IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 26 April 2024, accepted 22 May 2024, date of publication 27 May 2024, date of current version 5 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3405934

RESEARCH ARTICLE

Hydra Radio Access Network (H-RAN): Multi-Functional Communications and Sensing Networks, Initial Access Implementation, Task-1 Approach

RAFID I. ABD[1], (Member, IEEE), DANIEL J. FINDLEY[2], (Senior Member, IEEE), AND KWANG SOON KIM[1], (Senior Member, IEEE)

[1]School of Electrical and Electronic Engineering, Yonsei University, Seoul 03722, South Korea
[2]Department of CCEE, North Carolina State University, Raleigh, NC 27695, USA

Corresponding author: Kwang Soon Kim (ks.kim@yonsei.ac.kr)

This work was partly supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2024-00397216, Development of the Upper-mid Band Extreme massive MIMO(E-MIMO), 50%) and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.2022R1A5A1027646, Augmented Cognition Meta-Communications Research Center, 50%).

ABSTRACT Next-generation sensor and radio access networks (NG-SRANs) namely, Hydra radio access networks (H-RANs) represent a significant evolution in the telecommunications and sensor ecosystem landscape in anticipation of 6G deployment and beyond. H-RAN's vision derives its strength from integrating various technologies and networks into a single central network with the widespread incorporation of artificial intelligence (AI) technologies throughout the network. As a result, H-RAN's unique features and characteristics can serve as a baseline for innovating new applications and significantly enhance the overall functions of conventional open radio access networks (O-RANs). However, among the many improvements and innovations that the H-RAN architecture promises in its functionality, this paper focuses on the initial access implementation "Task_1" approach. Our solution contains several novelties that enhance both overhead and model accuracy. To this end, we define a novel intelligent perception network inspired by the knowledge distribution idea for collaborative H-RAN networks. We develop sparse multi-task learning (SMTL) as part of the AI/ML D-engine for federated learning to perform multiple tasks simultaneously. The SMTL is designed to select the optimal solution from a list of recommended solutions, namely "Tasks". In the simulation, figures of merit include metrics such as top-k validation accuracy, beam selection accuracy, throughput ratios, beam sweep time, latency, and initial access times, which are used to evaluate the performance and efficiency of the proposed technologies. Simulation results demonstrate that by exploiting contextual information from distributed collaborative SRUs, and UE sends its own sensing information via a physical random-access channel in addition to using SMTL, our H-RAN-based initial access scheme can achieve 82.9% throughput of an exhaustive beam search (EBS) based-O-RAN network without any beam search overhead and 96.7% by searching among as few as 5 beams. Compared to the conventional MMW 5G-NR solution, our proposed method significantly minimizes the beam search time needed to reach the desired throughput.

INDEX TERMS
Hydra radio access network (H-RAN), multi-functional networks, perceptive networks, heterogeneous data, AI/ML engines, collaborative-based approach, sparse multi-task learning (SMTL), initial access.

The associate editor coordinating the review of this manuscript and approving it for publication was Maurice J. Khabbaz.

I. INTRODUCTION

As technology evolves and new applications emerge, there is a growing recognition that 5G may not be able to meet all future requirements and demands. The transition to 6G is driven by the anticipation of future use cases and needs that may require even higher data rates, lower latency, greater reliability, and new communication paradigms beyond what 5G can offer. Research communities around the world have already initiated research and development projects focused on exploring 6G's fundamental technologies, concepts, and architectures. These initiatives aim to identify key technology trends, challenges, and opportunities that will shape the future of RAN. Hydra radio access networks (H-RANs) [1] employ sophisticated data processing and analytics techniques, such as edge computing, machine learning, and artificial intelligence, to extract actionable insights from sensor data. H-RANs play a crucial role in laying the foundation for future communication technologies, including 6G and beyond. By embracing a holistic approach to network design, the next generation of sensing and radio access networks (NG-SRANs) will drive innovation in wireless communication, sensor integration, and data-driven intelligence, paving the way for transformative advancements in connectivity, automation, and digitalization. The H-RAN architecture was introduced as a novel paradigm to supplement the state-of-the-art standardization of open radio access networks (O-RANs) [2], [3]. H-RAN is characterized as a comprehensive perceptual RAN in which H-RAN components broadly utilize sensor data and AI/ML engines along with conventional communication parameters to ensure its capabilities evolve agilely over time. H-RAN's vision revolutionizes the conventional 5G gNodeB, by segmenting the single gNodeB into a cluster of perceptive SRUs deployed in a broad network area controlled and managed by a single intelligent H-DU.

Ongoing research and advancements in technologies such as millimeter wave (MMW)/terahertz (THz) communications [4], [5], sensing capabilities [6], and artificial intelligence (AI)/machine learning (ML) algorithms [7] clearly indicate that future wireless networks can be more autonomous, intelligent, and resilient [7], [8], [9]. H-RAN's vision is designed to automatically adjust its parameters and configurations to changing environmental conditions or network requirements. In addition, H-RAN promotes the network's ability to identify and repair faults or disruptions automatically without manual intervention, ensuring continuous service availability and improved fault tolerance. This holistic approach transforms traditional radio access networks into intelligent, context-aware ecosystems capable of meeting the evolving demands of modern RAN.

Initial access is an essential component of MMW/THZ communication systems. These systems often rely on narrow, directional beams to establish and maintain reliable communication links [10], [11], [12], [13]. Nevertheless, achieving accurate beam alignment and beam selection can be challenging and time-consuming due to narrow beams, propagation characteristics, multi-path propagation, mobility environments, and antenna variability [14], [15], [16], [17], [18]. Therefore, efforts to simplify and automate beam alignment and beam selection processes are ongoing across the industry, with advancements in algorithms, hardware, and system integration. These efforts aim to make MMW communication systems more accessible, reliable, and cost-effective, paving the way for their widespread deployment in next-generation wireless networks [19]. However, among the several improvements and innovations promised by our previous study on the H-RAN architecture [1], in this paper, we emphasize initial access procedures, particularly beam alignment and selection methods, namely "Task_1" among the many tasks proposed as H-RAN network promising solutions. Our solution incorporates several novelty features that enhance both the overhead and the model's accuracy. In particular, we propose a novel intelligent perception network that utilizes knowledge distribution for collaborative H-RAN networks. These improvements collectively contribute to the goal of making H-RAN more agile, cost-effective, and capable of meeting the diverse requirements of emerging 6G and beyond. H-RAN specifications are expected to address several MMW/THZ beam management problems and enable novel and exciting applications through perceptive networks. By exploiting the rich information received from different sensors, H-RANs can acquire a comprehensive understanding of the network environment. Moreover, by training sparse multi-task learning (SMTL) models locally and aggregating them collaboratively, SMTL reduces the need for complex computation and communication overhead. SMTL extends the concept of federated learning, which typically focuses on training a single global model across distributed SRUs, to handle multiple related tasks concurrently. SMTL for federated learning enables efficient and scalable learning across multiple tasks in distributed environments, making it well-suited to applications where data is distributed across the edge, IoT, and mobile devices.

A. RELATED WORK

The proliferation of data sources, including sensors, IoT devices, wearables, and web applications, has led to an abundance of data for AI/ML model training and analysis [20]. This wealth of data enables the development of more accurate and robust AI/ML models across a wide range of domains and applications [6], [7]. Advancements in sensor technology have led to the development of smaller, cheaper, and more energy-efficient sensors, expanding their accessibility and applicability [21]. Collaboration between AI/ML researchers, sensor experts, and academic communication engineers from various disciplines has facilitated the development of H-RAN solutions tailored to broad applications and domains. This interdisciplinary approach ensures that AI/ML workflows, sensors, and communication technologies are designed and deployed effectively to address real-world challenges and emulate future aspirations [1]. For example, researchers in

several studies have demonstrated that the integration of context information extracted from sensors with AI/ML algorithms offers a powerful approach to address challenges and enhance various functions in MMW communication systems [22], [23], [24], [25], [26], [27], [28], [29], [30]. The authors of [5] claimed that AI/ML techniques have gained substantial attention for beam management frameworks in MMW/THz bands due to their ability to extract and track nonlinear environmental characteristics. A comprehensive survey of AI/ML-enabled beamforming techniques using out-of-band and multimodal data for MMW communication in next-generation networks is presented in [31]. The study demonstrates that Incorporating (AI/ML) methods within the 5G wireless standard promises autonomous network behavior and ultra-low-latency reconfiguration. Shi et al. [32] showed that integrating high-performance intelligent algorithms (e.g., learning to branch-and-bound graph neural networks for structured optimization, deep reinforcement learning for stochastic optimization, end-to-end learning for semantic optimization, as well as wireless federated learning for distributed optimization) into diverse domains of 6G wireless networks has been an inevitable trend and disruptive shift for supporting highly transparent, reliable, and large-scale 6G communication systems. The article in [33] discusses how the heterogeneity of future services and devices necessitates a highly adaptive and intelligent framework to ensure efficient future network performance. Many innovative services are emerging to accommodate various sensing and IoT services, which extend beyond the traditional provisions of cellular networks and Internet connectivity [20]. To tackle these challenges, researchers are turning to the potential of data-driven methods, specifically AI/ML, and sensors to usher in the next era of intelligent design and decision-making in wireless networks [21]. Reference [34] asserted that the O-RAN paradigm requires several innovations in 6G networks, considering a system-level and architectural perspective. Researchers have confirmed the increasing use of application of AI/ML-based optimization in network slicing, scheduling, and service provisioning, adapting the network to different slices and user needs. Moving forward, deep learning (DL) is adopted to extract the complex dependence in heterogeneous networks between sub-6 GHz and MMW channels for achieving high prediction accuracy for optimal MMW beams in [35]. On the other hand, a location- and orientation-based single and multi-task DNN architecture for the beam selection method to enable context information (CI)-based beam alignment has been proposed in [36]. The study in [37] develops federated multi-task learning (FMTL), for channel estimation with beam-split correction THz channel and user direction-of-arrival (DoA) estimation to improve communications efficiency. The researchers in [38] propose a collaborative service placement-based decentralized algorithm for a network of small cell BSs to optimize service placement decisions collaboratively and address several challenges in mobile edge computing systems. To address the problems mentioned above, cooperative service placement is developed by placing diversified latency critical services to maximize time utility with deadlines and resource constraints [39]. To address the problems mentioned above, cooperative service placement is developed by placing diversified latency critical services to maximize time utility with deadlines and resource constraints [39]. The spatiotemporal graph filter method which integrates graph learning and model-based estimation to achieve multi-view sensor fusion for collaborative object localization is outlined in [40].

B. CONTRIBUTION

The specific contributions of this paper can be summarized as follows:

- We propose an H-RAN initial access approach, namely "Task_1", in which the network incorporates (location, direction, velocity, weather conditions, blockage status, and object distinguishing) as input to AI/ML D-engines. We develop sparse multi-task learning (SMTL) as part of the AI/ML D-engine for federated learning to perform multiple tasks simultaneously. The SMTL is designed to select the optimal solution from a list of recommended solutions, namely "Tasks″, according to online sensing data input and communication parameters. As part of this paper, only "Task_1" will be discussed in the beam recommendation solution for the line-of-sight (LOS) scenario. "Task_1" yields a list of recommended beam pairs that should, later, be sensed by H-DU and UE. This approach results in a drastic reduction in overhead and significantly improved accuracy compared to an exhaustive beam search (EBS) [4].
- We develop multi-functional perceptual networks, which significantly improve beam classification accuracy, primarily in the LOS scenario "Task_1". We propose a curriculum training strategy, which increases convergence speed and final beam prediction accuracy. Our findings indicate that for samples with a dominant LOS component, the strongest propagation path is significantly predictable as it is highly dependent on real-time sensing information, which is largely determined by the surrounding environmental conditions.
- We propose that each UE sends its own sensing information extracted from GPS (e.g., location coordinates, velocity, direction, etc.) via a physical random access channel (PRACH) during the random-access preamble transmission period.
- We designed collaborative sensing signal processing and decision-level fusion approaches through different neural networks for SRU, and H-DU, which are crucial for integrating distributed sensing data among network nodes.
- We propose collaborative service placement through H-RAN-enabled dense SRU networks, in which a single H-DU optimizes service placement decisions collaboratively to address various challenges in SRUs.

