf{X}_{\text{Mask}}[t] \in \mathbb{R}^{\hat{W}\times\hat{H}}$, where $\hat{W}$ and $\hat{H}$ correspond to the downsampled width and height of the image mask, respectively. These masks offer more detailed and fine-grained depictions of the spatial extent of the potential users within the wireless environment. Furthermore, the image segmentation model employed in our solution not only outputs the binary masks but also provides the bounding box information for the detected objects. Let $\mathbf{X}_{\text{B-Mask}}[t] \in \mathbb{R}^{U\times 4}$ denote the bounding boxes extracted during the image segmentation.

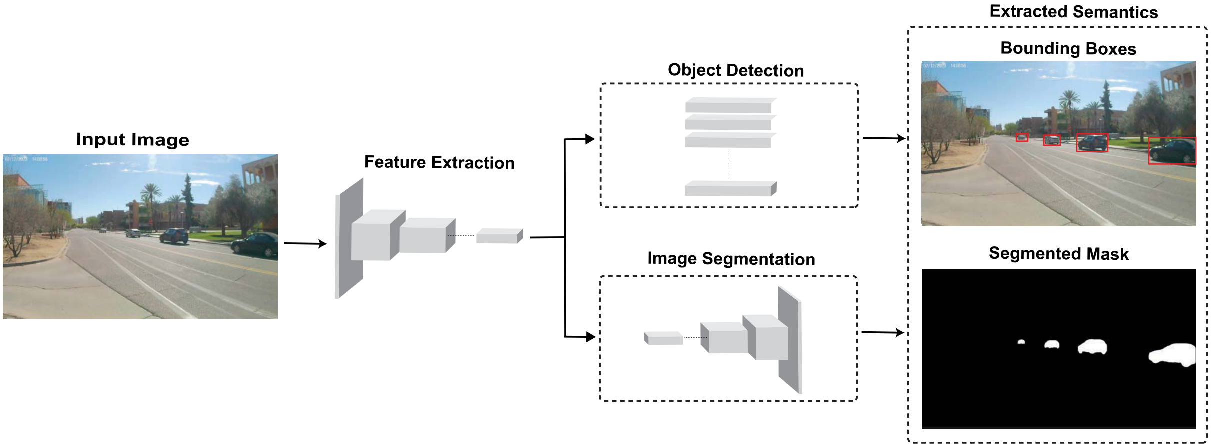


FIGURE 4. The environment semantics extraction stage in our proposed solution. A camera installed at the distributed node captures real-time images of the wireless environment, which a machine learning model then processes to extract the bounding boxes and masks of the mobile objects present in the images.

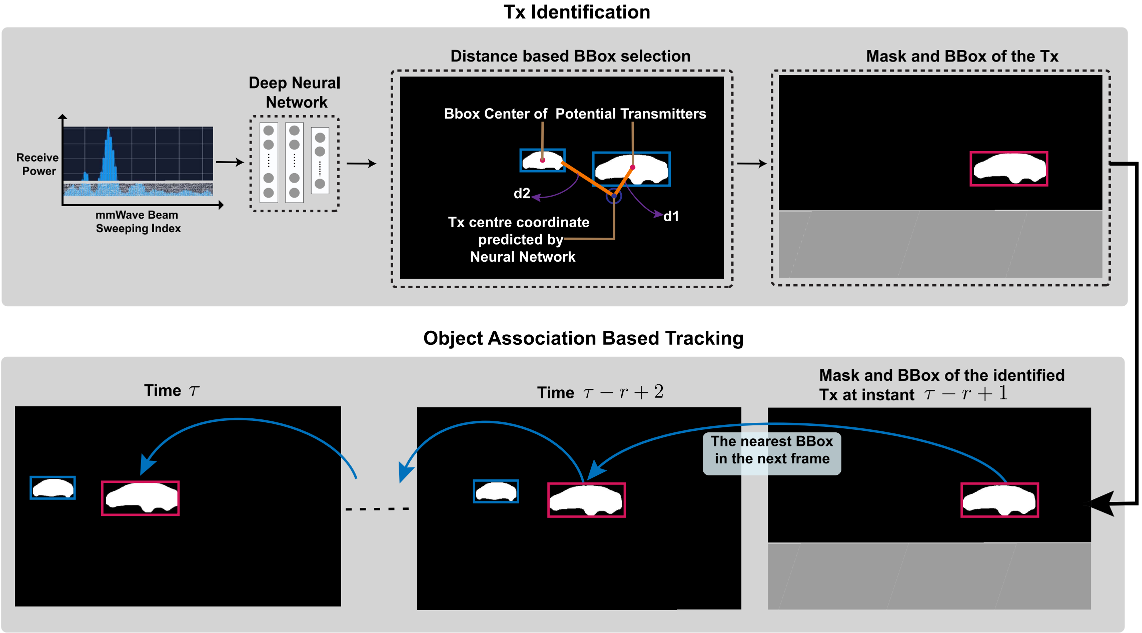

Tx Identification
Receive Power
mmWave Beam Sweeping Index
Deep Neural Network
Distance based BBox selection
Bbox Center of Potential Transmitters
d2
d1
Tx centre coordinate predicted by Neural Network
Mask and BBox of the Tx
Object Association Based Tracking
Time τ
Time $\tau - r + 2$
Mask and BBox of the identified Tx at instant $\tau - r + 1$
The nearest BBox in the next frame

FIGURE 5. The transmitter identification and object association-based tracking module. The transmitter is identified in the first frame using the receive power vector and then tracked for the remaining frames using the nearest neighbor algorithm.

B. STAGE 2: TRANSMITTER IDENTIFICATION AND TRACKING

By adopting YOLOv7, which allows the simultaneous generation of bounding boxes and masks, we eliminate the need for separate runs, resulting in faster inference speed. Moreover, YOLOv7 achieves a significant reduction of approximately 40% in parameter size compared to other real-time object detectors, leading to enhanced computational efficiency. Utilizing pre-trained models based on the COCO dataset is advantageous as they can detect most of the relevant objects commonly encountered in wireless environments, such as cars, bikes, and people. We now present the second stage of our proposed solution. Specifically, two tasks must be accomplished to predict the optimal beam index. Firstly, the transmitter must be identified among the detected objects, and secondly, the transmitter needs to be tracked over the subsequent $r - 1$ samples. To address these challenges, we introduce a two-stage solution encompassing transmitter identification and object association-based tracking, as presented in Fig. 5. In the following sections, we provide a comprehensive description of our proposed transmitter identification and object association-based tracking solution.

1) TRANSMITTER IDENTIFICATION

It refers to accurately determining the transmitter's location within the wireless environment using the extracted semantic information. The adopted image segmentation model not only provides binary masks but also outputs the bounding box information for the detected objects. In the task of transmitter identification, we leverage the extracted bounding boxes from both types of environment semantics. Therefore, the objective is to leverage the receive power vector $\mathbf{p}[\tau - r + 1]$ from the ULA that has the user in its field of view and the semantic information of masks and bounding boxes at time $t = \tau - r + 1$ to predict the center coordinates of the transmitter's bounding box $\mathbf{b}_{\text{Tx}}[\tau - r + 1] \in \mathbb{R}^{2\times 1}$ within the image. For this, we employ a prediction function g_η, parameterized by a set of parameters η, which maps the receive power vector to the predicted bounding box center coordinates $\hat{\mathbf{b}}_{\text{Tx}}$. Mathematically, this can be expressed as:

$$g_\eta : \mathbf{p}[t] \rightarrow \hat{\mathbf{b}}_{\text{Tx}}[t]. \tag{7}$$

To train the prediction function, we construct a dataset $\mathcal{D}_2$ comprising pairs of mmWave receive power vectors $\mathbf{p}_v$ and their corresponding ground-truth bounding box center coordinates of the transmitter $\mathbf{b}_{\text{Tx}v}$. This dataset is a subset of the larger dataset $\mathcal{D}$, and it contains V samples, such that $\mathcal{D}_2 = \{(\mathbf{p}_v, \mathbf{b}_{\text{Tx}v})\}_{v=1}^{V}$. We do not have access to ground-truth bounding box coordinates for the transmitter vehicle, which are required to construct the dataset $\mathcal{D}_2$. To address this issue, we manually select samples that contain only the transmitting vehicle. These samples are then processed using the YOLOv7 deep learning model, and the resulting bounding boxes are manually reviewed for accuracy. The bounding boxes generated by YOLOv7 for images containing only the transmitter vehicle are subsequently used as the ground-truth annotations for the dataset $\mathcal{D}_2$. The goal is to minimize the error between the predicted and ground-truth center coordinates of the transmitter's bounding box across all the samples in $\mathcal{D}_2$. This optimization problem can be formulated as:

$$g^\star_{\eta^\star} = \underset{g_\eta}{\text{argmin}} \frac{1}{V} \sum_{v=1}^{V} \left\| \hat{\mathbf{b}}_{\text{Tx}v} - \mathbf{b}_{\text{Tx}v} \right\|^2, \tag{8}$$

where $g^\star_{\eta^\star}$ represents the optimal prediction function that minimizes the squared l_2 norm of the error between the predicted and ground-truth bounding box center coordinates.

To learn g_η, we use a two-layered fully connected neural network with 512 nodes in each layer. The obtained $\mathbf{b}_{\text{Tx}}$ from g_η is not intended to be the final prediction but rather only an initial estimate. We utilize this initial estimate together with the semantic information at that time instant to identify the bounding box and mask of the object responsible for the received signal. The bounding box of the transmitter is identified by locating the bounding box in $\mathbf{X}_{\text{BBox}}[\tau - r + 1]$ and $\mathbf{X}_{\text{B-Mask}}[\tau - r + 1]$ whose center coordinate is closest to $\hat{\mathbf{b}}_{\text{Tx}}[\tau - r + 1]$. For instance, consider an image with two candidates for transmitter, each with its own bounding box. Let d_1 and d_2 represent the distances between $\hat{\mathbf{b}}_{\text{Tx}}[\tau - r + 1]$ and the bounding boxes of the first and second candidates, respectively, as shown in Fig. 5. The bounding box with the smaller distance, $\min(d_1, d_2)$, will be selected as that corresponding to the transmitter. Furthermore, we determine the transmitter's mask by identifying the group of pixels within the transmitter's bounding box. It is worth noting here that we assume a transmitter is present in the wireless environment at each time step t. The next step involves tracking the bounding box and mask of the transmitter for the next $r - 1$ samples.