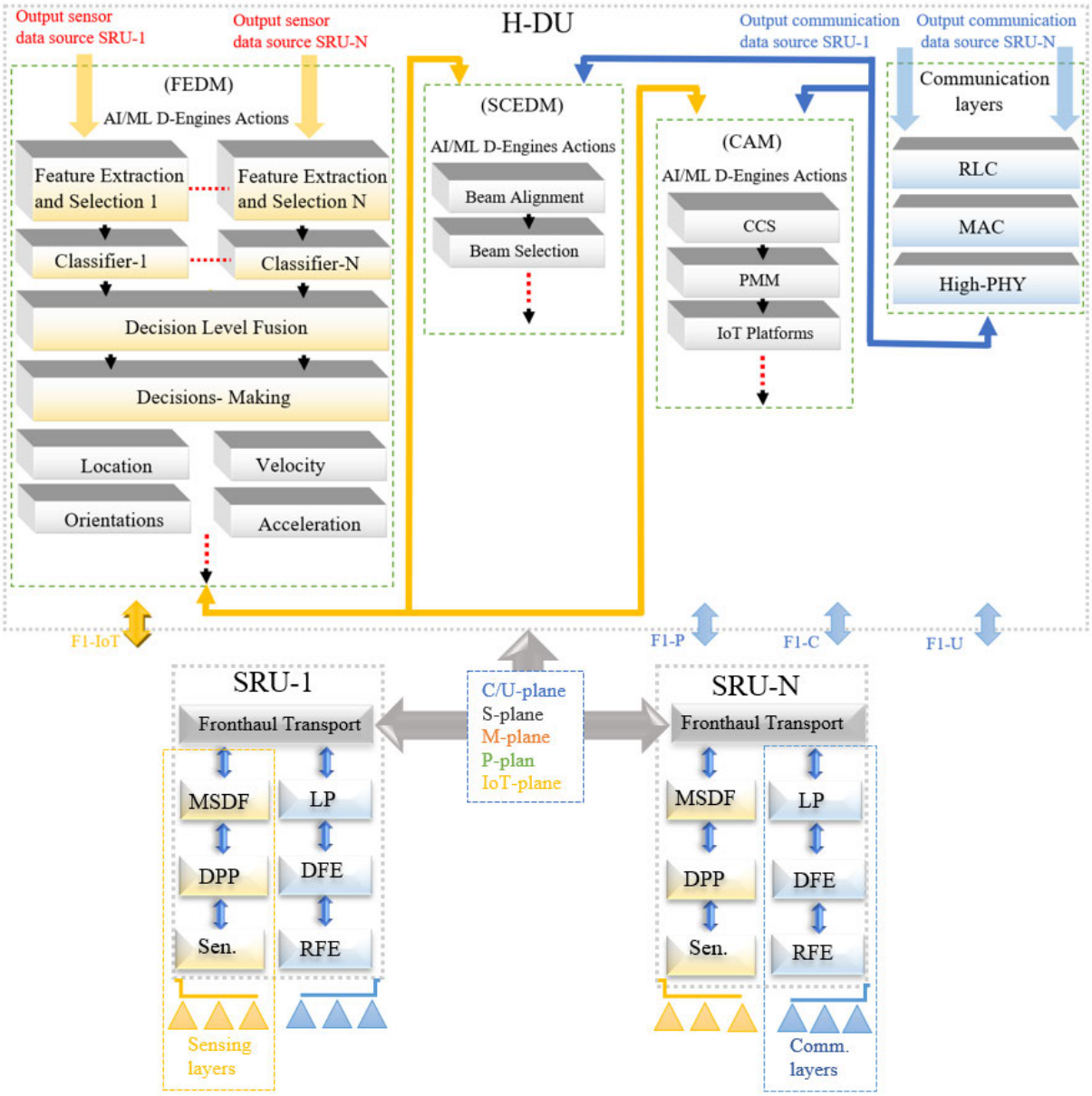


FIGURE 1. The disaggregated architecture of SRUs and H-DU perceptual networks facilitates the deployment of functional units. These networks are augmented with heterogeneous data, sensing, and extensive AI/ML functionality. Fronthaul interfaces and planes enable the split of physical layer functionalities across SRU and H-DU.

- SMTL has been incorporated into the AI/ML D-engine as a baseline approach to serve as an initial benchmark for future work [Task_2, Task_3, Task_4, . . . , Task_n]: aimed at selecting the optimal solution from a list of recommended solutions derived from an online surrounding environment.

II. SYSTEM MODEL

A. H-RAN ARCHITECTURE BACKGROUND

The H-RAN perceptual network is intended to complement the existing O-RAN architecture [1]. To achieve this goal, additional hardware, layers, protocols, algorithms, interfaces, and the widespread use of AI/ML engines are added to supplement each other. We are also pursuing the integration of dual network functionalities through a common infrastructure and building an intelligent network capable of supporting future applications. In this paper, we provide an overview of SRUs and H-DU functionality. We do so by analyzing H-RAN technical specifications, architectural components, and AI/ML functionalities. According to H-RAN specifications, H-RAN disaggregation splits conventional RANs into different functional components, thereby effectively embracing and extending the functional disaggregation paradigm for NG-SRANs. When it comes to control and optimization (e.g., beamforming and beam selection), sensor data, beam parameters, and AI/ML D-engines play a crucial

role in selecting the optimal solution from a list of predefined solutions. Moreover, when it comes to reduced complexity and interoperability challenges, a single multi-functional network by combining communication and sensor functionalities offers numerous benefits that contribute to reducing operational and maintenance expenses and lowering the costs of establishing and managing the network. These cost reductions, coupled with improved efficiency, scalability, and service quality, make multi-functional networks an attractive option for a wide range of applications and industries [1].

B. SENSING AND RADIO UNIT (SRU) ARCHITECTURE

The SRU is a component of the H-RAN architecture, which is designed to be more perceptive, open, interoperable, resilient, and fully collaborative than traditional open radio unit (O-RU) [2] allowing for different components from various vendors and companies to work together such as (telecommunications, autonomous driving, the internet of things (IoT), security, traffic management/alerts, smart applications, etc.) [1]. The protocol stack is split between the SRU, which hosts communication and sensing components, such as the lower part of the physical layer (low PHY), radio frequency (RF), and sensor data pre-processing [1]. SRU is designed to perform simple functions, and interface with the Fronthaul (FH) gateway between the SRUs and H-DUs [1], thereby making it affordable and easy to deploy. As shown in Fig. 1, the H-RAN FH interface protocol comprises several planes, each serving a specific functionality in the communication and management of H-RAN components, including the control plane (C-plane), the user plane (U-plane), the synchronization plane (S-plane), the management plane (M-plane), the perception plane (P-plane), and the internet of things (IoT-plane) [1], [2], [41].

At the sensing layers, to achieve the highest efficiency at the lowest cost, combined data from the most widely used and standard real-world sensors, GPS, MMW radar, and cameras were used to verify SMTL operations and verification. This approach contributes to the improvement of the accuracy and robustness of beamforming and beam selection processes. It is worth mentioning that in the H-RAN philosophy, different types of sensors can be combined with a variety of sensor data processing methodologies according to the network functions to be implemented. For example, sensors such as GPS, radar, cameras, and lidar can play an essential role in improving the initial access and beam management mechanism [22], [23], [24], [25], [26]. In contrast, other types of sensors (e.g., RF sensors, power sensors, environmental sensors, motion sensors, etc.) can be utilized to achieve a comprehensive monitoring and control system, enabling network operators to proactively manage and optimize network functionality [6].

In the SRU, sensor data pre-processing includes feature extraction and early data fusion. Here, deep learning-based features extracted from convolutional neural networks (CNNs) are used to extract relevant features from sensor data, including (e.g., object location, direction, velocity, angles, size, etc.). Also, early fusion is performed by combining features from sensor data at the earliest stage. This results in a single integrated feature vector that represents both modalities. The combined vector is then used as input for classifiers.

C. DATA FUSION

Utilizing prior information obtained from a single sensor (e.g., GPS, radar, lidar, cameras, etc.) may not be sufficient in complex and challenging environments [40]. The reason for this is that single-sensor systems cannot cope with complex environments [43]. GPS provides a wide range of information (e.g., positioning, velocity, synchronization, tracking, mapping) [42]. The GPS signal, however, can be obstructed or weakened by tall buildings, dense foliage, and tunnels. In contrast to cameras and LiDAR, the MMW-radar provides longer wavelengths from 0.1 to 1.0 cm, with a certain amount of anti-pollution and anti-blocking properties, which can handle fog, rain, snow, and low light conditions [24], [25], [26]. MMW radar with a frequency range from 30 to 300 GHz sends multiple consecutive chirps to solve for detecting targets (range, speed, elevation angle, and azimuth angle) as well as tracking multiple targets simultaneously [43]. In contrast, image detection provides more accurate azimuth estimation, classification, and tracking than radar detection [29], [30]. According to the trade-off, multisensory fusion is complementary to each other. A fusion of sensing data offers the ability to take advantage of the information and characteristics of several sensors, thus reducing missed detection rates under poor environmental conditions [35], [40], [43]. Based on the trade-off between accuracy and cost, this approach selects the most widely used sensors. Thanks to combining different data methods, the proposed method is more robust against inaccuracies in sensors to determine the actual location and the potential impact of environmental challenges. Here are some pre-processing steps related to sensor data.

D. RANDOM ACCESS PREAMBLE

In H-RAN deployments, SRUs periodically transmit synchronization signals and system information blocks to facilitate communication with the UE. System information blocks contain critical information about the network, including cell identity, available frequency bands, cell configuration, and other relevant parameters [2]. It is broadcast periodically by the network to accommodate UEs that may enter the coverage area or those already connected to the network. When the UE is powered on, entering a new cell, or has data to transmit, it initiates the random-access procedure. The UE selects a random-access preamble from a predefined set of possibilities. A random preamble is chosen and serves as a unique identifier for the UE during the random-access procedure. The UE transmits the selected random-access preamble over the PRACH to the dedicated channel allocated

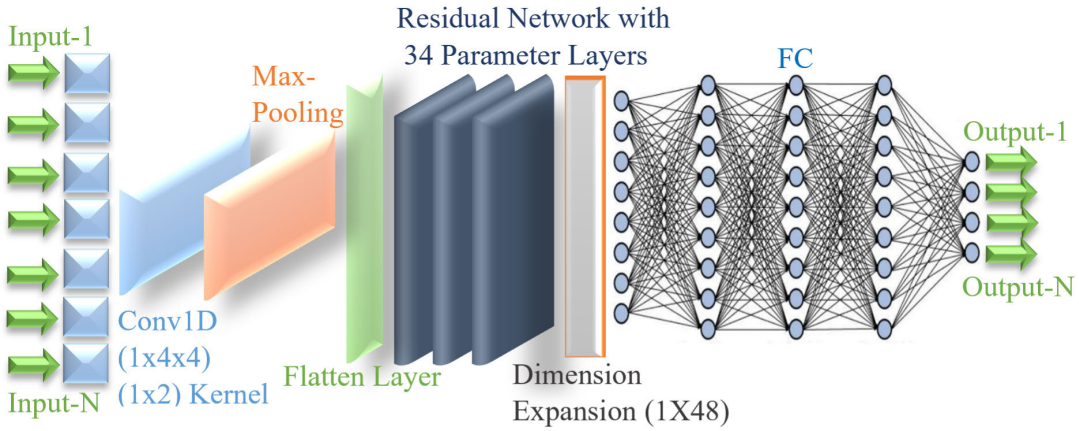


FIGURE 2. AI/ML D-engines, system architecture GPS, and radar mapping transformation.

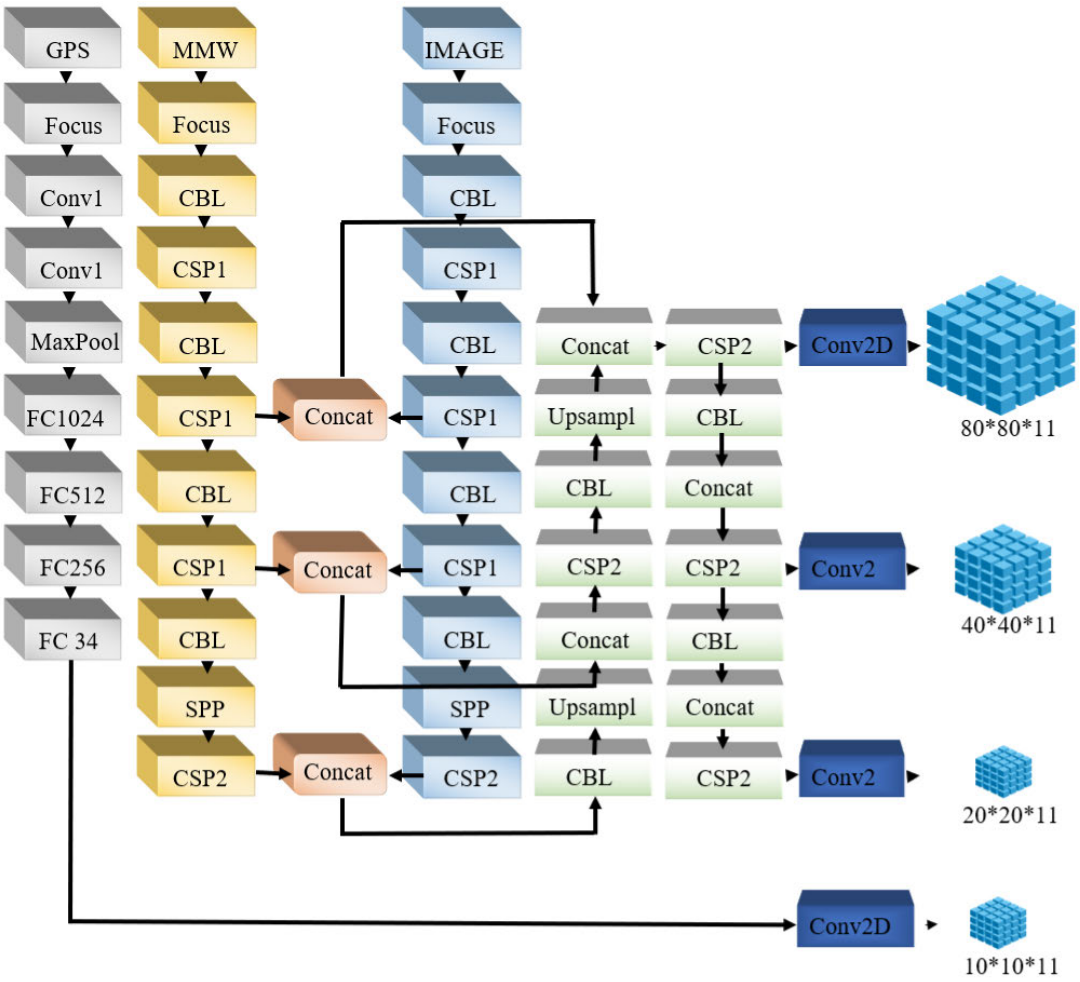


FIGURE 3. AI/ML D-engines, system architecture object detection method.

for handling random access and announces their presence and intention to access the network [41]. We propose that each UE sends its own sensing information extracted from GPS (e.g., location coordinates, velocity, direction, etc.) via a physical random-access channel (PRACH) during the random-access preamble transmission period. Since each preamble is designed to be unique, networks are able to identify the specific UE that sent the preamble. This uniqueness is

essential for the network to respond appropriately to the requesting UE. The initiated communication and the request for network access can be illustrated as follows. First, a UE transmits random access preambles over PRACH, including information regarding the array's location and direction. Each SRU that receives a PRACH signal uses position and direction information in addition to its own sensor information to locate the target and perform an early data fusion stage independently. An SRU successfully detects a target and performs an early data fusion stage, transmitting the information to the associated H-DU. The H-DU selects the optimal SRU to communicate with the UE from among all the SRUs according to the accuracy of detecting the target through the sensor unit. The network responds with a random-access response, assigning a temporary identifier (RA-RNTI) to the UE to identify itself and facilitate channel estimation in subsequent messages.

- **Transformation neural network:** Fig. 2 illustrates the GPS and radar design of the mapping transformation neural network. By parsing the controller area network information generated by the GPS and radar modules, eight types of object state information can be derived [43]. This information represents the network's input layer and includes angle, longitudinal distance, transverse distance, longitudinal velocity, transverse velocity, category, length, and width. A network output needs to identify the bounding box corresponding to the object detected by GPS and radar in the image. Therefore, the output layer of the transformation neural network is set to four, which contains the coordinates of the upper left and lower right corners of the bounding box. As shown in Fig. 2 the mapping transformation neural network consists of a convolution operation, a residual network with 34 parameter layers, and fully connected layers. The convolutional layer has a single input channel, four output channels, and a 1×2 convolution kernel, with a step size of two. By using the residual module [44], the deep-level features can be merged with the shallow-level features. As a result, the deep-level feature map and the shallow-level feature map are superimposed, allowing the number of parameters to be reduced while maintaining high network performance.
- **Early fusion detection** In "Task$_1$", to enhance target detection accuracy, speed, and multi-target tracking in real-time, the multi-source object detection network (MS-YOLO) [43] results from the COCO dataset [45] are applied to enhance target detection accuracy and meet the data transmission delay. MS-YOLO is used for fusing data from multiple sensors, whereby the extracted features of the multiple sensors are combined to implement the object detection model. The MS-YOLO network constructs a double backbone structure based on GPS, radar, and image feature extraction in the early stages, which is then used for later fusion to improve detection accuracy. Fig. 3 depicts a schematic diagram of the MS-YOLO network in which (Focus) is a slicing operation, defined by the width and height of the feature map. Convolution batch normalization Leaky ReLU (CBL) is the standard convolution layer, consisting of two-dimensional convolution, batch normalization, and an activation function, which here uses LeakyReLU. Cross Stage Partial Network (CSP) consists of various bottlenecks and several standard convolution layers. Spatial pyramid pooling (SPP) enables the combination of local and global features. In this network, there are three layers: a middle layer, a backbone layer, and a detection layer. The backbone is an organized combination of several modules, such as Focus, CBL, CSP, and SPP. The key function of the backbone is feature extraction; the middle layer is an orderly combination of CSP, CBL, and upsampling modules; and its primary function is feature fusion. The detection layer consists of a convolution block that receives three groups of features from the middle layer, and in turn generates three groups of detection results, as shown in Fig. 3.

Let's assume the surrounding FoV of the SRU side can be represented by $\mathbf{F_v}[k] \in \mathbb{R}^{H\times W\times\times V}$, where H, W are height and width for V number of views. Therefore, the data and feature level fusion result for a single SRU at a given time k can be estimated as follows

$$\begin{cases} f(x)_1^k = [f_1,\ f_2, \ldots,\ f_m] \\ f(x)_2^k = [f_1,\ f_2, \ldots,\ f_m]\ \ldots \\ f(x)_n^k = [f_1,\ f_2, \ldots,\ f_m] \end{cases} \tag{1}$$

where f_m denotes the feature fusion vector and n is the number of extracted features.

Meanwhile, on the radio side, we consider an uplink massive MIMO system with N_{rx} receiving antennas at the SRU that serve u_{th} users. An uplink transmission is represented by the received signal on the SRU side at a time k

$$\mathbf{x}_s = \sum_{k=1}^{K} \mathbf{H}_k \mathbf{c}_k s_k + \mathbf{n}, \tag{2}$$

where $\mathbf{H}_k \in \mathbb{C}^{N_{\text{SRU}}\times N_{\text{UE}}}$ represents the uplink channel response matrix, $\mathbf{c}_k \in \mathbb{C}^{N_{\text{UE}}\times 1}$ indicates the combined matrix on SRU side, $\mathcal{S}_k = \{s_{k,1},\ s_{k,2},\ \cdots,\ s_{k,|\mathcal{S}_k|}\}$ describes the signal transmitted with the uplink transmit power $\mathbb{E}[||\mathbf{x}_k||^2] \leq P$, $\mathbf{n} \sim \mathcal{CN}(\mathbf{0}_{N_r}, \sigma^2\mathbf{I}_{N_r})$ is the Gaussian additive noise.

III. HYDRA DISTRIBUTED UNIT (H-DU) ARCHITECTURE

The H-RAN paradigm disaggregates the gNodeB functionalities into function entities where the lower layer protocol stack, decision level fusion layers, and AI/ML D-sub-engines are located at the H-DU. In the H-RAN vision, the H-DU is a logical node that hosts various network functions related to sensing, AI, and communication layers. The H-DU plays a crucial role in the disaggregated RAN

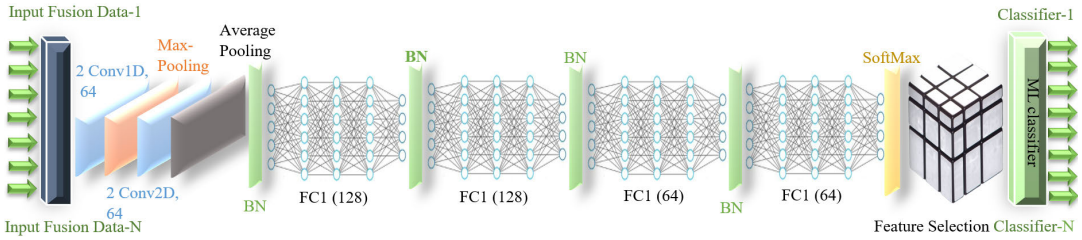


FIGURE 4. AI/ML D-engines, system architecture classifier.

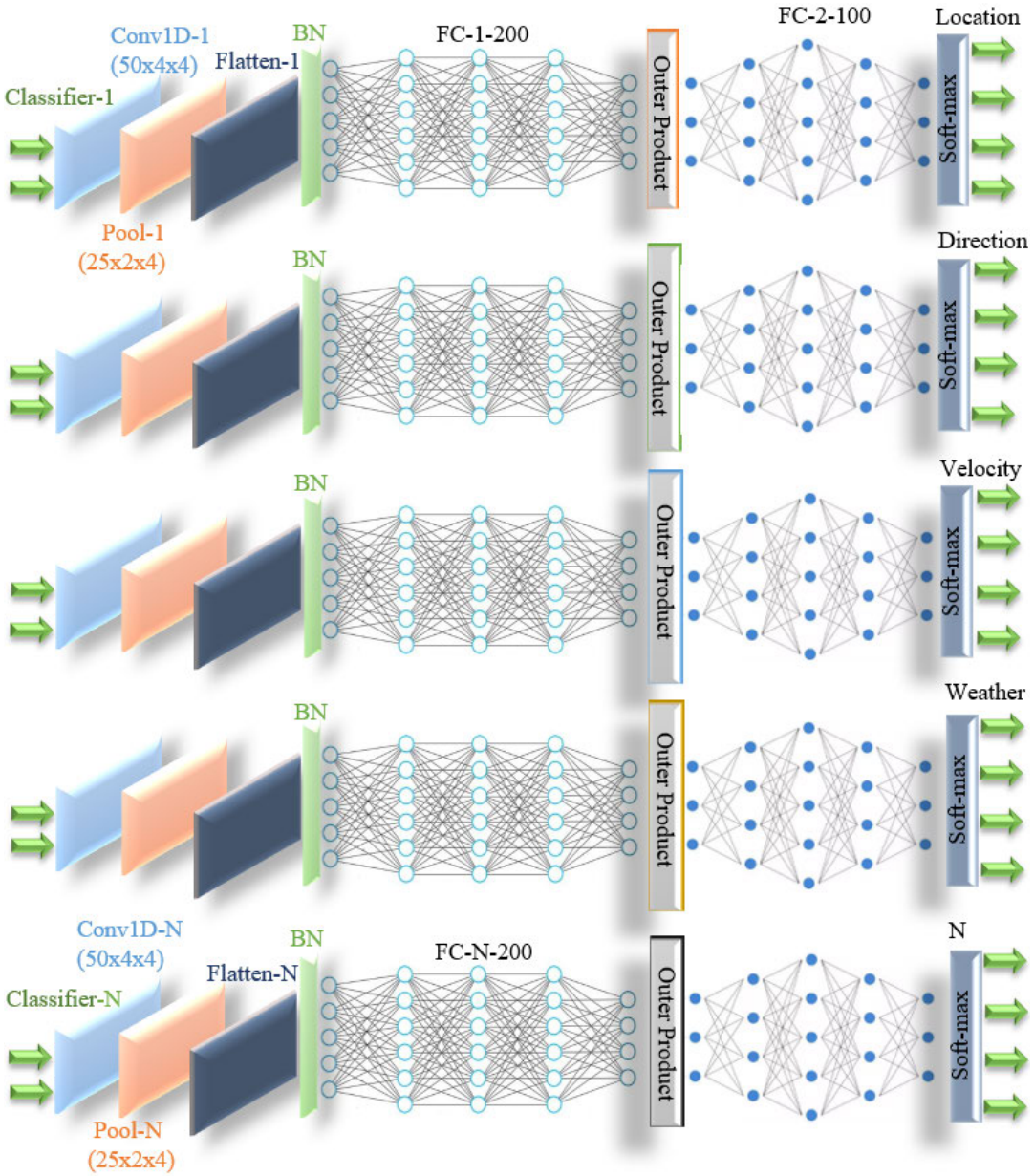


FIGURE 5. AI/ML D-engines, system architecture decision level fusion.

architecture by providing a centralized processing entity for managing and controlling multiple SRU functionalities. A single H-DU can support a cluster of SRUs, (e.g., process numerous cells with one H-DU and various SRUs, coordinate and manage multiple SRUs to support carrier and data aggregation, quality of service (QoS)

management, internet of things (IoT) applications, etc.). As depicted in Fig. 1, the radio link control (RLC) layer, the medium access control (MAC) layer and the physical layer (PHY) are essential layers of the H-DU protocol stack [1], [2]. In contrast, sensing/AI layers include features extraction and decision-making (FEDM), sensing and communication-based AI/ML D-engines decision-making (SCEDM), and control and adaptive mechanisms (CAM) [1]. In the H-DU architecture, preprocessing heterogeneous data involves several steps to provide a means to account for missing values, normalize, scale the data, and transform it into a suitable format for analysis. By utilizing both sensor data and communication parameters, the AI/ML D-sub-engines in the H-DU gain a comprehensive understanding of the network environment and can make data-driven decisions.