2) OBJECT ASSOCIATION BASED TRACKING

In the previous step, we successfully identified the transmitter in the first image sample (in a sequence of r images). However, to predict the current optimal beam index, tracking the transmitter's location throughout the remaining $r-1$ samples is essential. This section presents two distinct approaches for transmitter tracking for the different environment semantics: (i) Bbox-based object tracking and (ii) mask-based object tracking.

(i) Bbox-based Object Tracking: Numerous state-of-the-art algorithms [31], [32], [33] have been proposed in the field of multiple object tracking (MOT). However, considering the emphasis of this work on V2I communication, primarily involving mobile vehicles, a simple Euclidean distance-based object association algorithm is adopted [34]. This algorithm determines the transmitter in the next sample by finding the bounding box in $\mathbf{X}_{\text{BBox}}$ (of the following sample) with the closest center coordinate to the bounding box in the current sample as shown in Fig. 5. The key underlying idea is that, for two consecutive image samples, the distance between the center coordinates of the bounding box will be the smallest for the same object compared to other objects in the scene.

(ii) Mask-based Object Tracking: To facilitate object association-based tracking using masks, the median color value of mobile vehicles can be utilized. Using binary masks, we extract the color information of all the detected vehicles at the distributed nodes. This is achieved by performing a Hadamard product between the binary mask and the RGB image, followed by calculating the mean value of the pixels where the binary mask contains a 1. We filter out the vehicles whose color does not match with that of the transmitter identified in the first sample. Let $\rho \in \mathbb{R}^3$ denote the median RGB color value of a probable candidate. Let ρ_{Tx} and ρ_z further represent the median RGB color values of the transmitter and the z^{th} potential user in the mask, respectively. The potential user is considered a candidate for subsequent object association if the following criterion is satisfied

$$\|\rho_{\text{Tx}} - \rho_z\|_F \leq \epsilon, \tag{9}$$

where ϵ is a tunable threshold. The decision of which candidate will be retained in the list of potential users depends on the choice of ϵ, which we have kept as 20. In the context of this paper, we refer to this filtering step that utilizes color information as "semantic-aided filtering". To identify the transmitter's mask in the subsequent sample, we select the mask of the vehicle with the shortest distance to the transmitter's mask in the previous frame as the nearest neighbor. Consequently, this selected mask is designated as the transmitter's mask in the subsequent frame. By incorporating color similarity as a refining criterion in the object association process, we enhance the accuracy of the tracking algorithm.

It is important to note that the proposed transmitter identification and object association-based tracking are not fully effective in all conditions, particularly when the transmitter vehicle is occluded. In cases of vehicle overlap, stage 1 of environment semantics extraction remains largely unaffected, as YOLOv7 is robust to partial occlusion and can provide bounding boxes and masks unless the vehicle is heavily occluded. Occlusion poses a greater challenge during transmitter identification and tracking. If the occluding vehicle is closer to the predicted coordinates, the transmitter may be incorrectly identified. This issue is mitigated when the transmitter and occluding vehicles are traveling in the same lane and direction, as a misidentified transmitter may still result in a sub-optimal beam for the transmitter vehicle with a reasonable rate. Semantic-aided filtering further mitigates incorrect transmitter identification in association based tracking by filtering out vehicles whose color does not match the initially identified transmitter. The rate performance together with the advantages and limitations of semantic-aided filtering are discussed in Section VI. In our future work, we can focus on utilizing multiple camera views to improve robustness against occlusion by providing additional perspectives of the transmitter vehicle, which will help in accurately identifying and tracking the transmitter even in challenging conditions.

C. STAGE 3: BEAM PREDICTION

This section introduces the final step of our proposed solution, which aims to predict the optimal beam index for the transmitter. The goal is to use the sequence of bounding-box coordinates or image masks obtained from the previous object association-based tracking to make this prediction. However, since we are interested in predicting the current optimal beam index rather than future ones, it may be sufficient to use the available semantics for the current time step only. In order to address this, we propose two approaches: (i) Single instance-based beam prediction and (ii) Sequence-based beam prediction. In the single instance-based approach, we use the bounding box or mask at the current time step t to predict the optimal beams. For the sequence-based approach, we utilize the sequence of r available environment semantics to make the prediction. Next, we present both of these proposed solutions.

1) SINGLE INSTANCE-BASED BEAM PREDICTION

Due to the distinct nature of the environment semantics (bounding box and image mask), each requires a specific approach for predicting the optimal beam index. We present both solutions, highlighting their effectiveness in utilizing the corresponding environment semantic for accurate beam prediction.

1) *Bounding Box-based Beam Prediction:* This baseline model takes the user's bounding box at the current time instant t as input and predicts the corresponding beam index. Mathematically, we can express this as

$$\omega : \mathbf{x}_{\text{bbox}}[t] \rightarrow \hat{\mathbf{f}}[t], \tag{10}$$

where ω represents the mapping function and $\mathbf{x}_{\text{bbox}}[t] \in \mathbb{R}^{2\times 1}$ represents the center coordinate of the transmitter vehicle's bounding box at time t. This mapping function takes the form of a two-layered fully connected neural network with 512 neurons in each layer as our baseline model. Fully connected neural networks (FCNNs) excel at handling structured data by leveraging the network weights to capture the relationships among input elements. Additionally, FCNNs establish dense connections between adjacent layers, enabling them to learn intricate associations between input elements.

2) *Mask-based Beam Prediction:* In this step, similar to bounding box-based beam prediction, we utilize another mapping function that takes the transmitter vehicle's mask at the current time instant t as input and predicts the corresponding beam index as follows

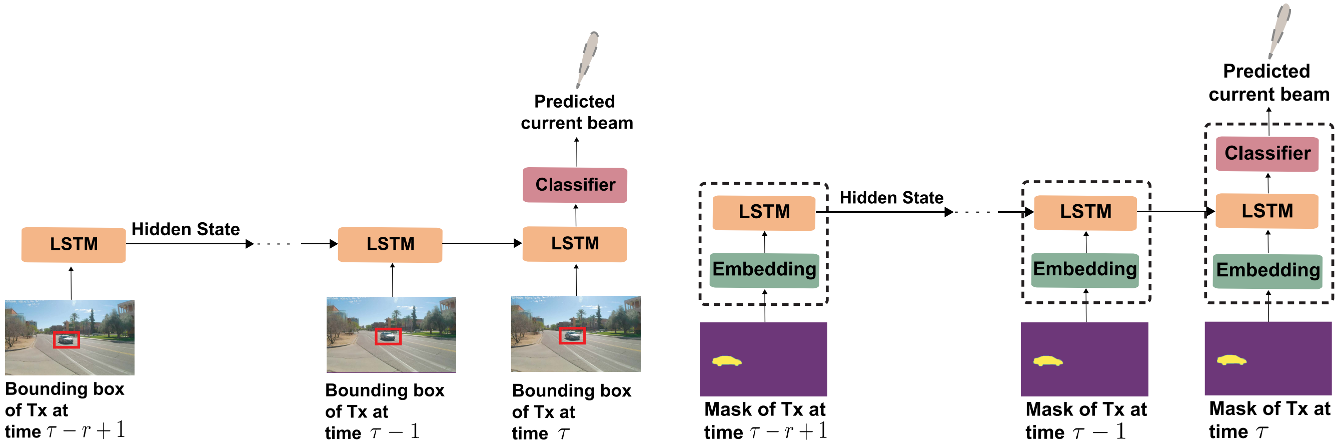
$$\beta : \mathbf{x}_{\text{mask}}[t] \rightarrow \hat{\mathbf{f}}[t], \tag{11}$$

where β represents the mapping function for this task and $\mathbf{x}_{\text{mask}}[t] \in \mathbb{R}^{\hat{W}\times\hat{H}}$ represents the transmitter vehicle's mask at time t. We note that convolutional neural networks (CNNs) have demonstrated superior performance and robustness in leveraging spatial relationships among neighboring pixels in image data. Therefore, the mapping function β for this baseline model of mask-based beam prediction takes the form of a simple CNN model, similar to LeNet [35], consisting of two convolutional layers followed by five fully connected layers.

2) SEQUENCE-BASED BEAM PREDICTION

We use a recurrent neural network (RNN) [36], [37] which processes a sequence of semantic representations of the transmitter and predicts the optimal beam index. We chose the RNN architecture for two reasons. First, RNNs have achieved good accuracy in various sequential modeling tasks, such as natural language processing and speech recognition, due to their ability to extract crucial information from previous sensory data. This allows the model to capture the temporal dependencies and patterns in the semantic information, enabling accurate beam prediction. Second, as compared to other neural network architectures like Transformers [38], RNNs offer advantages in terms of computational complexity and inference time. In this section, we present the two different solutions developed to target the different semantic modalities.