According to the H-RAN methodology, a single SRU creates a robust data collection and integration pipeline for managing and analyzing heterogeneous data from various sources [1]. In this model, sensor data comes from GPS, MMW radar, and cameras, providing crucial information related to H-IA (e.g., environmental monitoring, location information, obstacles, user behavior, etc.).

The standardized H-fronthaul interfaces (H-FH) defined by H-RAN vision [1] are used to coordinate the exchange of data and control information between the SRUs and the H-DU. Since the H-RAN components receive heterogeneous data simultaneously. Therefore, portions of the pre-processing for each type of data are required independently by the H-DU components, and then all processed data is collected in AI/ML D-sub-engines for training. In a heterogeneous network, the H-DU protocol stack comprises different layers and features designed to facilitate communication and coordination among diverse elements within the network. The H-DU protocol stack typically integrates components from different technologies, allowing them to work together seamlessly. Here's a brief breakdown of how heterogeneous data is handled in H-DU.

A. FEATURE EXTRACTION AND DECISIONS-MAKING (FEDM)

The FEDM layer is responsible for several functions, including collecting early data fusion sets from a cluster of SRUs, feature extraction and selection, classification, decision-level fusion, and decision-making, which refers to the process of extracting relevant features and information for further analysis. Convolutional neural networks (CNNs) are well-known DL models capable of learning and representing complex features automatically. Therefore, they are applied to a wide range of problems in image classification, text classification, speech recognition, and object detection [20]. As part of this stage, the major objective is to learn meaningful representations from the fused information, thereby optimizing the overall framework's detection and recognition accuracy. The feature selection algorithm uses the selected features to filter the extracted features by choosing the most relevant features that maximize accuracy. Fig. 4 demonstrates the system architecture for feature extraction, selection, and classifier for FEDM layers. The current model is divided into three input groups, a feature extraction group using the CNN model, a feature selection group based on the Aquila optimizer algorithm [20] and a classifier group based on an ML classifier. According to Fig. 4, the CNN structure consists of 2 convolutional layers (Conv), two pooling layers, and four fully connected layers (FC), with four batch normalizations and one Softmax. Conv1 uses a rectified linear unit (ReLU) [45] with 64 filters, a kernel of size three, and a stride of size one. Conv2 incorporates an adaptive average pooling layer. FC1, FC2, and FC3 are fully connected layers having 128, 128, and 64 neurons, respectively. FC1, FC2, and FC3 provide feature extraction layers to output the learned features from the fused data, while FC4 is the final FC layer to output the classification predictions. Four-batch normalization (BN) involves normalizing and stabilizing inputs to a layer within a neural network by adjusting and scaling the inputs.

Here's a general overview of the steps involved: Assuming that the output early fusion of the dataset of the cluster of SRUs $\mathbf{F}^{k}_{ear}(t) = (F_1(t), F_2(t), \ldots, F_k(t))$ are input to the FEDM layer. From a set of SRUs, certain features are extracted from the sensor data streams. Next, feature extraction and selection are used to choose a subset of the most relevant features from the fused dataset while discarding irrelevant or redundant ones. This subset of selected features should provide sufficient information for analysis, classification, and modeling tasks. Assuming that we have f_eth feature sets extracted from f_uth fused dataset measured simultaneously, where each feature set consists of dth samples as follows

$$\begin{cases} f(x)^1_e = [k^1_1,\ k^1_2, \ldots,\ k^1_d] \\ f(x)^2_e = [k^2_1,\ k^2_2, \ldots,\ k^2_d]\ \ldots \\ f(x)^n_e = [k^m_1,\ k^m_2, \ldots,\ k^m_d] \end{cases} \tag{3}$$

where k_d denotes the feature vector and m is the corresponding label. The output of uni-modal feature selections of each sensor modality is used as data representation after passing through the feature extractors.

Fig. 5 demonstrates a network designed for decision-level fusion and decision-making estimation for various sensors of an SRU. Decision-level fusion refers to the process of collecting and aggregating results from multiple classifiers to reach a final decision. As shown in Fig. 5 the classifier results are used as inputs to decision-level fusion. The integrated output of decision-level fusion represents the combined decision or estimate regarding the (e.g., location, velocity, direction, weather condition, etc.). This integrated output is considered more robust and reliable compared with the individual outputs of the classifiers, as it leverages diverse sources of information and mitigates the weaknesses of individual classifiers. At each input of decision-level fusion, convolutional layers apply small filters that slide over the input data, processing small regions at a time. This allows them to capture local patterns and spatial relationships

within the input data. Max pooling is employed for feature down-sampling and spatial abstraction. Flattening refers to the process of converting multi-dimensional input tensors into one-dimensional vectors. FC layers enable end-to-end learning of complex functions directly from input data to output predictions. By stacking multiple fully connected layers together, neural networks can learn hierarchical representations of the input and automatically extract relevant features for the task at hand. Softmax layers are used in classification tasks to predict the class label and category of input data.

B. BEAMFORMING FORMULATION

Creating a pre-defined beamforming codebook $\mathbf{C}$ database involves systematically generating and storing a set of beamforming vectors or matrices that represent different directional configurations. These codebooks are then used to optimize beamforming for communication links based on the specific characteristics of the environment [52]. It should be noted that, by understanding expected communication scenarios, the codebook can be used to select appropriate beam pairs based on real-time channel conditions [36]. However, in MMW communication systems, beamforming performance can be significantly impacted by blockage scenarios, especially in non-line-of-sight (NLoS) links where obstacles obstruct the direct path between the transmitter and receiver [10], [11], [12], [13]. Indeed, traditional beamforming codebooks designed for line-of-sight (LoS) scenarios may not perform optimally under blockage conditions [14], [15], [16], [17]. Therefore, H-RAN vision is designed with adaptive AI/ML strategies to respond to all expected scenarios, including (1) Integrate environmental sensing functionality to detect obstacles and predict blockage scenarios, (2) implement specific beamforming codebooks $\widehat{\mathbf{C}}$ tailored for NLoS scenarios. These codebooks should take into account reflections and diffraction caused by obstacles, (3) develop an adaptive ML to dynamically switch between different codebooks according to perceived channel conditions, (4) develop joint optimization strategies that consider power adjustment, frequency band switching, and SRU switching simultaneously in challenging environments characterized by large obstacles.

In this subsection, we emphasize LoS scenarios and their associated codebook, among all other promising solutions, namely "Tasks" dedicated to the H-RAN solutions. To this end, let us consider a downlink orthogonal frequency-division multiplexing (OFDM) MMW communication system between an SRU and a UE. SRU and UE are equipped with a uniform linear array (ULA) of M_{SRU} antennas, and N_{UE} antennas, respectively. Considering the hybrid beamforming architecture with baseband and RF beamformer at the SRU and M_{RF} RF chains to communicate with $\mathcal{U} = \{1, \ldots, u_n\}$ UEs. A discrete Fourier transform (DFT)-based codebook simplifies beamforming when each transceiver is connected to a single RF chain. The downlink received signal for u_{th} UE can be written as

$$y[u] = \mathbf{H}[u]^T \mathbf{f}[u] s[u] + \sum_{i=u', u' \neq u}^{u} \mathbf{H}[u]^T \mathbf{f}[u'] s[k'] + n[u'], \quad (4)$$

where $\mathbf{H}[u] = \sum l = 1^M \alpha l \mathbf{a}(\phi l, \theta l)$ is the MMW downlink channel matrix at the u_{th} UE, $\mathbf{f}[u] = \mathbf{F}_{\text{RF}}\mathbf{f}_{\text{BB}}$ denotes the u_{th} hybrid beamforming vector, $\mathbf{s}[u] \in \mathbb{C}^{N_s \times 1}$ is the modulated symbols with total average transmitted power $\mathbb{E}[\mathbf{s}[u]\mathbf{s}^*[u]] = \frac{P_c}{uN_s}\mathbf{I}_{N_s}$, and $n[u] \sim \mathcal{CN}(0, \sigma^2)$ refers to additive white Gaussian noise. Consider a set of all possible combinations of precoders and combiners in the transceiver. Context information from sensors can boost the sensing group and reduce the space of sensed precoders and combiners.

The hybrid beamforming matrix $\mathbf{F} \in \mathbb{C}^{M \times U}$ can be formulated as $\mathbf{F} = \mathbf{F}_{\text{RF}}\mathbf{F}_{\text{BB}}$, where and $\mathbf{F}_{\text{RF}} \in \mathbb{C}^{M \times M_{\text{RF}}}$ is the RF beamforming matrix and $\mathbf{F}_{\text{BB}} \in \mathbb{C}^{M_{\text{RF}} \times U}$ is the baseband beamforming matrix. The design sensor-aided beamforming matrix is given based on predicted angles, which can be formulated as

$$\begin{cases} \mathbf{B}_{\text{SRU},i} = [\mathrm{a}_{\text{SRU}}(\theta_i, \phi_i) \ldots \mathbf{a}_{\text{SRU}}(\theta_{N_{\text{SRU},i}}, \phi_{N_{\text{SRU},i}})]. \\ \mathbf{B}_{\text{UE},i} = [\mathbf{a}_{\text{UE}}(\theta_i, \phi_i) \ldots \mathbf{a}_{\text{UE}}(\theta_{N_{\text{UE},i}}, \phi_{N_{\text{UE},i}})], \end{cases} \quad (5)$$

where θ_i and ϕ_i denote the predictive azimuth and elevation angles, respectively

$$\begin{cases} \mathbf{a}_{\text{SRU}}(\theta_i) = [1, e^{\mathrm{j}2\pi\,\Delta\sin(\theta_i)}, \ldots, e^{\mathrm{j}(N_{\text{RSU}}-1)2\pi\,\Delta\sin(\theta_i)}]^{\mathsf{T}}. \\ \mathbf{a}_{\text{V}}(\phi_i) = [1, e^{\mathrm{j}2\pi\,\Delta\sin(\phi_i)}, \ldots, e^{\mathrm{j}(N_{\text{V}}-1)2\pi\,\Delta\sin(\phi_i)}]^{\mathsf{T}}. \end{cases} \quad (6)$$

We assume the MMW massive MIMO channel between the SRU and UE for the discrete-time narrowband channel scattering cluster model can be expressed as [53]

$$\mathbf{H}_u = \mu \times \sum_{i=1}^{N_c} \sum_{k=1}^{N_{ra}} \zeta_{i,k} \alpha_{sr}\left(\phi_{i,k}^{sr}, \theta_{i,k}^{sr}\right) \times \alpha_u^*\left(\phi_{i,k}^{u}, \theta_{i,k}^{u}\right). \quad (7)$$

where $\mu = \sqrt{\frac{N_{tx}N_{rx}}{N_{cl}N_{ra}}}$ denotes the normalization factor based on the total number of scattering cluster N_c and the number of multi-paths within a cluster N_{ra} and the number of multi-paths within a cluster. $\zeta_{i,k}$ refers to the complex path gain associated with the *kth* ray in *ith* clusters. The $\alpha_{sr}(\phi_{i,k}^{sr}, \theta_{i,k}^{sr})$ indicates the angle of arrival (AOAs) at each SRU and $\alpha_u^*(\phi_{i,k}^{u}, \theta_{i,k}^{u})$ is the angle of departure (AODs at each UE. Array response vectors at the transmitter and receiver can be formulated by

$$\begin{cases} \boldsymbol{a}_{sr_{tx}}(\phi) = \xi_{tx}\left[1, e^{j\frac{\eta}{\lambda}\cos\phi}, \cdots, e^{j(N_{sr_{tx}}-1)\frac{\eta}{\lambda}\cos\phi}\right]^T, \\ \boldsymbol{a}_{u_{rx}}(\theta) = \xi_{rx}\left[1, e^{j\frac{\eta}{\lambda}\cos\theta}, \cdots, e^{j(N_{u_{rx}}-1)\frac{\eta}{\lambda}\cos\theta}\right]^T, \end{cases} \quad (8)$$

where ξ_{tx}, and ξ_{tr} are $\frac{1}{\sqrt{N_{srtx}}}$, $\frac{1}{\sqrt{N_{srrx}}}$, respectively, and $\eta = 2\pi d$ is the inter-element spacing.

Hence, we consider a downlink OFDM MMW system, which relies on SRUs to serve UEs in its coverage area. The SRU and the UE are equipped with N_t and M_t antenna arrays, respectively. Assuming fixed beam codebooks.

$\mathcal{C}^{\text{SRU}} = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_i\}$ and $\mathcal{C}^{\text{UE}} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_j\}$, $i = j$ at the transmitter and receiver sides, consecutively.

Assume that $\mathcal{C}^{\text{SRU}}$ is the beam codebook adopted by the SRU, which can be expressed as

$$\mathcal{C}_{\text{M}}{}^{\text{SRU}} = \frac{1}{\sqrt{M}}\left[e^{j\theta_1}, e^{j\theta_2}, \ldots, e^{j\theta_M}\right]^T. \tag{9}$$

where θ_m is the phase shift chosen from a finite set, by considering the pair $(i, j) \in \mathcal{C}^{\text{SRU}} \times \mathcal{C}^{\text{UE}}$ of precoder and combiner vectors.

IV. AI/ML D-ENGINES

A. SPARSE MULTI-TASK LEARNING-BASED AI/ML D-ENGINE

We propose SMTL as a component of the AI/ML D-engine for federated learning [54] to perform multiple tasks simultaneously. Unlike traditional supervised learning, SMTL attempts to learn a function that maps from the multi-input sample space to multi-output spaces, in which each output addresses a specific objective. The proposed SMTL model is designed to perform multiple tasks simultaneously during training. Each task has a specific set of labeled data, and the model learns to perform all tasks jointly. As shown in Fig. 6, the input to the SMTL model is divided into several groups [group_1, group_2, $\text{group}_3, \ldots,$ group_n] and each group represents a specific observation of input features. At each group, the input features for the neural network model are derived from observations collected up to that point in time. This approach is particularly useful when dealing with tasks that involve sequential or time-series data, where observations occur at distinct time points. The SMTL model is trained to perform multiple tasks simultaneously, with each task corresponding to a particular prediction or classification problem. For each group, the model produces task-specific outputs based on the observed input features and the learned representations captured by the network. As depicted in Fig. 6, the SMTL model has a shared layer structure shared by all functions. These shared layers capture features and representations applicable to multiple tasks. This common representation enhances the model's generalization across different domains, as well as decreases the number of network parameters, and the knowledge gained from one task can act as an inductive bias for another. We designed SMTL by employing clustering algorithms to group users with similar environmental characteristics. Predict the optimal solution for each user cluster. Furthermore, we break down the solution into multiple tasks. [Task_1, Task_2, $\text{Task}_3, \ldots,$ Task_n], thereby reducing the complexity of predicting the outcome. Here's a breakdown of the key tasks that represent some of the promising solutions for the H-RAN network.

- **Non-blockage scenario, optimal beam pair selection [$\textbf{Task}_1$] :** In this setting, the SMTL model maps between input user online sensing information [groups] and an optimal beam index from dedicated non-blockage adaptive environment-aware beamforming codebooks, tailored to the LoS scenario [36].
- **Blockage scenario, optimal beam pairs selection [$\textbf{Task}_2$]:** In this setting, and under blockage scenarios caused by small or medium-sized obstructions, the SMTL model maps between input user online sensing information [groups] and an optimal beam index from a dedicated (blockage adaptive environment-aware beamforming codebook). Blockage codebook vectors should be designed to account for reflections, diffractions, and other effects caused by blockages, tailored to the NLoS scenario. It may include beamforming patterns that are more robust to obstructions. These patterns may include wider beams or beams directed at specific incidence or alignment angles to eliminate blockage effects [18].
- **Special scenarios, adaptive strategies [$\textbf{Task}_3$, $\textbf{Task}_4$, $\textbf{Task}_5, \ldots,$ $\textbf{Task}_n$]:** In this setting, if the signal is blocked by a large obstacle or in a scenario of adverse weather conditions such as rain and snow, adaptive strategies may be necessary to implement, such as power adjustment [Task_3], frequency switching [Task_4], SRU switching [Task_5] according to input real-time sensing data, and communication parameters to contribute to high-quality communication links. First, by monitoring online sensing information and RSS, the system can adaptively adjust transmission power to compensate for signal loss. Second, the system can dynamically switch between frequency bands to maintain reliable communication. Lower-frequency bands, such as sub-6 GHz, may be more resilient to attenuation caused by large obstacles or precipitation. Third, the H-DU can dynamically switch between its associated SRUs to ensure reliable communication. This is achieved by referencing the online feedback reports provided by each SRU.

As illustrated in Fig. 6, the SMTL model divides the network into three sub-layers (sparse features generation (SFG) sub-layers, common feature selection (CFS) sub-layer, and task-specific deep feedforward neural network (TSD) sub-layers). First, the sparse feature generation sub-layers (SFG) are designed to reduce the number of trainable weights in neural networks, particularly in layers involving interactions between different sets of features, which is essential for managing model complexity and avoiding overfitting. As shown in Fig. 6, current SFG neural networks sparse the feature generation layers into three groups (position/direction group, velocity/weather condition group, and blockage/object distinguishing group). However, in the SMTL model, a variety of input features [groups] can be incorporated according to the tasks to be performed, which allows for the adaptation and customization of the network architecture to suit specific objectives or requirements.

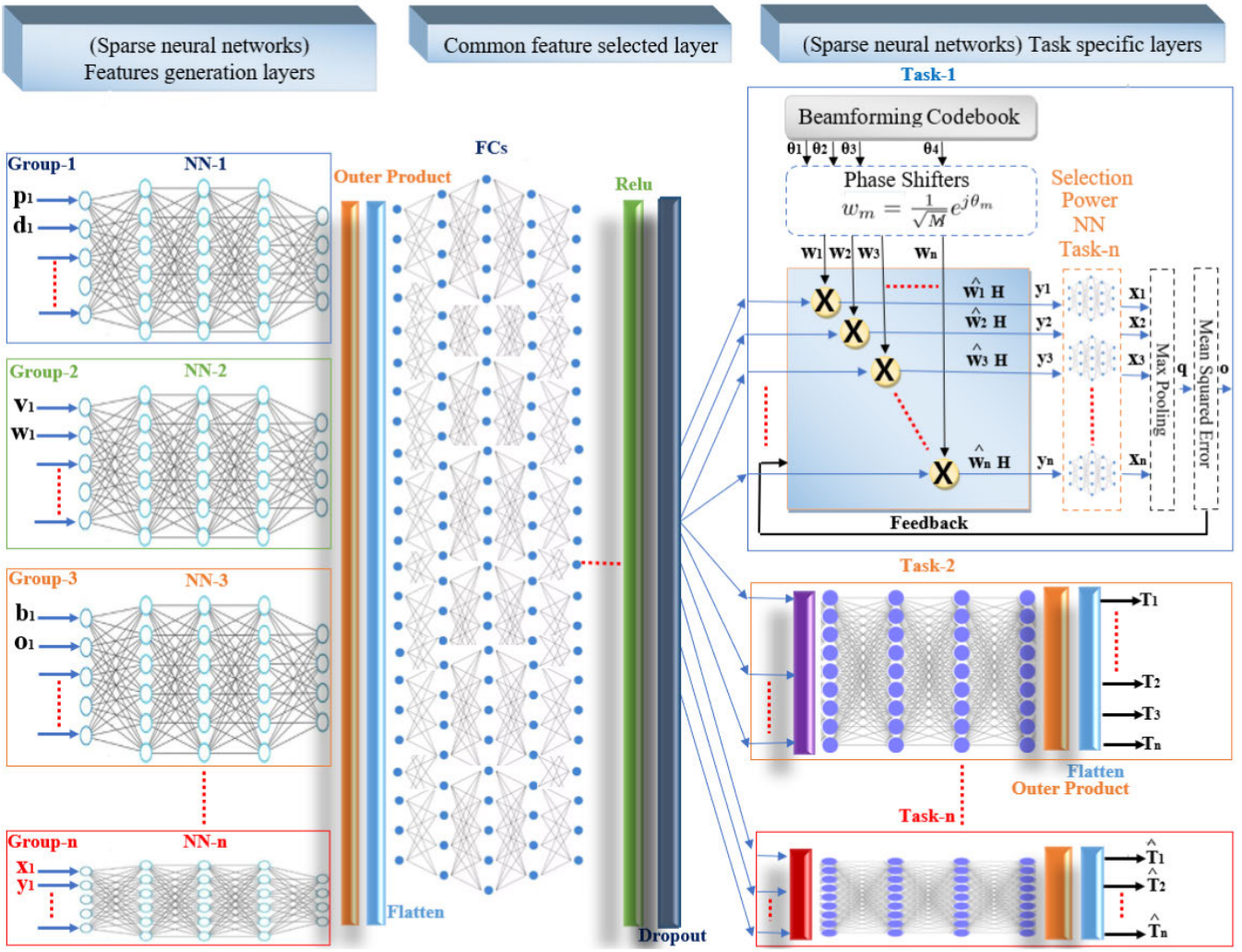


FIGURE 6. AI/ML D-engine system architecture for a sparse multi-task learning (SMTL)-based Task 1 approach.

Sparse neural networks enable the selection of ''task-specific features'' by activating only relevant connections or neurons based on the requirements of the given task. By incorporating features that are most informative or discriminative about the situation at hand, the network can focus its resources more efficiently. For further clarity, through the combination of multiple sets of ''task-specific features'' into a single sparse neural network, multi-task learning can be achieved. This approach allows the network to simultaneously learn and perform multiple solutions, leveraging shared representations and improving generalization across domains of interest. Indeed, SMTL networks are designed to reduce the number of active connections or parameters in the network, making them more efficient. To be more specific, sparsity in the SMTL architecture allows it to activate or deactivate certain connections in response to input features. This can help the network adapt to different combinations of user conditions and reduce the overall number of trainable parameters.

Furthermore, conditional neural networks are used with each sparse feature generation layer, which involves modifying the network's architecture or parameters in response to certain conditions or input features. We designed the SMTL-based AI/ML D-engine with conditional branches and layers that activate in response to specific input features. For example, certain neural network branches will activate only when the current scenario is within a classification (a particular category), allowing the network to specialize in handling this scenario. For greater clarity, for instance, in the (blockage/object distinguish) group, distinguished parameters are triggered only when the blockage scenario is within (a high category) (e.g., blockage by bus, truck, or building, e.g.) enabling the network to specialize in handling obstructed scenarios. We consider the task to be a multi-label classification problem, in which each instance can be associated with multiple classes rather than a single class. This might help capture essential information in a lower-dimensional environment, thereby reducing computation complexity.

It is worth noting that fewer trainable weights reduce computational complexity throughout the training process

and inference. This is crucial for deploying models in resource-constrained environments, such as edge devices or mobile applications, where computational efficiency is a significant consideration. Training a model with fewer trainable weights can lead to faster iterations, allowing for faster experimentation and model development. Therefore, we designed the SMTL with task-specific modifications (classification tasks) to the output layer, which can effectively tailor the neural network to the specific requirements of the problem. For further simplification, we introduced several thresholds to the SMTL model. For instance, velocity thresholds are used to group users into various speed categories (e.g., low, medium, high). The network can then predict the solution for each speed category. Meanwhile, blockage thresholds are used to categorize users into different levels of blockage (e.g., low, medium, high) equivalent to the activation of network output tasks [Task_1, Task_2, $\text{Task}_3, \ldots,$ Task_n], respectively. Thresholds allow us to reduce the number of distinct outputs based on different blockage probability levels. The network predicts the activation of each task according to the threshold level. Furthermore, we group user directions into ranges (e.g., 0-45 degrees, 45-90 degrees, 90-135 degrees etc.). To further reduce the number of trainable parameters, and complexity, the network is designed as a hierarchical structure where [groups] predicts the action, thereby activating or deactivating the specific feature. Moreover, subsequent levels are used to determine the activation of tasks within each category.

Note: The output might come from two or more output tasks simultaneously, depending on the current scenario. To illustrate, for example, in the first scenario, assuming (G_3) confirms that there is no obstruction ($G_3 = 0$), [Task_1] will be initiated to choose the optimal beam pairs through (the non-blockage codebook). Meanwhile, if (G_2) reports unfavorable weather conditions ($G_2 = 1$), [Task_3] is also activated to adjust the transmission power in accordance with the recorded conditions

$$\beta_{i,j,e} = \begin{cases} G_1 = 1, & \text{if } i\text{−th , } \mathbf{T_1} \text{ is selected} \\ G_2 = 1, & 0 \\ G_3 = 0, & \text{if } e\text{−th , } \mathbf{T_3} \text{ is selected.} \end{cases} \tag{10}$$

Here $\mathbf{T_1}$, $\mathbf{T_2}$, and $\mathbf{T_3}$ refer to [Task_1], [Task_2], and [Task_3], respectively. Furthermore, (G_1) represents [group_1] (position/direction) features, (G_2) denotes [group_2] (velocity/weather condition) features, and (G_3) refers to [group_3] (blockage/object distinguishing) features. And (i, j, e) represent the activation parameters of [$\text{Task}_1, \text{Task}_2, \text{Task}_3$], correspondingly.