1) *Bounding Box-based Beam Prediction:* We utilize a mapping function that takes a sequence of the transmitter vehicle's bounding boxes over r consecutive



(a) RNN model taking bounding box as input

(b) RNN model taking mask as input

FIGURE 6. The proposed RNN models for beam prediction. The first RNN model, shown in (a), takes the bounding boxes of the transmitter as input. Each unit consists of an LSTM block and a classifier block. The RNN model shown in (b) takes masks of the transmitter as input. Each unit consists of an embedding block, an LSTM block, and a classifier block.

time stamps and predicts the corresponding beam index at the last time step. Mathematically, we can express this as

$$\gamma : \{\mathbf{x}_{\text{bbox}}[t]\}_{t=\tau-r+1}^{t=\tau} \rightarrow \hat{\mathbf{f}}[\tau]\}, \tag{12}$$

where γ represents the mapping function for bounding box sequence-based beam prediction. The mapping function γ takes the shape of a RNN model. In Fig. 6(a), we present the block diagram of the proposed RNN model for beam prediction using bounding boxes as inputs. This model comprises r repeated blocks, each consisting of a single layer Long Short-Term Memory (LSTM) unit. LSTM is a type of RNN specifically developed to address the challenge of learning long-term dependencies in sequence data. By incorporating gates and memory cells, LSTMs effectively manage information flow over time steps and mitigate the vanishing gradient issue encountered in conventional RNNs. The center coordinates of the bounding box vector, $\mathbf{x}_{\text{bbox}}[t] \in \mathbb{R}^{2\times 1}$, are directly fed as input into the LSTM block. Following the LSTM unit, a fully connected layer acts as the classifier. The output of the classifier block is a score vector $\boldsymbol{\xi} = [\xi_1, \xi_2, \ldots, \xi_M]$. At the output of this fully connected layer, we utilize the cross-entropy activation function. The m^{th} element of the score vector corresponds to the m^{th} beam in the codebook. The beam index with the highest score is the predicted beam.

2) *Mask-based Beam Prediction:* Similar to bounding box sequence based-beam prediction, we utilize a mapping function that takes a sequence of the transmitter vehicle masks over r consecutive time stamps and predicts the beam index at the last time step. We can formally express this as

$$\psi : \{\mathbf{x}_{\text{mask}}[t]\}_{t=\tau-r+1}^{t=\tau} \rightarrow \hat{\mathbf{f}}[\tau]\}, \tag{13}$$

where ψ represents the mapping function for this task. This mapping function again takes the shape of a RNN as shown in Fig. 6(b). This model also consists of r repeated blocks, each comprising an LSTM unit. Due to the structural differences between masks and bounding boxes in terms of semantic representation, an additional embedding block is included in this model. The embedding block transforms the high-dimensional semantic mask $\mathbf{x}_{\text{mask}}[t]$ into a low-dimensional vector $x[t] \in \mathbb{R}^{\nu\times 1}$, where ν denotes the input state size of the LSTM, reducing the trainable parameters of the model. The input state size of the LSTM unit is $\nu = 64$. The embedding block utilizes a simple CNN model, similar to LeNet [35], consisting of two convolutional layers and three fully connected layers. The output from the final layer in the embedding block is used as input to the LSTM unit. The remaining components, including the LSTM block and the classifier block with the cross-entropy activation function, are kept same as that in the bounding box-based model.

In conclusion, we propose two different models designed to effectively capture the relevant information from the semantic representations and predict the optimal beams accurately.

D. BASELINE SOLUTIONS

In order to evaluate the accuracy of our proposed transmitter identification solution and the subsequent object association step, it is important to have ground-truth bounding box center coordinates of the transmitter in cases where there are multiple mobile vehicles as potential candidates for the transmitter. We shall refer to these instances as multi-candidate scenarios. Unfortunately, we do not have access to ground-truth bounding box coordinates of the transmitter vehicle. In Section IV-B1, we addressed this limitation by manually selecting the samples that contain only the transmitter vehicle. These selected samples are then processed using the YOLOv7 deep learning model, and the resulting bounding boxes are manually validated. These bounding boxes generated by YOLOv7 are subsequently used as the ground-truth bounding boxes for training the transmitter identification model. However, these bounding boxes are only available for cases where only the transmitter vehicle is present in the image, not for multi-candidate scenarios. We note that the dataset used in this study provides highly accurate position data of the transmitter. This position data offers reliable and granular information about the transmitter's location, enabling the model to make more precise predictions. Therefore, to test the accuracy of the proposed transmitter identification solution and the subsequent object association step, we utilize a position-aided transmitter identification approach. In this approach, a machine-learning model predicts the center coordinates of the transmitter's bounding box based on its GPS position. The network architecture for this position-aided identification is identical to the receive power vector-aided model. Both consist of a two-layered fully connected neural network with 512 neurons in each layer. However, unlike the receive power-aided transmitter identification solution, which involved identifying the transmitter only at the initial time step and subsequently tracking it across the subsequent frames, position-aided transmitter identification performs transmitter identification at every time step of the sequence. The results obtained from the position-aided transmitter identification will serve as a baseline for evaluating the performance of (i) the proposed transmitter identification model using the metric of comparative accuracy and (ii) the subsequent object association-based transmitter tracking using the metric of association accuracy. Both these metrics are defined in detail in Section VI.

E. SCALABILITY AND COMPUTATIONAL EFFICIENCY

The effectiveness of our sequence-based beam prediction solution depends on both its accuracy and its ability to manage distributed nodes efficiently while meeting the real-time latency requirements of V2I communication. As presented in Section V, our current testbed implementation demonstrates this capability in a controlled, asynchronous environment. Each distributed node in the testbed is equipped with a camera and GPS receiver, capturing visual data and precise timing information asynchronously. Data synchronization is achieved through post-processing, where we align data from different nodes and the basestation. This asynchronous approach allows us to utilize data from only one distributed node at a time, based on the user's location, effectively managing the current scale of our system without real-time constraints. However, real-world deployment scenarios present additional considerations beyond our current asynchronous testbed. These include managing real-time data from multiple nodes simultaneously and handling potential overlapping sensor coverages. The complexity increases in multi-transmitter scenarios, where real-time decision-making becomes crucial. While our current asynchronous approach effectively manages some of these aspects through post-processing, real-world deployments may require further adaptations to address these challenges. The exploration of these real-time adaptations and scaling strategies will be the focus of our future work.

Computational Complexity Analysis: Next, we analyze the computational complexities of each step in our proposed solution based on our current post-processing implementation. This analysis provides insights into the system's performance and its potential for future real-time operation in more complex environments. In our current asynchronous testbed, all processing occurs offline after data collection. The environment semantic extraction step, performed during post-processing to simulate distributed node computations, uses the YOLOv7 model and requires approximately 29.5 ms per frame when run on an Nvidia T4 GPU. The beam prediction step, simulating basestation processing, uses our proposed machine learning models and takes about 1-2 ms per prediction when executed on an Nvidia RTX A5000 GPU during post-processing. Data transmission time is not explicitly measured in our current setup, as data from distributed nodes is collected and processed offline. However, this aspect will be crucial for real-time implementations and is a key area for future investigation and optimization. The sub-region selection process, based on the user's location, has a complexity of O(k) for k sub-regions, with its latency being negligible compared to other steps in our post-processing pipeline. While these measurements are obtained during post-processing and don't reflect real-time performance, they suggest that our proposed solution has the potential to meet real-time requirements in future implementations. However, they also highlight areas where further optimization may be necessary for real-time deployment, particularly in managing simultaneous data from multiple nodes and handling real-time decision-making in multi-transmitter scenarios.

Storage and Transmission Efficiency: We now discuss how utilizing environment semantics lowers the storage and transmission requirements. Four integer values can represent the bounding box of a vehicle. A high-definition RGB image captured at the distributed node is approximately 5.93 MB in size. If there are four vehicles in the image, the bounding boxes of these vehicles can be represented by 16 integer values, occupying just 64 bytes of storage. This is about five orders of magnitude smaller than the storage and transmission requirements of the full RGB image. The storage and transmission requirements of the masks would depend on the camera angle. Fig. 4 shows a mask of the image captured at unit 3. The image contains four vehicles. This mask can be efficiently stored and transmitted by encoding only the mask pixel values and the vehicles' bounding boxes. This approach reduces storage and transmission requirements by approximately three orders of magnitude compared to transmitting the entire RGB image.

Overlapping Sensor Coverage: We now present strategies to manage sensor coverage overlap at distributed nodes,

TABLE 1. Complexity analysis: Time taken (ms).

ML Model	Mask-YOLOv7	BBox-YOLOv7	Tx. Id - FCNN	Mask-CNN	Bbox-FCNN	Mask-LSTM	BBox-LSTM
Inference Latency (milliseconds)	29.5	29.5	0.037	1	0.4	1.46	0.37

which can cause variations in optimal beam prediction due to differences in the semantic information provided by each node. Recall that in the current setup, we have the optimal beam index information at the first time stamp and predict the beam at the r^{th} timestamp, where r is the length of the observation window. Further recall that the region served by the basestation is divided into sub-regions. Using the optimal beam index at the first timestamp, we select the sub-region and the corresponding distributed node whose semantic data is used for beam prediction at the r^{th} timestamp. The current setup has minimal overlap between the sub-regions covered by the distributed nodes and their camera fields of view such that, at a given time instant, only semantic data from a single distributed node is utilized for beam prediction at the basestation. However, we expect that when systems try to densify the sensor network, then increasing the number of nodes could lead to more overlaps. This can result in variations in beam predictions due to the different semantic information provided by each node. Addressing this issue remains an open problem and warrants further investigation. In the next paragraph, we briefly discuss some directions for potential solutions.

An initial solution for the problem discussed above may involve training the node-specific beam prediction models with additional data samples from their respective coverage areas, particularly in the overlapping regions. By including sufficient training samples from these overlapping areas, along with their associated ground truth beam indices, each model will learn to make consistent beam predictions regardless of which node's perspective is used. An alternative future approach to address the variations in beam predictions, due to the differing semantic information from each node, could be to transition from individual models to a unified machine learning model that performs sensor fusion across all distributed nodes. The key idea is to incorporate learnable parameters in the neural network architecture to learn optimal weights for combining features from different distributed nodes based on the user's location (identified from the first beam). One promising way to achieve that is through an attention layer, where the network would learn to generate attention weights for each distributed node's features based on the user's location. For instance, if the user is in a region covered by both node 1 and node 2, the attention mechanism would learn to assign appropriate weights to features from both nodes for beam prediction. These weights, learned during training, would help the model determine how to best combine features from multiple nodes in overlapping regions, while naturally focusing on a single node's features in non-overlapping regions. This structured approach to learning feature combination weights enables sophisticated feature fusion, potentially leading to more accurate beam predictions in complex scenarios with multiple overlapping node coverage.