In the second scenario, if (G_3) indicates the presence of an obstacle ($G_3 = 1$), the procedure for detecting the type and size of the obstacle will be initiated (distinguishing objects). If the detected obstacle falls within the (small or medium) category, then [Task_2] will be activated to select the most suitable beam pairs through (the blockage codebook). At the same time, if (G_2) reports adverse weather conditions ($G_2 = 1$), then [Task_3] must also be activated to adjust the transmission power according to the recorded weather conditions

$$\beta i, j, e = \begin{cases} G1 = 1, & 0 \\ G_2 = 1, & \text{if } j\text{−th , } \mathbf{T_3} \text{ is selected} \\ G_3 = 1, & \text{if } e\text{−th , } \mathbf{T_3} \text{ is selected.} \end{cases} \tag{11}$$

Note: The blockage codebook contains a set of predefined patterns or configurations that describe the possible scenarios of signal blockage in the communication environment. These configurations may include information about the size, shape, location, and material properties of obstacles that can cause blockage, which will be investigated in future works as [Task_2] one of the promising H-RAN solutions.

In the third scenario, given that (G_3) indicates the presence of an obstacle, an obstacle detection procedure will be activated (distinguishing objects). If the detected obstacle is within the (large) categories, then [Task_n] will be activated to select the most suitable solution from a predefined set of solutions (e.g., frequency switching, SRU switching, wide beam transmission, etc.), which will be elaborated extensively in future work. However, at the same time, if (G_2) identifies adverse weather conditions, then [Task_3] will be triggered to adjust the transmission power based on the recorded weather conditions

$$\beta i, j, e = \begin{cases} G_1 = 1, & 0 \\ G_2 = 1, & 0 \\ G_3 = 1, & \text{if } e\text{−th , } \mathbf{T_3} \text{ is selected.} \end{cases} \tag{12}$$

To be more precise, attention mechanisms allow the network to focus on specific parts of the input sequence when making predictions rather than using all the input features at once. This can help the network focus on relevant information for different scenarios. Moreover, to prevent overfitting, the dropout technique [55] is used as a regularization mechanism in SMTL-based AI/ML D-engine, particularly at the hidden layers. Dropout is a form of regularization that involves randomly "dropping out" a fraction of neurons during each training iteration. This means that the output of those neurons is set to zero, effectively removing them from the network for that iteration.

Accordingly, to clarify the proposal more thoroughly, during the first stage, by deploying various sensors, the H-RAN becomes adaptive according to user behavior in real-time, as well as the properties of the surrounding environment. Given a set of observed data $\mathcal{D} := \{(\mathbf{p}_i), (\mathbf{d}_i), (\mathbf{v}_i), (\mathbf{w}_i), (\mathbf{b}_i), (\mathbf{o}_i) : i = 1, \ldots, M\}$, which represent real-time environmental properties (e.g., localization, direction, velocity, weather conditions, blockage status, objects distinguished), respectively. Consisting of a set of input samples $\mathcal{X} := \{\mathbf{x}_i \in \mathbb{R}^p : i = 1, \ldots, M\}$.

As shown in Fig. 6, the detected and predicted features by the first stage of the AI/ML D-engine are used as input to feed the second stage of the AI/ML D-engine. By separating and merging the input features into three groups before feeding them into the second stage, feature input arrays can be

represented as $\mathbf{G}_{[1]}(\mathbf{x}_i, \widehat{\mathbf{x}}_i)^{(k)} \in [\mathbf{p}_{(\mathbf{x}_i)}, \mathbf{d}_{(\widehat{\mathbf{x}}_i)}]^{(k)}$ (position and direction) observations that feed into the first neural network "NN-1", $\mathbf{G}_{[2]}(\mathbf{y}_j, \widehat{\mathbf{y}}_j)^{(k)} \in [\mathbf{v}_{(\mathbf{y}_j)}, \mathbf{w}_{(\widehat{\mathbf{y}}_j)}]^{(k)}$ (velocity, and weather conditions) observations that feed into the second neural network "NN-2", $\mathbf{G}_{[3]}(\mathbf{z}_e, \widehat{\mathbf{z}}_e)^{(k)} \in [\mathbf{b}_{(\mathbf{z}_e)}, \mathbf{o}_{(\widehat{\mathbf{z}}_e)}]^{(k)}$ (blockage status, and objects distinguishing) observations that feed into the third neural network "NN-3", which can be defined as

$$\mathbf{G}_{[1]}(\mathbf{x}_i, \widehat{\mathbf{x}}_i)^{(k)} = \begin{bmatrix} x_i^{(k)}(1) \\ \widehat{x}_i^{(k)}(2) \\ \vdots \\ x_i^{(k)}(n) \end{bmatrix}$$

$$\mathbf{G}_{[2]}(\mathbf{y}_j, \widehat{\mathbf{y}}_j)^{(k)} = \begin{bmatrix} y_j^{(k)}(1) \\ \widehat{y}_j^{(k)}(2) \\ \vdots \\ y_j^{(k)}(n) \end{bmatrix}$$

$$\mathbf{G}_{[3]}(\mathbf{z}_e, \widehat{\mathbf{z}}_e)^{(k)} = \begin{bmatrix} z_e^{(k)}(1) \\ \widehat{z}_e^{(k)}(2) \\ \vdots \\ z_e^{(k)}(n) \end{bmatrix}. \quad (13)$$

Specifically, $(\mathbf{x}_i, \widehat{\mathbf{x}}_i)$, $(\mathbf{y}_j, \widehat{\mathbf{y}}_j)$, and $(\mathbf{z}_e, \widehat{\mathbf{z}}_e)$ refer to the input vector features at the given time (k) into "NN-1", "NN-2", "NN-3", and "NN-n", respectively.

In the second stage, an SMTL is used to learn the mapping between the user's online environmental sensing information and an optimal predefined solution, which is classified as a statistical and probabilistic learning-based scheme. As shown in Fig. 6, the active environmental properties are fed as inputs to the feed-forward, fully interconnected SMTL layers. SMTL predefined solutions are divided into several specific classification tasks, and each task is equipped with q neurons at the output layer. Neurons in an FC layer collectively learn to represent complex features of the input data through the combination of weighted inputs and activation functions. Each neuron specializes in detecting specific patterns and features relevant to the task at hand. For further clarification, for instance, in [Task$_1$], these neurons correspond to all possible combinations of the M_{SRU} and N_{UE} beams at the SRU and UE, respectively. Each group has l_{th} input layers which are responsible for receiving input features, q_h represent hidden layers with n_h neurons at each hidden layer. The non-linear function applied to the output of each neuron in a neural network is known as the activation function. The activation function introduces non-linearity to the model, allowing it to learn complex patterns and relationships from the data. In a neural network, each layer consists of activation functions f_i, weights w_i, and biases b_i. The hyperbolic tangent [tanh] function [56] is used as a non-linear activation function in hidden layers, which can be defined as.

$$\tanh ax = \frac{e^{ax} - e^{-ax}}{e^{ax} + e^{-ax}}. \quad (14)$$

where ax, and $-ax$ represent the value being passed into the function, which represents the weighted sum of inputs from the previous layer, with a bias a. The [tanh] function is applied element-wise to the weighted sum of inputs and biases before being passed to the next layer. The function squashes its input values into the $(-1,1)$ range, which mitigates issues like the vanishing gradient problem. The [tanh] activation function introduces nonlinearity to the network, allowing it to model more complex relationships in the data [36]. The output $(O[i])$ of the i_{th} layer in a neural network with [tanh] activation in the hidden layers can be expressed as [57]

$$O[i] = [\tanh](\sum_{i=1}^{n} \widehat{w} i x_i + b_i), \quad i \in \{1, \ldots, n\}. \quad (15)$$

where [tanh] is the activation function, $\widehat{w}$ denotes the weight matrix, x_i is the input features and b_i is the bias vector.

In the proposed SMTL-based AI/ML D-engine, mini-batch gradient descent [58] is applied to split the entire training dataset into smaller mini-batches. It updates the model parameters only with a mini-batch data set at each iteration. This makes it well-suited to large datasets and computational efficiency, as it allows parallel processing and utilizes matrix operations. During each training iteration, a random subset of neurons in the hidden layers is selected to be dropped out [55]. By doing so, those neurons are temporarily removed from the network (set to zero), and the remaining neurons must learn to compensate for their absence. On the other hand, the outer product [59] is used as a binary operation that takes two vectors as input and produces a matrix as output. Matrix elements are obtained by multiplying each element of one vector by every element of the other vector.

$$a^T \otimes b^T = a^T b = \begin{pmatrix} a_{d1}b_{d1} & \cdots & a_{d1}b_{dk} \\ \vdots & \ddots & \vdots \\ a_{dk}b_{d1} & \cdots & a_{dk}b_{dk} \end{pmatrix}. \quad (16)$$

where two feature vectors $\mathbf{a} = (a_1, a_2, \ldots, a_n)^T$ and $\mathbf{b} = (b_1, b_2, \ldots, b_n)^T$ are the input feature vectors data to the neural networks model non-linear relationships between two feature vectors.

In addition, the Softmax activation function [60] is employed for multi-class classification problems, where the goal is to assign an input sample to one of several classes. Each output of the Softmax function represents the probability of the corresponding class. Note that the number of neurons in the output layer is set to 2, representing the two classes ("optimal (1)" and "not optimal (0)").

In [Task$_3$] case, assuming $\mathbf{F}_i^{(k)}$ is final data fusion, the Softmax activation function is applied to predict the optimality of each beam pair as $\mathbf{P}_o^{(k)} = \text{Softmax}(\mathbf{F}_i^{(k)})$, where SoftMax activation function is defined as

$$\text{Softmax } (\mathbf{x})_i = \frac{e^{x_i}}{\sum_{j=1}^{|\mathcal{P}|} e^{xj}}, i \in \mathcal{P}\{1, \ldots, n\}. \quad (17)$$

where x is codebook elements, and $\mathcal{P}$ indicates a list of beam candidates from a codebook $\mathcal{C}$, in which $\mathcal{P}$-th element

indicates the probability for them being the optimal beam pairs

$$\mathcal{P} = \{(t_n, r_m) | t_n \in \mathcal{C}_{tx}, r_n \in \mathcal{C}_{rx}\}. \tag{18}$$

where $\mathcal{C}_{tx}$ and $\mathcal{C}_{rx}$ are the size codebooks at transmitter and receiver, respectively.

B. TASK-1: NON-BLOCKAGE SCENARIO, OPTIMAL BEAM PAIR SELECTION

In this subsection, we demonstrate only "Task_1" among the several tasks assigned to the SMTL-based AI/ML-D engine, while [Task_2, Task_3, $\text{Task}_4, \ldots,$ Task_n] will be investigated in future research. In this approach, "Task_1" leverages the power of AI/ML D-engines to learn complex relationships between input data and the optimal beam pairs from a predefined non-blockage codebook, allowing the system to adapt to various real-time scenarios in the environment. Accordingly, in "Task_1", only "NN-1" is active which implies that the input to SMTL comes only from "group-1" (position and direction) features, whereas "NN-2" (velocity, and weather conditions) features, and "NN-3" (blockage status, and objects distinguishing) features are in a dropout state. For further clarification, this means that the sensor system in the $\text{SRU}_{(m)}$ which is located at a designated position and at a given time k_{th}, observes that the user $\text{UE}_{(n)}$ is present at the location (x,y), facing the direction (z), and moving at a speed of (0), with suitable weather, and no obstacle blocking the LoS beam, thus the object distinguishing algorithm is ineffective.

The input data from "NN-1" is fed through the trained model using a forward pass. In a neural network, each layer contributes to specific input processing. The output values from the neural network represent the model's predictions or probabilities for each beam pair in the codebook that is most appropriate for the given user information. For the current user scenario, the beam pair with the highest predicted probability is considered the recommended beam pair

$$\boldsymbol{b}_p = \arg \underset{i,j}{\mathbb{C}^{\mathrm{M}\times\mathrm{N}}} (o_{i,j}). \tag{19}$$

The network predicts the top K beams for each direction range, employing a hierarchical approach where the network first predicts a coarse category of beams (e.g., sector-level), and then refines the prediction within that category to select the top K beams. Indeed, static, and mobile scatterers in the communication environment may change signal propagation properties by blocking or reflecting paths [16]. Therefore, H-RAN leveraging online sensing information becomes a highly nonlinear classification problem. Probabilistic and machine learning strategies are common approaches to extracting knowledge about the communication environment and beam patterns from training data [61]. These approaches leverage statistical and computational methods to model and learn patterns, relationships, and probabilistic behaviors from complex nonlinear input-output data. Using pre-defined codebooks simplifies the beamforming process, making it more practical for implementation in real-time communication systems.