Latency Mitigation Strategies: Finally, we discuss the strategies to mitigate the latency challenges in the proposed solution. The time taken for extracting environment semantics can be significantly reduced by using more computationally efficient models such as MobileNetv2 [22], which utilizes about 3.5 million parameters compared to YOLOv7's 37.5 million parameters. Note that the time taken for transmitting environment semantics will depend on the type of environment semantic information being transmitted, with bounding boxes requiring less time than masks. At the basestation, beam prediction takes about 1-2 ms, which is relatively low. It is important to highlight here that our solution can also be extended to predict future beams, enhancing the proactive nature of the system. By adopting a proactive approach and predicting future beams, we can effectively overcome the latency issue, ensuring timely and accurate beam prediction in distributed sensing-based mmWave communication systems.

V. TESTBED DESCRIPTION AND AI-READY DATASET

To assess the effectiveness of our proposed distributed sensing-aided beam prediction solution, we employ the DeepSense 6G dataset [28]. DeepSense 6G is a comprehensive real-world dataset specifically designed for sensing-aided wireless communication applications. It encompasses diverse multi-modal data, including vision, mmWave wireless communication, GPS, LiDAR, and radar. In this section, we provide an overview of scenario 41 adopted from the DeepSense 6G dataset and subsequently analyze the AI-ready dataset used to evaluate the performance of our proposed solution.

A. DEEPSENSE 6G TESTBED

The study adopts scenario 41 of the DeepSense 6G dataset specifically designed to study distributed sensing-aided communication in a multi-user scenario. The hardware testbed and the locations for collecting this data are shown in Fig. 7. The DeepSense testbed 7 is utilized for this data collection. It consists of (i) three stationary units, one acting as the basestation and the other two acting as the distributed nodes, and (ii) a mobile transmitter. All the stationary units, namely the basestation (unit 1), the first distributed node (unit 2), and the second distributed node (unit 3), are equipped with an RGB camera. The distributed units are also equipped with a GPS receiver. The basestation further adopts three 16-element (M=16) 60 GHz-band phased arrays, and it receives the transmitted signal using an over-sampled

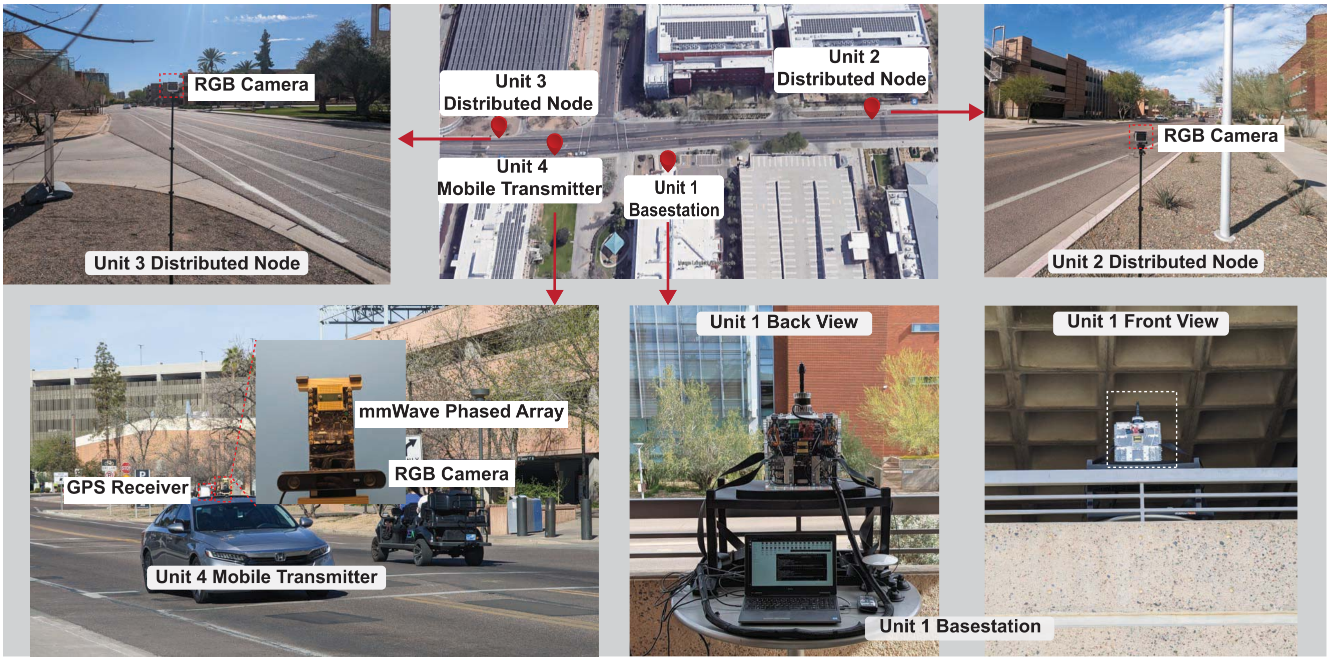


FIGURE 7. The testbed setup for the DeepSense 6G AI-ready dataset used in our experiments. It consists of a stationary unit (unit 1), acting as the basestation, a mobile unit (unit 4), acting as the transmitter, and two distributed nodes (unit 2 and unit 3).

codebook of 64 pre-defined beams ($Q = 64$). The mobile unit (unit 4) is a vehicle equipped with a mmWave transmitter and GPS antenna/receiver. The transmitter consists of a quasi-omni antenna constantly transmitting (omnidirectional) at the 60 GHz band.

It is crucial to note that this setup reflects our data collection process, not a real-time system implementation. Our current testbed implementation uses an asynchronous data collection approach in which all data processing occurs after data collection. Each distributed node and the base station are equipped with a camera and a GPS receiver. At each node, cameras capture images at 60 frames per second, while the GPS receiver operates at 10 samples per second. Additionally, at the base station, receive power measurements are recorded at 10 Hz. During data collection, each sample from every modality (images, GPS data, and receive power measurements) is recorded with its corresponding UTC timestamp. We align the sensor data collected at different time instances and sampling rates to create a uniform set of samples at a single sampling rate of 10 Hz. The GPS receiver serves as the reference modality, with data from other sensors aligned by selecting the nearest sample to each GPS timestamp. The maximum synchronization error for each modality (except the GPS) is given by $\frac{1}{2\text{Fs}}$, where Fs represents the sampling frequency of the modality. For more information regarding the data collection setup, testbed, and synchronization method, please refer to [28].

B. DEEPSENSE 6G AI-READY DATASET

The evaluation of the proposed distributed sensing-aided beam prediction solution necessitates real-world data obtained from a wireless environment. As such, we utilize scenario 41 of the DeepSense 6G dataset. This dataset is collected at McAllister Ave., Tempe, during the daytime. The speed limit on the road where this testbed was deployed was about 48.3 km/h. Images at the distributed nodes are available at a frequency of 10 Hz, meaning one image is available for every 100 ms. The time duration between timestamp τ and $\tau - 1$ is therefore 100 ms. Throughout the data collection process, the road was actively utilized by other vehicles, pedestrians, and cyclists. The raw dataset includes RGB images from both the basestation (unit 1) and the distributed nodes (unit 2 and unit 3), receive power vectors from the three ULAs, and the user's GPS position. Fig. 8 shows the sample dataset images from each unit. We process the RGB images from unit 2 and unit 3 using a sliding window of size $r = 5$, generating time-series sequences of RGB images for each unit. The AI-ready dataset comprises these processed RGB image sequences, along with the receive power at the initial time step, $\mathbf{p}[\tau - r + 1]$, and the optimal beam index $\mathbf{f}^{\star}$ at the last time step of each sequence. Furthermore, it also incorporates the transmitter's GPS position at every time instant. Only the sequences where the transmitter car is present in the camera's field of view are retained in the AI-ready dataset. There are 2991 and 5476 image sequences for unit 2 and unit 3, respectively, which are further split into training, validation, and testing categories with a ratio of 70:20:10. Our previous works [10], [11], [22] have indicated that the accuracy of vision-based beam prediction solutions using RGB images captured at night can approach that achieved with daytime images, with some additional processing of

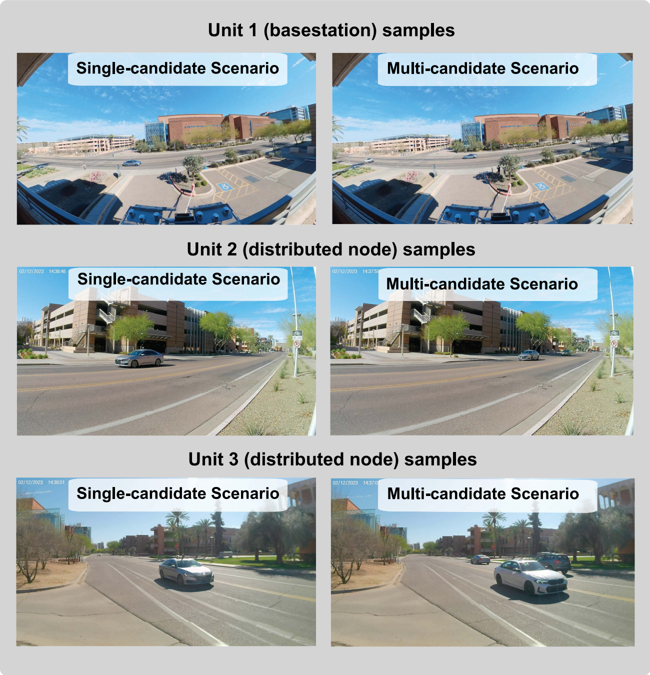

Unit 1 (basestation) samples
Single-candidate Scenario
Multi-candidate Scenario
Unit 2 (distributed node) samples
Single-candidate Scenario
Multi-candidate Scenario
Unit 3 (distributed node) samples
Single-candidate Scenario
Multi-candidate Scenario

FIGURE 8. The RGB image samples captured at the basestation (unit 1) and the distributed nodes (units 2 and 3), illustrating both single-candidate and multi-candidate scenarios.

the nighttime data. Therefore, we anticipate that the beam prediction accuracy for images from scenario 41 will not significantly degrade under night conditions. In future work, we are considering collecting a similar dataset for nighttime scenarios and using it to validate this study's findings further.

For the transmitter identification models, we construct separate datasets for each node. The training dataset for the position-aided transmitter identification models consists of pairs of GPS positions and the corresponding center coordinates of the transmitter's bounding box. Similarly, the training dataset for the receive power-aided transmitter identification models includes pairs of receive power vectors and their corresponding bounding box center coordinates. It is important to note there that we do not have access to ground-truth bounding box coordinates for the transmitter vehicle. To address this issue, we manually select samples that contained only the transmitting vehicle. These samples are then processed using the YOLOv7 deep learning model, and the resulting bounding boxes were manually reviewed for accuracy. The bounding boxes generated by YOLOv7 for images containing only the transmitter vehicle are subsequently used as the dataset for training the transmitter identification models. We have 343 and 1124 such samples for unit 2 and unit 3, respectively.

VI. PERFORMANCE EVALUATION

This section focuses on evaluating the performance of the proposed distributed sensing-aided beam prediction solution. In Section VI-A, we provide a description of the experimental setup utilized in this work. We then analyze the results of the proposed solution in Section VI-B.

A. EXPERIMENTAL SETUP

We first outline the neural network training parameters of the machine learning models adopted in this work. Next, we discuss the evaluation metrics which we utilize to assess the performance of different stages of the proposed solution.

Network Training: As described in Section IV, the proposed distributed sensing-aided beam prediction solution consists of three steps: 1) environment semantics extraction 2) transmitter identification and tracking and 3) beam prediction. In the transmitter identification and tracking stage, we use a two-layered fully connected neural network with 512 neurons in each layer to predict the center coordinates of the transmitter's bounding box within the image. For the beam prediction stage, we employ distinct LSTM models for bounding box-based beam prediction and mask-based beam prediction, as elaborated in Section IV-C. In the case of bounding-box based beam prediction, we employ a baseline model consisting of a two-layered FCNN with 512 neurons in each layer. For mask-based beam prediction, we evaluate the LSTM model for it against the LeNet CNN model. In the beam prediction classification task, the LSTM models and their respective baselines are trained using cross entropy loss. On the other hand, the receive power-aided transmitter identification FCNN and its corresponding baseline FCNN are trained using mean squared error loss. In the transmitter identification regression task, both the FCNNs, one taking receive power vector as input and the other taking position as input, are trained using mean squared error loss. We use Adam optimizer to train all the aforementioned models. The detailed hyperparameters used to fine-tune each model are presented in Table 2.

Evaluation Metrics: The evaluation metric used to assess the proposed beam prediction solution is the top-k accuracy, which measures the percentage of test samples where the ground-truth beam falls within the top-k predicted beams. In this work, we present the top-1, top-2 and top-3 accuracies to evaluate the performance of the beam prediction stage. We further assess the performance of the proposed solution using the metric of achievable rate R defined as

$$R = \log_2(1 + \mathrm{SNR}). \tag{14}$$

We use the evaluation metric of comparative accuracy to assess the proposed transmitter identification solution. We define comparative accuracy as the percentage of samples in which the transmitter identified by the receive power-aided FCNN matches the one predicted by the position-aided FCNN. Whereas comparative accuracy is used to evaluate the performance of the proposed transmitter identification solution, the metric of association accuracy is used to evaluate the performance of tracking transmitter across subsequent frames after initial identification. Once the transmitter is identified in the first frame, the association accuracy for each following frame represents the percentage of samples where

TABLE 2. Beam prediction: Design and training hyper-parameters.

ML Model	Mask-LSTM	BBox-LSTM	Mask-LeNet	Bbox-FCNN	Position-Aided FCNN	Receive Power-Aided FCNN
Batch Size	5	8	5	8	50	50
Learning Rate	1×10^{-3}	1×10^{-2}	1×10^{-3}	1×10^{-2}	1×10^{-2}	1×10^{-2}
Learning Rate Decay	-	epoch 20	-	epoch 20	epoch 30 and 70	epoch 30 and 70
LR Reduction Factor	-	0.1	-	0.1	0.1	0.1
Total Training Epochs	50	50	50	50	100	100

the transmitter identified by object association-based tracking (detailed in Section IV-B) matches the transmitter identified by the position-aided FCNN. Note that association accuracy varies across frames and assumes that the transmitter was correctly identified in the first frame, where both the receive power-based and position-based FCNNs identify the same object as the transmitter. For example, an association accuracy of 97% at the second frame indicates that, after the initial identification, the transmitter predicted by object association-based tracking aligns with the position-aided FCNN prediction 97% of the times in the second frame. In computing association accuracy, it is important to note that we do not include the sequences where the difference between the predicted center coordinates by the position-aided FCNN and the center coordinate of the closest bounding box in $\mathbf{X}_{\text{BBox}}$ and $\mathbf{X}_{\text{B-Mask}}$ exceeds a specified threshold.

B. NUMERICAL RESULTS

This section presents a detailed evaluation of the results from the proposed solution.

How accurately can the proposed transmitter identification and tracking solution identify and track the transmitter?

The proposed transmitter identification solution involves using a neural network to predict the center coordinates of the transmitter in the first sample in the sequence using the optimal receive power vector at that time instant. The bounding box and the mask closest to the predicted coordinates is identified as that of the transmitter. The proposed transmitter identification model achieves a comparative accuracy of 82% and 92.42% for unit 2 and 3, respectively. Fig. 9 shows how the association accuracy varies against the sequence length with and without semantic-aided filtering for both unit 2 and unit 3. We observe that the association accuracy for unit 3 decreases only marginally as the sequence length increases and remains above 99% for the whole length of the sequence. However, the association accuracy for unit 2 decreases as the sequence length increases. This may be attributed to the different environmental conditions that the cameras of each unit face. These environmental conditions can include anything from lighting conditions to traffic stoppages.

The semantic-aided filtering method aims to enhance association accuracy by utilizing the physical appearance of vehicles across frames. This approach offers advantages such as improved accuracy in scenarios with partial occlusion of the transmitter vehicle. It further has the potential for enhanced performance when incorporating additional semantic information like vehicle type or color. However, it also has several limitations. The method is sensitive to environmental factors such as lighting conditions and camera angles, which can affect the perceived appearance of vehicles. Maintaining consistent performance across varying conditions is challenging due to the reliance on appearance-based features. Additionally, determining an optimal threshold (ϵ) that works effectively across different scenarios presents difficulties. The efficacy of semantic-aided filtering is contingent on the quality and consistency of the visual data obtained. While it can significantly improve accuracy in favorable conditions, particularly in complex scenarios involving multiple vehicles, its performance may degrade in challenging environmental conditions or when visual distinctions between vehicles are minimal. These factors highlight the trade-offs involved in implementing semantic-aided filtering for vehicle association in diverse real-world environments.

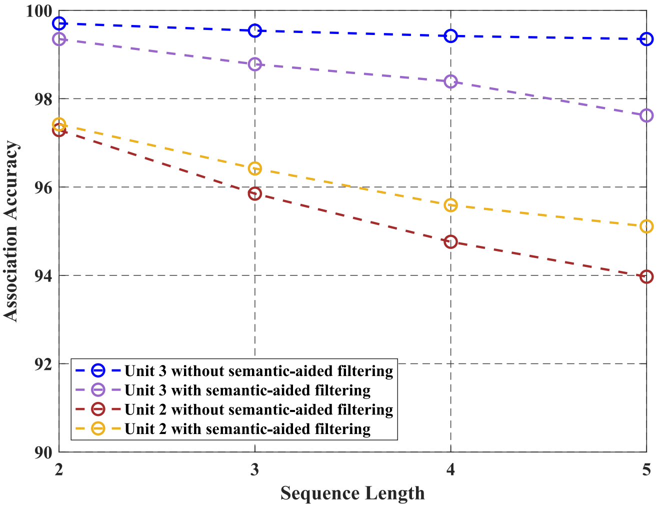
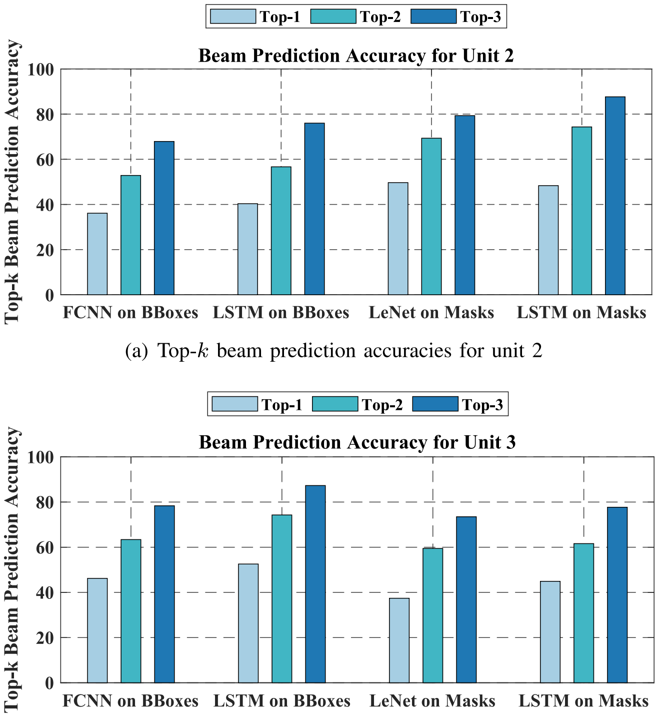


FIGURE 9. The variation of the association accuracy with sequence length for both units 2 and 3 with and without semantic-aided filtering. We observe that semantic-aided filtering has contrasting effects on the association accuracy from both units 2 and 3 across all sequence lengths. Semantic-aided filtering increases the association accuracy for unit 2 while decreasing it for unit 3.