The baseband beamforming matrix for requesting and detecting unblocked users is designed as $\mathbf{F}_{\mathrm{BB}} = [\mathbf{f}_1^{\mathrm{BB}}, \mathbf{f}_2^{\mathrm{BB}}, \ldots, \mathbf{f}_K^{\mathrm{BB}}]$, where $\mathbf{f}_k^{\mathrm{BB}} \in \mathbb{R}^{N_{\mathrm{RF}}\times 1}$, and the RF beamforming matrix $\mathbf{F}_{\mathrm{RF}}$ can be expressed as $\mathbf{F}_{\mathrm{RF}} = \mathbf{C}\boldsymbol{\zeta}$ for unblocked status.

Accordingly, we consider the Butler matrix-based hybrid architecture [62] to be used at each SRU, where the total number of antenna elements is set to M, and each M_{RF} RF chain is configured to connect to one of the DFT beam ports.

where $\mathbf{C}$ refers to the DFT codebook matrixs

$$\mathbf{C} = \frac{1}{\sqrt{M}} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & e^{j2\pi\frac{1}{M}} & \cdots & e^{j2\pi\frac{M-1}{M}} \\ \vdots & \vdots & & \vdots \\ 1 & e^{j2\pi\frac{M-1}{M}} & \cdots & e^{j2\pi\frac{(M-1)^2}{M}} \end{bmatrix}. \tag{20}$$

where $\boldsymbol{\zeta} = [\boldsymbol{\zeta}_b]_{b\in\mathcal{J}} \in \mathbb{C}^{M\times M_{\mathrm{RF}}}$ is beam selection metrics.

We used a reinforcement learning framework [63] that learns a codebook of beam patterns optimized to serve users in the particular scenario. The proposed framework autonomously optimizes and allocates codebook beam patterns in response to surrounding environmental properties. The categorical cross-entropy loss function [64] is employed in training neural network models. It measures the dissimilarity between the true distribution of the target classes and the predicted probability distribution produced by the neural network, as follows.

$$\mathcal{L}(\boldsymbol{L}, \widehat{\boldsymbol{L}}) = -\sum_{i=1}^{N_{\mathrm{SRU}}} \sum_{j=1}^{N_{\mathrm{UE}}} L_{i,j} \log(\widehat{\boldsymbol{L}}_{i,j}). \tag{21}$$

where $\boldsymbol{L}$ and $\widehat{\boldsymbol{L}}$ represent the beam pair labels, and the neural network's output respectively.

The optimum beam label is given by $b^* = (i^*, j^*) = \text{argmax}_{(i,j)}\, y_{ij} \subset \mathcal{C}_t \times \mathcal{C}_r$ The output layers are designed to have neurons corresponding to classification classes. On a case-by-case basis, we consider the prediction from the SMTL framework as well as the channel efficiency to properly adjust the beam search space. Thus, the control variable beam search space is not arbitrarily chosen, but tightly coupled to scenario constraints. Assume that the matrix labeled $\mathbb{R}^{N_{\mathrm{RF}}\times 1} \in \{0,1\}^{N_t\times|\mathcal{B}|}$ encodes the vector with binary representation encoding of $\mathcal{B}$ beam pairs, where the optimum beam pair is equivalent to 1 if the transmitted symbol is matched to the RF chains $k_i \in \mathcal{K}_S$, and the rest are 0. Since only one optimal beam is encoded per sample, thus only one class per sample is encoded.

In each task output layer, the Softmax function is used as the activation function to predict the probability of each beam pair being the most optimal, i.e. providing the highest RSS. The Softmax function takes a vector of raw scores (logits) and transforms them into a probability distribution. The output of the last hidden layer, before applying the Softmax function, is a vector of raw scores or logits. Each element of the

vector corresponds to a different beam pair, in which the higher the score, the more likely it is to be the most effective, while each element represents the predicted probability of the corresponding beam pair, such that the probabilities sum to 1

$$f_i(x) = \frac{e^{x_i}}{\sum_{k=1}^{n} e^{x_k}}. \tag{22}$$

The output $\mathbb{O}^k$ of the entire neural network, considering all non-linear functions in hidden layers, can be expressed as

$$\mathbb{O}^k = f_L(w_L \cdot f_{L-1}(w_{L-1}, \ldots, f_1(w_1 \cdot x + b_1) + b_{L-1}) + b_L). \tag{23}$$

here x is the input to the neural network, w_L refers to the weight matrix for the output layer, and b_L corresponds to the bias vector for the output layer.

Since each entry in the predefined codebook represents a specific beamforming direction and configuration. Each example in the dataset consists of sensing information (input) and the corresponding favorable beam pair (output). The neural network architecture is designed for the input layer to have neurons representing the features of the input sensing data, and for the output layer to have neurons corresponding to the number of entries in the codebook

$$\beta_{i,j} = \begin{cases} 1, & \text{if } (i,j) = \arg\max_{M,N} \mathbb{C}_{M,N} \\ 0, & \text{otherwise .} \end{cases} \tag{24}$$

We intend to select a small set ($\mathcal{P}_{Sub} \subseteq \mathcal{P}$) of K beam search candidates. Therefore, we use $((t^*, r^*) \in \mathcal{P}_K)$ to identify the highest beam configuration $((t^*, r^*))$ to maximize the normalized signal power. As a result, we consider a hybrid beamforming scheme with fixed-size codebooks for both transmitter and receiver antenna arrays as follows

$$\begin{cases} \mathbf{A}_{tx} = \{t_1, \ldots, t_N\}. \\ \mathbf{B}_{rx} = \{r_1, \ldots, r_M\}. \end{cases} \tag{25}$$

Assume that N, M are the number of transmitter and receiver codebook components, respectively. Each codebook element specifies the beam direction used by transmitter and receiver antenna arrays for a specific coverage area. As a result, all potential beam pairs $\mathcal{P}$ can be expressed as

$$\mathcal{P} = \{(t_n, r_m) | t_n \in A_{tx}, r_n \in B_{rx}\}. \tag{26}$$

With the assumption that $|\mathcal{P}| = M \times N$, and based on the spatial information coordinates $(t^*, r^*) \in \mathcal{P}_K$ extracted from the fusion of data, the normalized signal power can be calculated as follows

$$f_{(t^*,r^*)} = |w_{t_m}^H \mathbb{H} w_{r_n}|^2. \tag{27}$$

where the weights w_{t_n} and w_{r_m} represent the corresponding beam weight vectors associated with the beam element $|w_{t_m}| \in t_n$ and $|w_{r_n}| \in t_m$, respectively. $\mathbb{H} \in \mathbb{R}^{M \times N}$ indicates the channel matrix concerning the transpose conjugate.

Since optimal pair beam search is restricted to a small set of beam candidates $\mathcal{P}_K$. Hence, maximizing normalized signal power can be formulated as

$$\mathcal{P}_K = \underset{I \subseteq \mathcal{P}, |I| = K}{\arg\max} \mathbf{f}((t^*, r^*) \in I). \tag{28}$$

Accordingly, the throughput between a beam pointing toward optimal beamforming can be calculated as

$$\mathcal{T}\langle_{i,j} = \eta \log 2 \left(1 + \frac{P_{tx} g_{i,j}(\phi)}{Bn[u]}\right). \tag{29}$$

where η refers to an attenuation factor, P_{tx} is the transmit power, $g_{i,j}(\phi)$ denotes the beamforming gain of the beam directed towards an azimuth angle, B corresponds to the bandwidth, and $n[u]$ represents noise vector.

V. COLLABORATIVE-BASED H-RAN

A. COLLABORATIVE-BASED SENSING APPROACH

When distributed sensors independently detect a target or an event, the system can have increased confidence in the information's accuracy. Collaborative signal processing and fusion approaches are crucial for the integration of distributed data among network nodes, as well as for the fusion of multiple modalities of data between sensor nodes, to make accurate and reliable decisions [40]. Collaborative SRUs are orchestrated by the associated H-DU during regular network operations. Accurate target detection and tracking constitute one of the key components of communication applications. Moving targets cause data changes in the time domain, which are detected differently by sensors distributed throughout the H-RAN network. Therefore, collaboration techniques are required both within a sensor node and across nodes to aggregate such various types of data [46]. In the H-RAN vision, collaborating between distributed SRUs allows combining information from a variety of sources to compensate for individual SRU limitations. Effectively integrating information from multiple sensors enhances target detection accuracy and reliability. It also contributes to a more comprehensive understanding of the surrounding environment. Collaboration enables the pooling of information from various SRUs, each with its strengths and weaknesses, thereby facilitating information cross-verification. Assume that the local posteriors formed in SRU nodes are the densities of labeled random finite sets $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_{n_s}$ where n_s denotes the number of sensors in each SRU, and the labels of the same objects in each labeled random finite set are the same. Assume that the FoV of the $s_t h$ sensor by $\text{FoV}_s \in \mathbb{X}$, and its detection probability can be formulated by [46]

$$\begin{aligned} \Pr(X_2^{(\ell)} = \varnothing \,\big|\, X_1^{(\ell)} = \varnothing) \\ = 1 - \frac{r_{k|k-1}^{(\ell)} \left[1 - \langle p_{k|k-1}^{(\ell)}, p_{D_2} \rangle_{(\text{FoV}_2 \setminus \text{FoV}_1)}\right]}{1 - r_{k|k-1}^{(\ell)} \langle p_{k|k-1}^{(\ell)}, p_{D_2} \rangle_{(\text{FoV}_2 \setminus \text{FoV}_1)}}, \end{aligned} \tag{30}$$

where $r_{k|k-1}^{(\ell)}$ denotes the probability of existence at the time $k|k-1$, and $\frac{r_{k|k-1}^{(\ell)}[1 - \langle p_{k|k-1}^{(\ell)}, p_{D_2} \rangle (\text{FoV}_2 \setminus \text{FoV}_1)]}{1 - r_{k|k-1}^{(\ell)} \langle p_{k|k-1}^{(\ell)}, p_{D_2} \rangle (\text{FoV}_2 \setminus \text{FoV}_1)}$ refers to the

probability of the existence of target ℓ at node $s = 2$. If the two fields of view are completely separate, then $\text{FoV}_2 \cap \text{FoV}_1 = \varnothing$

As a result, in scenarios where certain SRU sensors have a limited FoV, cooperative sensor fusion enables the integration of information from multiple sensors. This results in a broader understanding of the environment, and a higher confidence level in detection.

B. COLLABORATIVE-BASED RAN APPROACH

In order to reduce service provision latency and save backhaul network bandwidth, H-RAN-based edge computing relocates some computing functionalities away from the centralized cloud [47], [48]. In this view, we consider collaborative service placement through H-RAN-enabled dense SRU networks, in which a single H-DU optimizes service placement decisions collaboratively [49] to address various challenges in SRUs, such as processing heterogeneous data, spatial demand coupling, and coordination, to name a few. Collaboration-based H-RANs are characterized by their ability to provide decentralized services that are compatible with geographically dispersed SRUs. Consider a dense SRU network of s_{th} SRUs are connected to an HDU that provides communication, sensing, IoT, and edge servers. Each H-DU has a set of subscribed user equipment (UE), which are authorized access services offered by H-DU. Due to the dense deployment of SRUs, a UE may be within the coverage of several other SRUs (though inaccessible) in addition to its home SRU. Therefore, for u_{th} UE, let us assume $\mathcal{U}_m \subseteq \mathcal{U}$ to be the set of reachable communication ranges for SRUs. Thus, $SRUs_{i,j}$ can potentially cooperate to serve common UEs. With this definition, the network can be described as an undirected graph based on the edge between neighbors $\text{SRUs}_{i,j}$ and neighbors hops. According to the UEs' predicted/reported service demand, the H-DU periodically updates their service placement decisions. The H-DU leverages virtualization capabilities to construct execution environments, e.g., virtual machines or containers as part of the service placement process [12]. Once a service application is selected for placement at an H-DU, virtual machines are configured by the application's resource requirements (e.g., RAM, CPU, data storage, I/O, etc.) [50]. However, various services demand different amounts of computational resources, as determined by the service provider. Considering the heterogeneity of data and services and assuming that there are D where $\mathcal{D} = \{1, 2, \ldots, D\}$, and S where $\mathcal{S} = \{1, 2, \ldots, S\}$, types of heterogeneity of data and services, respectively. Without losing generality, we assume a decision-making problem based on a collaborative service placement [21] considering that all SRUs are obedient and fully cooperative. Through collaboration with other SRUs, a UE can offload computation tasks to H-DUs other than its home SRU. Therefore, the placement of all H-DUs should be considered jointly to optimize computing resources. By choosing the optimal collaborative service placement, collaboration SRUs aim to maximize distributed service utility.