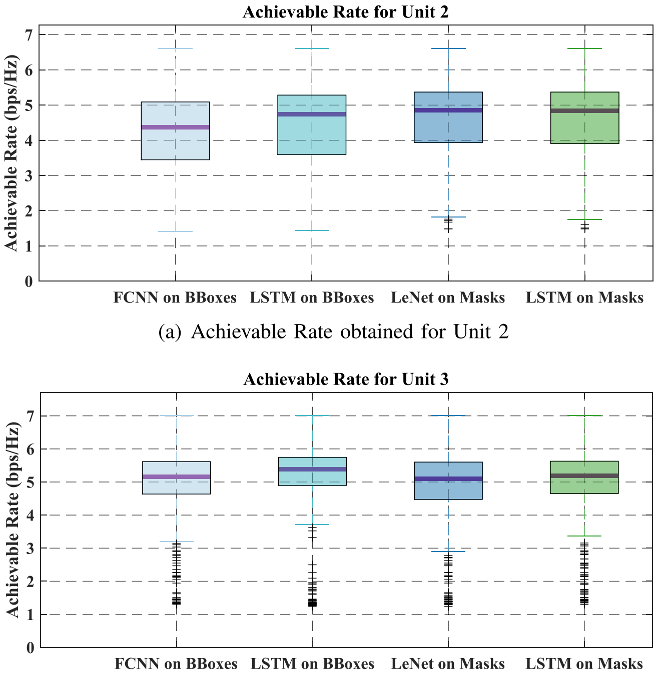
Can the environment semantics extracted from distributed nodes be used for beam prediction at the basestation?



(a) Top-k beam prediction accuracies for unit 2

(b) Top-k beam prediction accuracies for unit 3

FIGURE 10. Beam prediction accuracies of the proposed LSTM models for both units 2 and 3. Overall, the LSTM models, which consider a sequence of environment semantic information as input, achieve better beam prediction accuracy than the solutions that only use the last sample's semantic information as input.



(a) Achievable Rate obtained for Unit 2

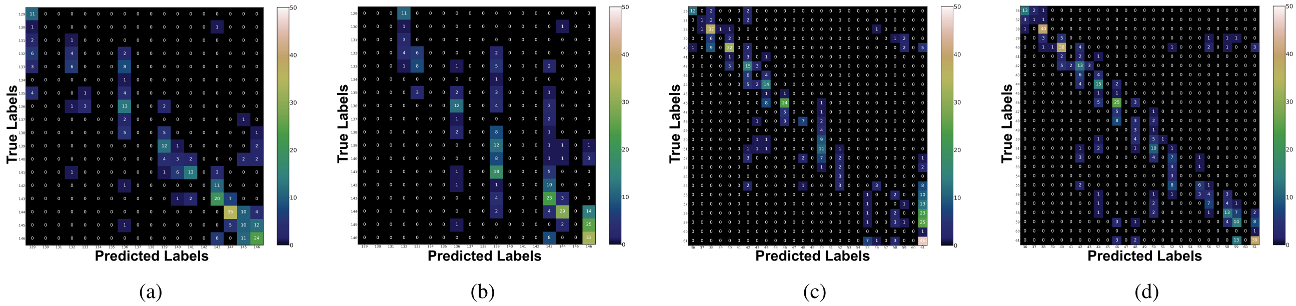
(b) Achievable Rate obtained for Unit 3

FIGURE 11. Achievable rate obtained by the proposed LSTM models for units 2 and 3. Overall, the LSTM models, which process a sequence of environment semantic information as input, outperform solutions that rely solely on the semantic information from the most recent sample in terms of achievable rate.

Fig. 10(a) and 10(b) show the top-1, top-2, and top-3 beam prediction accuracies obtained for units 2 and 3 respectively. We observe that for both units 2 and 3, the LSTM model that takes bounding boxes as input achieves better top-1, top-2, and top-3 beam prediction accuracies than the corresponding FCNN model. On the other hand, the LSTM model that takes masks as input achieves better top-2 and top-3 accuracies for unit 2 and top-1, top-2, and top-3 accuracies for unit 3 compared to the corresponding LeNet model. The top-1 accuracy obtained by the mask-based LSTM model of unit 2 is only marginally less than that obtained by the corresponding LeNet model. We further note that the top-3 accuracy obtained from both the bounding box-based and mask-based LSTM models is more than 75% for both units 2 and 3. This means that using either of the proposed LSTM beam prediction model, the basestation can find the optimal beam in over 75% of instances for both units 2 and 3, thereby significantly reducing the beam training overhead to just three in the case of exhaustive search.

Fig. 11(a) and 11(b) show the achievable rate performance for units 2 and 3. The results show that LSTM models generally achieve higher rates compared to single instance models. An exception is the mask-based LSTM for unit 2, where the rate is only marginally less than the corresponding LeNet model. Additionally, we observe that unit 3 exhibits more outliers in achievable rate performance compared to unit 2 across all machine learning models. However, the median rates for unit 3 are consistently higher than those for unit 2 across all models. The interquartile range of achievable rates for unit 3 is also smaller than that for unit 2 across all models. This difference can be attributed to the disparity in the number of training sequences, with unit 2 having significantly fewer training sequences than unit 3. The improved accuracies and rates obtained using LSTM models can be attributed to their ability to capture better the temporal dependencies in the semantic information, enabling more accurate beam prediction. We observe that the bounding box-based LSTM model performs better for unit 3, while the mask-based LSTM model performs better for unit 2 in terms of beam prediction accuracy and achievable rate. This suggests that specific semantic representations may be more effective in certain regions than in others. For instance, masks can capture the user's shape and orientation, which may be more beneficial for beam prediction in certain regions than in others.

Fig. 12(a) and 12(b) show the confusion matrix plots for unit 2, utilizing masks and bounding boxes as input, respectively. We note that for unit 2, the mask-based LSTM model gives more correct predictions than the bounding box-based LSTM model. Fig. 12(c) and 12(d) show the confusion matrix plots for unit 3 utilizing masks and bounding boxes as input, respectively. We observe that for unit 3, the bounding box-based LSTM model gives more correct predictions than the mask-based LSTM model. We further note that the confusion matrices of the mask-based LSTM model



(a) (b) (c) (d)

FIGURE 12. Fig. (a) and (b) present the confusion matrix plots for unit 2, showing the results obtained from the mask-based LSTM model and bounding box-based LSTM model, respectively. On the other hand, Fig. (c) and (d) present the confusion matrix plots for unit 3, showing the results obtained from the mask-based LSTM model and bounding box-based LSTM model, respectively. The mask-based LSTM model for unit 2 gives more correct predictions than the bounding box-based LSTM model. For unit 3, however, the bounding box-based LSTM model gives more correct predictions than the mask-based LSTM model.

for unit 2 and the bounding box-based LSTM model for unit 3 show a more pronounced concentration of elements near the diagonal. The mask-based and bounding box-based LSTM models demonstrate varying performance for units 2 and 3. This variation is influenced by factors such as camera angle during image capture and the distance between the transmitter and distributed node. Our analysis of the impact of transmitter-node distance on beam prediction accuracy shows that certain semantic representations are more effective in specific regions than in others. It is also important to note that unit 3 has a significantly larger number of training sequences than unit 2, with each unit capturing different conditions in the wireless environment.

Can the proposed sequence-based beam prediction solution meet real-time latency requirements?

Table 1 shows the computational complexity, in terms of time taken (milliseconds), of the deployed machine learning models. The time taken to extract the semantic information of BBox and mask from the image by using YOLOv7 model is about 29.5 millisecond. The semantic information is extracted and transmitted to the basestation before the next image is taken. When the last image in the sequence is captured, the semantic information from previous images is assumed to have already been received at the basestation. We assume that the communication from the distributed node to the basestation is instantaneous. At the basestation, transmitter identification step takes only 0.037 millisecond. An additional 1-2 ms is needed for beam prediction at the basestation. The total time for beam prediction using the mask-based LSTM model at the basestation starting from the point of the last image capture is around 31.5 ms, significantly less than the duration of the timestamp, which is 100 ms. These measurements suggest that our current implementation has the potential to meet real-time requirements. We note that the mask-based LSTM model does not take significantly more time than the mask-based CNN model. Moreover, the bounding box-based LSTM model actually takes less time than the bounding box-based FCNN model. We selected an RNN model for sequence-based beam prediction instead of other neural network architectures such as Transformers because RNNs offer advantages in computational complexity. For example, a Transformer model with the same hidden state size, 8 attention heads, and 6 encoder layers, processing the sequence of masks as input, would take about 3.43 ms to predict the beam, which is more than twice the time taken by the LSTM model. As such, the LSTM model, due to its superior computational complexity, is better suited to manage varying data rates and synchronization demands, justifying its choice for sequence-based beam prediction. It is important to note here that our solution can also be extended to predict future beams, enhancing the proactive nature of the system. By adopting a proactive approach and predicting future beams, we can further overcome the latency issue, ensuring timely and accurate beam prediction in distributed sensing-based mmWave communication systems.

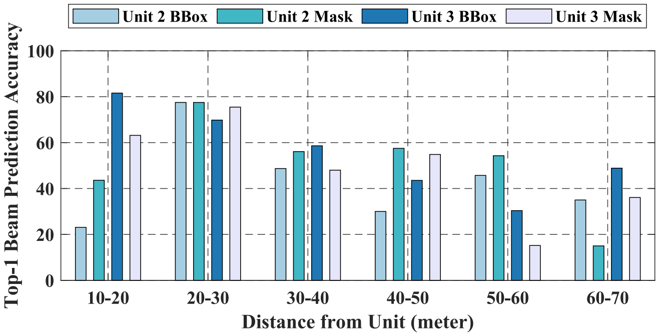


FIGURE 13. Top-1 beam prediction accuracy of LSTM models versus distance from the distributed node for units 2 and 3. We note that the mask-based LSTM model achieves higher beam prediction accuracy for certain distances, while the bounding box-based LSTM model performs better for other distances. This shows that certain semantic representations may be more effective in certain regions than in others.

How does the distance between the transmitter and the distributed node affect beam prediction accuracy?

Fig. 13 illustrates the top-1 beam prediction accuracies plotted against the distance from the distributed node for both units 2 and 3, comparing the performance of the bounding box-based LSTM model and the mask-based LSTM model. We observe that for unit 2, except for the 10-20 meter distance range, the top-1 beam prediction accuracy from the mask-based LSTM model decreases as distance increases.

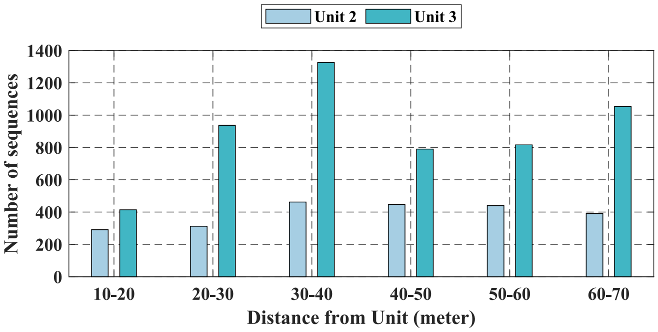


FIGURE 14. Histogram showing the number of sequences falling within various distance ranges for the two distributed nodes. Unit 3 has significantly more sequences compared to unit 2.

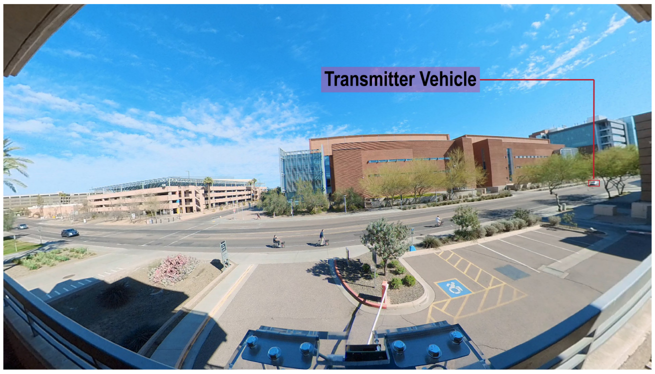


FIGURE 15. The mobile vehicle observed from the basestation at about 10 m from the distributed node of Unit 2. Within the 10-20 m range, some trees partially obstruct the LOS path between the basestation and the mobile vehicle, leading to a degradation in beam prediction for Unit 2 in this range.

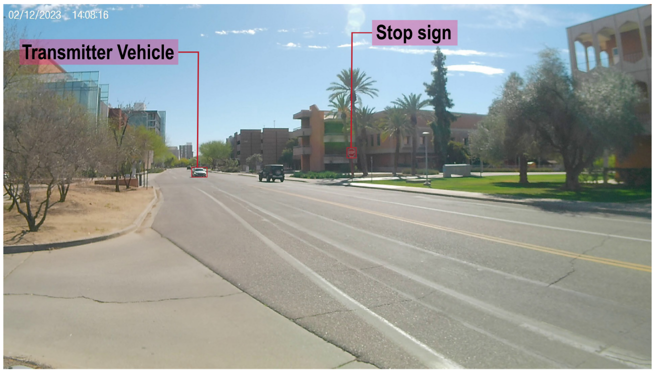


FIGURE 16. The mobile vehicle as observed from the distributed node of Unit 3. The vehicle is approximately 65 meters away from the distributed node. Upon reaching this distance, it stops at a stop sign and then accelerates. This acceleration makes it challenging for the machine learning model to predict the beam at the 5th time stamp.

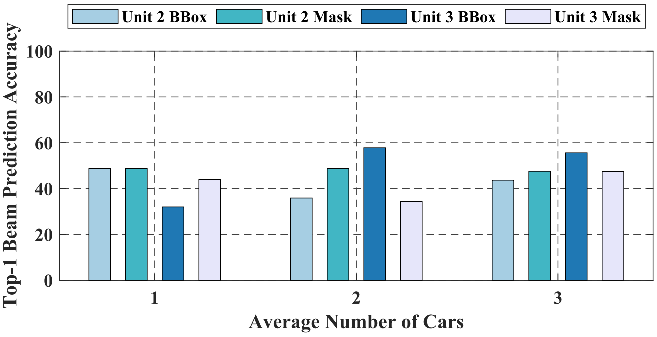


FIGURE 17. Variation of the top-1 beam prediction accuracies of the LSTM models with the number of mobile objects present in the wireless environment for both units 2 and 3. We observe that the beam prediction accuracies remain stable and, in some cases, even improve as the average number of objects in the wireless environment increases.

On the other hand, for unit 3, except for the 60-70 meter distance range, the top-1 beam prediction accuracy from the bounding box-based LSTM model decreases as distance increases. We further note that the top-1 beam prediction accuracies from the bounding box-based LSTM model of unit 2 and the mask-based LSTM model of unit 3 fluctuate across different distance ranges without showing a consistent trend. Furthermore, it is only in the distance ranges of 10-20 meters and 60-70 meters that both the bounding box-based and mask-based LSTM models of either distributed node achieve significantly better beam prediction accuracy over the other. These observations hold true even though a considerably larger number of sequences are available for unit 3 than for unit 2 across all distance ranges, as shown in Fig. 14.

Environmental factors and vehicle dynamics influence the observed accuracy variations. In the 10-20 meter range, both LSTM models of unit 3 achieve better beam prediction accuracy compared to unit 2. Fig. 15 reveals that within this range, trees partially obstruct the line-of-sight path between the base station and the mobile vehicle, leading to degraded beam prediction for unit 2. Conversely, in the 50-60 meter range, both LSTM models of unit 2 outperform those of unit 3. Fig. 16 shows the mobile vehicle at approximately 65 meters from unit 3's distributed node. In this scenario, the vehicle's dynamics–stopping at a stop sign and then accelerating–affect prediction accuracy. Beam prediction at the 5th time instance is more accurate when the vehicle maintains constant velocity or remains stationary, compared to periods of acceleration. The acceleration observed in the 50-60 meter range from unit 3 contributes to reduced prediction accuracy in this region. To address these challenges, particularly in scenarios with variable vehicle velocities, increasing the number of images in the sequence or expanding the training dataset could potentially enhance prediction accuracy.

How does the average number of objects of interest present in the wireless environment affect beam prediction accuracy?

Fig. 17 shows how the top-1 beam prediction accuracies from the LSTM models vary with the average number of objects of interest present in the wireless environment for both units 2 and 3. The average is determined by considering the total number of relevant objects across the five image samples in the sequence. We note that the beam prediction accuracies remain stable and even increase in some instances as the average number of objects

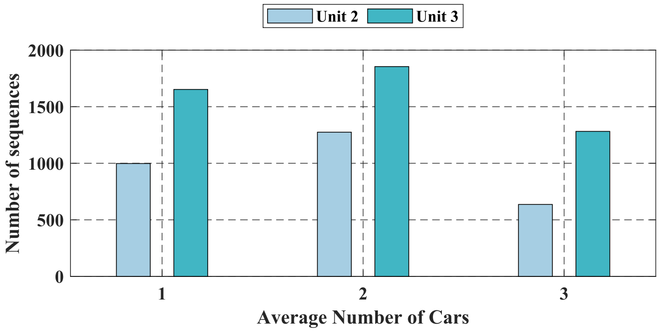


FIGURE 18. Number of image sequences versus the average number of objects of interest present within those sequences. We observe that unit 3 has a considerably larger number of sequences.

in the wireless environment increases. This underscores the efficacy of the proposed transmitter identification and tracking solution and demonstrates the overall effectiveness of the proposed beam prediction solution in a multi-candidate scenario. Fig. 18 depicts the relationship between the number of image sequences and the average number of relevant objects present within those sequences. However, we do not see a proportional increase in beam prediction accuracies for unit 3 compared to unit 2 across the average number of objects categories. One might expect higher beam prediction accuracies for unit 3 compared to unit 2 across various categories of the average number of objects in the wireless environment, as a larger number of sequences typically results in better-trained beam prediction models.

However, it is important to recognize that beam prediction accuracy is influenced by additional factors. These include the camera angle at which the image is captured, the type of environment semantic information transmitted to the basestation, and the distance between the mobile vehicle and the basestation. Our earlier investigation into how the distance between the transmitter and the distributed node affects beam prediction accuracy revealed that certain semantic representations are more effective in specific regions than in others. For instance, when the transmitter vehicle is located 50-60 meters from the distributed node and is correctly identified in a multi-candidate scenario, unit 2 will achieve higher beam prediction accuracy than unit 3, as shown in Fig. 13. This is the case even though more data sequences are available for unit 3 at this distance than for unit 2. Consequently, we observe that unit 3 does not consistently obtain better beam prediction accuracies than unit 2 across various categories of the average number of objects in the wireless environment.

VII. CONCLUSION

This paper presents a distributed sensing-aided beamforming approach. The proposed solution involves deploying multiple distributed nodes, which extract masks and bounding boxes of potential users from raw RGB images. We effectively reduce the storage and transmission requirements by transmitting these semantics to the basestation instead of raw RGB images. We also propose a transmitter identification and tracking solution at the basestation, enabling the proposed solution to operate in a multi-candidate setting. Experimental results on the DeepSense 6G dataset demonstrate the effectiveness of the proposed solution in identifying and tracking the transmitter over multiple frames. The results further show that the proposed solution can predict the optimal beam effectively and demonstrates robustness against both increasing distances from the distributed nodes and a higher number of objects of interest present in the wireless environment. These findings highlight the potential of utilizing environment semantics to facilitate distributed sensing-aided communication. In future work, we plan to utilize sensors other than cameras at the distributed nodes and transmit diverse semantic information to construct a digital twin at the basestation. This digital twin can be utilized to predict future beams, further enhancing the proactive capabilities of the system. By predicting future beams, we aim to address the latency challenges and ensure timely and accurate beam prediction in distributed sensing-based mmWave communication systems.

REFERENCES

[1] 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. 37328–37348, 2018.

[2] T. S. Rappaport et al., “Wireless communications and applications above 100 GHz: Opportunities and challenges for 6G and beyond,” *IEEE Access*, vol. 7, pp. 78729–78757, 2019.

[3] Z. MacHardy, A. Khan, K. Obana, and S. Iwashina, “V2X access technologies: Regulation, research, and remaining challenges,” *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 1858–1877, 3rd Quart., 2018.

[4] S. Hur, T. Kim, D. J. Love, J. V. Krogmeier, T. A. Thomas, and A. Ghosh, “Multilevel millimeter wave beamforming for wireless backhaul,” in *Proc. IEEE GLOBECOM Workshops (GC Wkshps)*, 2011, pp. 253–257.

[5] A. Alkhateeb, O. El Ayach, G. Leus, and R. Heath, “Channel estimation and hybrid precoding for millimeter wave cellular systems,” *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, Oct. 2014.

[6] S. Jayaprakasam, X. Ma, J. W. Choi, and S. Kim, “Robust beam-tracking for mmWave mobile communications,” *IEEE Commun. Lett.*, vol. 21, no. 12, pp. 2654–2657, Dec. 2017.

[7] M. Saquib Khan, Q. Sultan, and Y. Soo Cho, “Position and machine learning-aided beam prediction and selection technique in millimeter-wave cellular system,” in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, 2020, pp. 603–605.

[8] A. Alkhateeb, S. Jiang, and G. Charan, “Real-time digital twins: Vision and research directions for 6G and beyond,” *IEEE Commun. Mag.*, vol. 61, no. 11, pp. 128–134, Nov. 2023.

[9] J. Morais, A. Bchboodi, H. Pezeshki, and A. Alkhateeb, “Position-aided beam prediction in the real world: How useful GPS locations actually are?” in *Proc. IEEE Int. Conf. Commun.*, 2023, pp. 1824–1829.

[10] G. Charan, T. Osman, A. Hredzak, N. Thawdar, and A. Alkhateeb, “Vision-position multi-modal beam prediction using real millimeter wave datasets,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2022, pp. 2727–2731.

[11] G. Charan, M. Alrabeiah, T. Osman, and A. Alkhateeb, “Camera based mmWave beam prediction: Towards multi-candidate real-world scenarios,” 2023, *arXiv:2308.06868*.

[12] S. Jiang, G. Charan, and A. Alkhateeb, “LiDAR aided future beam prediction in real-world millimeter wave V2I communications,” *IEEE Wireless Commun. Lett.*, vol. 12, no. 2, pp. 212–216, Feb. 2023.

[13] U. Demirhan and A. Alkhateeb, “Radar aided 6G beam prediction: Deep learning algorithms and real-world demonstration,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2022, pp. 2655–2660.

[14] U. Demirhan and A. Alkhateeb, “Integrated sensing and communication for 6G: Ten key machine learning roles,” *IEEE Commun. Mag.*, vol. 61, no. 5, pp. 113–119, May 2023.

[15] M. Lötscher, N. Baumann, E. Ghignone, A. Ronco, and M. Magno, “Assessing the robustness of LiDAR, radar and depth cameras against Ill-reflecting surfaces in autonomous vehicles: An experimental study,” 2023, *arXiv:2309.10504*.

[16] J. Park et al., “Communication-efficient and distributed learning over wireless networks: Principles and applications,” *Proc. IEEE*, vol. 109, no. 5, pp. 796–819, May 2021.

[17] W. Xu, Z. Yang, D. W. K. Ng, M. Levorato, Y. C. Eldar, and M. Debbah, “Edge learning for B5G networks with distributed signal processing: Semantic communication, edge computing, and wireless sensing,” *IEEE J. Sel. Topics Signal Process.*, vol. 17, no. 1, pp. 9–39, Jan. 2023.

[18] Y. Yang, F. Gao, X. Tao, G. Liu, and C. Pan, “Environment semantics aided wireless communications: A case study of mmWave beam prediction and blockage prediction,” *IEEE J. Sel. Areas Commun.*, vol. 41, no. 7, pp. 2025–2040, Jul. 2023.

[19] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, “Deep learning enabled semantic communication systems,” *IEEE Trans. Signal Process.*, vol. 69, pp. 2663–2675, Apr. 2021.

[20] Z. Weng and Z. Qin, “Semantic communication systems for speech transmission,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2434–2444, Aug. 2021.

[21] Z. Qin, F. Gao, B. Lin, X. Tao, G. Liu, and C. Pan, “A generalized semantic communication system: From sources to channels,” *IEEE Wireless Commun.*, vol. 30, no. 3, pp. 18–26, Jun. 2023.

[22] S. Imran, G. Charan, and A. Alkhateeb, “Environment semantic aided communication: A real world demonstration for beam prediction,” in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, 2023, pp. 48–53.

[23] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,” 2022, *arXiv:2207.02696*.

[24] S. Niknam, H. S. Dhillon, and J. H. Reed, “Federated learning for wireless communications: Motivation, opportunities, and challenges,” *IEEE Wireless Commun. Mag.*, vol. 58, no. 6, pp. 46–51, Jun. 2020.

[25] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, “Energy efficient federated learning over wireless communication networks,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1935–1949, Mar. 2021.

[26] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, “In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning,” *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, Sep./Oct. 2019.

[27] E. Li, L. Zeng, Z. Zhou, and X. Chen, “Edge AI: On-demand accelerating deep neural network inference via edge computing,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 447–457, Jan. 2020.

[28] A. Alkhateeb et al., “Deepsense 6G: A large-scale real-world multimodal sensing and communication dataset,” *IEEE Commun. Mag.*, vol. 61, no. 9, pp. 122–128, Sep. 2023.

[29] S. Wu, C. Chakrabarti, and A. Alkhateeb, “Proactively predicting dynamic 6G link blockages using LiDAR and in-band signatures,” *IEEE Open J. Commun. Soc.*, vol. 4, pp. 392–412, 2023.

[30] T.-Y. Lin et al., “Microsoft COCO: Common objects in context,” in *Proc. 13th Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[31] W. Min, M. Fan, X. Guo, and Q. Han, “A new approach to track multiple vehicles with the combination of robust detection and two classifiers,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 174–186, Jan. 2018.

[32] L. Zheng, M. Tang, Y. Chen, G. Zhu, J. Wang, and H. Lu, “Improving multiple object tracking with single object tracking,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2021, pp. 2453–2462.

[33] Z. Weng, Y. Zhu, Z. Lin, and H. Li, “Real-time multiple object tracking with discriminative features,” in *Proc. 16th Int. Conf. Control, Autom., Robot. Vis. (ICARCV)*, 2020, pp. 309–314.

[34] G. Charan and A. Alkhateeb, “User identification: A key enabler for multi-user vision-aided communications,” *IEEE Open J. Commun. Soc.*, vol. 5, pp. 472–488, 2024.

[35] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffne, “Gradient-based learning applied to document recognition,” *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[36] K. Cho et al., “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” 2014, *arXiv:1406.1078*.

[37] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, “LSTM: A search space odyssey,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.

[38] A. Vaswani et al., “Attention is all you need,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

SHOAIB IMRAN received the B.S. degree (with Distinction) in electrical engineering from the Lahore University of Management Sciences, Lahore, Pakistan, in 2021. He is currently pursuing the Ph.D. degree in electrical engineering with Arizona State University, Tempe, AZ, USA. He was included on the Dean’s Honor List with LUMS from 2018 to 2021. From July 2021 to November 2022, he worked as a Machine Learning Researcher with the Smart Data Systems and Applications Laboratory, Lahore. His research interests encompass wireless communications, wireless sensing, and machine learning.

GOURANGA CHARAN received the B.Tech. degree in instrumentation engineering from the Indian Institute of Technology Kharagpur, India, in 2015, the M.S. degree in electrical engineering from Arizona State University, USA, in 2021, and the Ph.D. degree in electrical engineering from Arizona State University in 2024. From July 2015 to June 2017, he was an IC Design Engineer with Broadcom Inc., Bengaluru, India. He is currently a Postdoctoral Research Scholar with Arizona State University, Tempe, AZ, USA. He has completed three research internships with Nokia Bell Labs, Murray Hills, NJ, USA; META (formerly Facebook), Redmond, WA, USA; and Apple, Seattle, WA, USA. His current research interests include studying the different applications of deep learning in computer vision, wireless communications, and wireless sensing, with my primary focus on sensing-aided wireless communication.

AHMED ALKHATEEB received the B.S. degree (Distinction with Hons.) and the M.S. degree in electrical engineering from Cairo University, Egypt, in 2008 and 2012, respectively, and the Ph.D. degree in electrical engineering from The University of Texas at Austin, USA, in August 2016. From September 2016 to December 2017, he was a Wireless Communications Researcher with the Connectivity Lab, Facebook, Menlo Park, CA, USA. He joined Arizona State University in Spring 2018, where he is currently an Associate Professor with the School of Electrical, Computer and Energy Engineering. He has held Research and Development Internships with FutureWei Technologies, Chicago, IL, USA, and Samsung Research America, Dallas, TX, USA. His research interests are in the broad areas of wireless communications, communication theory, signal processing, machine learning, and applied math. He is the recipient of the 2012 MCD Fellowship from The University of Texas at Austin, the 2016 IEEE Signal Processing Society Young Author Best Paper Award for his work on hybrid precoding and channel estimation in millimeter wave communication systems, and the 2021 NSF CAREER Award.