C. RESILIENT INFERENCE

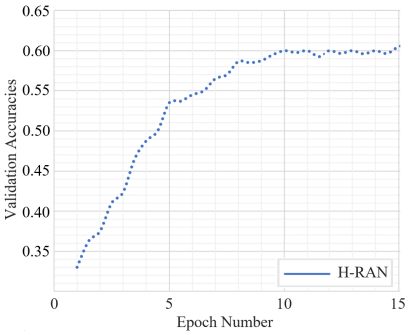
An intelligent NG-SRAN must ensure the accuracy of current information to provide precise information about the target state at any given time. The SRU receives global real-time multimodal data from a variety of sensors. SRUs require inputs from all sensor modalities at any given time to achieve high accuracy. However, even though using a collaborative-based sensing approach [40], it may not be possible to do so at a particular time due to climatic changes, blockages, or hardware problems that prevent data from a particular sensor from being available at that moment [46]. For instance, if a neural network is partitioned and allocated over physical nodes, the failure of physical nodes affects the neural units placed on those nodes, which results in a significant performance drop [39]. Therefore, a multimodal data technical adaptation is developed that compensates for the lack of data from a given sensor. This is done by utilizing copies of earlier data from the same sensor. By using historical information, resilient inference is enabled with minimal performance degradation. Therefore, a pipeline of data adaptation modalities is used when a sensory data type is missing at a particular time. The so-called backtracking AI/ML D-engine retrieves the last available historical data for that sensor for the same scene and applies it to the inference process [51]. Incorporating a diverse set of historical data during the training phase helps neural networks learn from a wide range of scenarios. This diversity can enhance the model's ability to generalize well to new, unseen data during inference. Transfer learning involves using pre-trained models on large datasets for a specific task and fine-tuning them for a target task. Historical information, in the form of pre-trained models, can be transferred to another task, speeding up the training process and potentially improving performance.

VI. NUMERICAL EVALUATIONS AND SIMULATIONS

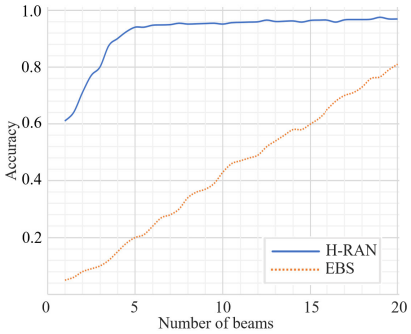
In this section, we demonstrate the performance enhancements and features that each proposed technique and architectural design can offer. Afterward, we evaluated the proposed solutions with each enhancement, compared them against existing 5G NR standards, and demonstrated their superiority. To compare performance, we consider classification accuracy, throughput ratio, and access time. In the simulation scenario, the SMTL attempts to find the optimal beam pair for only the LoS scenario "Task_1". The surrounding environment is accurately observed, and all environmental events are captured by GPS, MMW radar, and camera providing three-dimensional vision.

A. DATASET

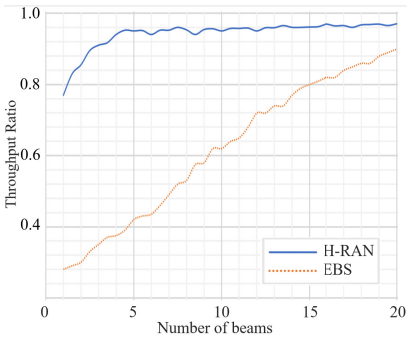
This section introduces the datasets we used to evaluate the H-RAN framework. To leverage AI/ML D-sub-engine vision for the proposed H-DU architecture, it is imperative to have sufficient and appropriate datasets for data analysis. In scenario simulations, data is collected from the same scene



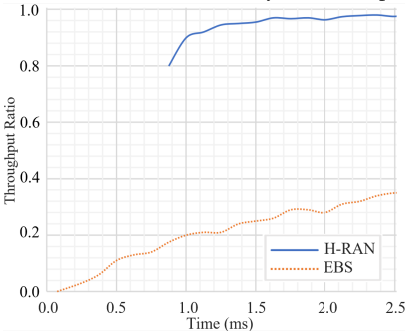
(a) Evolution of the top-k validation accuracy over training epochs.



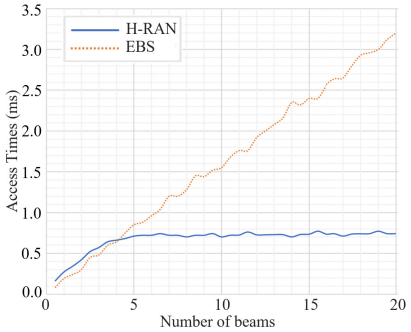
(b) Comparison of beam selection accuracy over sweep beam numbers.



(c) Comparison of throughput ratios over sweep beam numbers.



(d) Comparison of throughput ratios over beam sweep time.



(e) Comparison of initial access times over sweep beam numbers.

FIGURE 7. **Simulation results.**

and captured by GPS, MMW radar, camera, and MMW communication channels.

We consider the outdoor urban environment presented in the ViWi dataset [66] for simulation evaluation. A simulation scenario with multiple wireless users depicts a busy downtown street with its various elements, e.g., cars, buses, trucks, skyscrapers, buildings, lamp posts,...etc. In this scenario, there are two SRUs set at (x,y,z) = (80,14,4.5) and (x,y,z) = (160,−14,4.5). The SRU is set at 5 meters high and in the middle of the street. There are three different orientations of RGB cameras placed at each SRU. Cameras 1, 2, and 3 are installed at SRU 1, and cameras 4, 5, and 6 are mounted at SRU 2. The fields of view of those cameras are overlapping. The side cameras (1, 3, 4, and 6) have a field of view of 75 degrees while the central cameras (2 and 5) have a wider view of 110 degrees. The distance between the two SRUs is 80 meters, and they have a shared camera field of view. More specifically, cameras 3 and 4 view the same street segment. After the intended study environment has been generated, visual data from cameras, radars, and wireless data channel propagation must be collected and analyzed from the same simulation scenes. The wireless InSite ray-tracing software [67] is applied to create a wireless raw dataset and combine the channel quality of different beam pairs.

We consider a downlink orthogonal frequency-division multiplexing (OFDM) MMW communication system between an SRU and a UE. The study evaluates 16,16 and 4,4 uniform planar arrays (UPA) at the SRU and UE, respectively. After constructing the channel response for each UE location, we calculate the RSS for each beam pair using the RSS matrix.

$$R_{i,j} = \left|\sqrt{P_{\text{SRU}}}\boldsymbol{v}_j^H \boldsymbol{H}\boldsymbol{u}_i s + \boldsymbol{v}_j^H \boldsymbol{n}[u]\right|^2. \tag{31}$$

Here $\boldsymbol{u}$ and $\boldsymbol{v}$ denote the precoder and combiner at SRU and UE, respectively. $\boldsymbol{H} \in \mathbb{C}^{N_{\text{UE}} \times N_{\text{SRU}}}$ is the channel matrix. P_{SRUs}, $s \in \mathbb{C}$, and $\boldsymbol{n}$ are the transmission power, the transmitted symbol with unit power, and a complex Gaussian noise vector, in that order.

The NN structures are designed to have $h_l = i \in \{1, \ldots, n\}$ hidden layers with $n_l = j \in \{1, \ldots, n\}$ neurons at each hidden layer. Also, to prevent overfitting in NNs, 5% dropout for all hidden layers is employed. We use Adam optimizer [31] in the training phase with 15 epochs, while the minibatch size is progressively increased from 16, 32, 64, 128, to 7190 samples. To reduce the effects of initial weights of NNs on SRU performance, we averaged results with 7 random weight initializations for each experiment.

In the beam selection methods, the SRU_i sense the environment with all the beam pairs in the candidate list $|\mathcal{S}| = N_b|$ and selects the one that provides the highest RSS for that environment

$$i^*, j^* = \arg\max_{i,j}(R_{i,j}). \tag{32}$$

The SNR of $(i, j)_{th}$ beam pair can be written as:

$$\mathrm{SNR}_{i,j} = \frac{\left\| \sqrt{P_{\mathrm{SRU}}} \boldsymbol{v}_j^H \boldsymbol{H} \boldsymbol{u}_i s \right\|^2}{\sigma_n^2}. \quad (33)$$

B. BEAM SWEEPING LATENCY

The conventional IA time is divided into two-time components, namely the sweep time and the identification of beam pairs time [4]. Indeed, the sweep time dominates the overall IA time, which is relatively time-consuming [17]. Thus, using a subset of beams for IA/ML engines can significantly speed up the process and consume less power [1], [6].

The beam-sweeping latency includes the time required for the gNB to transmit the synchronization signals (SS) bursts with different beam pairs, the UE to measure the signal quality for each beam pair, and the UE to report the measurements back to the gNB. Latency for beam-sweeping depends on factors such as the number of beam pairs to be searched, the duration of each SS burst, and the processing time for measurement and reporting [52]. In the MMW 5G-NR standard-based EBS, during the IA, with the codebook sizes $|\mathcal{C}| = M \times N$, all the M_{Tx} directions are swept with a periodicity of $T_{per} = 20\,ms$. More specifically, the gNB transmits SS bursts sequentially of four OFDM symbols in each codebook element $t_m \in C_{Tx}$ with a swept periodicity of $T_{per} = 20\,ms$, and for an interval of duration, $T_{ssb} = 5\,ms$, which allows for a total of B = 32 SS blocks to be transmitted simultaneously with a different beam pair to be searched within a single SS burst [52]. Meanwhile, on the UE side, the receiver sweeps N_{Rx} directions in searching for SS bursts by receiving different codebook elements $r_n \in C_{Rx}$ for each possible beam configuration [16].

To explore all beam pair combinations, given the codebook sizes of M and N, hence beam pair combinations are represented as $|\mathcal{C}| = M \times N$. Assumedly, each SS block is transmitted simultaneously in two directions using hybrid beamforming, given the 32 SS blocks within a single SS burst. As a result, the 5G NR- MMW gNB can allocate up to 64 synchronization signal SS blocks for each non-overlapping two-direction beam sweep. Therefore, we can express the total time of the possible beam pairs as follows

$$T_{bs}^{\mathcal{C}}(|\mathcal{C}|) = T_{pe} \times \left\lfloor \frac{|M \times N| - 1}{64} \right\rfloor + T_{ssb}. \quad (34)$$

It is worth mentioning that even when using non-overlapping two-direction hybrid beamforming, when optimal beam pair combinations are not explored during the first SS burst ($|\mathcal{C}| > 64$), increasing delays will occur given the transmitted periodicity $T_{per} = 20\,ms$ between SS bursts. Meanwhile, exploring beam pairs smaller than 64 SS blocks during each SS burst or using non-overlapping one-direction beamforming will introduce additional overhead. This is because reducing the number of SS blocks may result in a less accurate alignment, which could lead to longer alignment times or the additional transmission of alignment signals [19]. According to the NR standard, the SS burst duration is set to $T_{ssb} = [5]\,ms$. This allows for a total of 32 SS blocks to be transmitted simultaneously in one direction [65]. Thus, using hybrid beamforming with 64 non-overlapping beams in two directions, the expected time for scanning and measuring a single beam can be written as

$$T_{single} = 5\,ms/64 = 7812\,ns. \quad (35)$$

However, H-RAN architecture is designed to reduce the time of beam search from $|T| = M \times N$ to $T(|\mathcal{F}|) = \mathcal{P}_K$. As a result, in H-RAN networks, the time needed for scanning and measuring the predefined $|\mathcal{F}|$ beam pairs extracted from AI/ML D-sub-engine can be written as

$$T(|\mathcal{F}|) = T_{per} \left\lfloor \frac{|\mathcal{F}| - 1}{64} \right\rfloor + T_{\mathrm{single}}((|\mathcal{F}| - 1) + 1). \quad (36)$$

C. SIMULATION METHODOLOGY

We compare the performance of the proposed model against the state-of-the-art standard for MMW communication called 5G-NR, which is defined based on a beam-sweeping process that sequentially explores all possible directions. In H-RAN simulation a combination of ML, federated learning, and deep learning has been applied through different H-RAN functions, e.g., nonlinear classification problems, fusion of heterogeneous data, computer vision, etc. Federated learning trains a centralized ML model across multiple centralized SRUs by sharing the parameters of the learned local model. This is done while maintaining the raw training data set where it was created. As the scenario considered in this study is based on ViWi, the analysis begins by examining the information provided in the ViWi dataset to determine the origin of the Cartesian coordinate system. We used the ViWi trajectories to identify the locations and then plotted the received signal strength against the UE location of each trajectory considered in ViWi. At each time instance of a scene, ViWi provides raw data for every UE. This includes user location, MMW channel, and link status for each SRU. For each scene, each camera and radar in the SRUs in the network collects data coordinates for a given scene, while considering the areas of overlap between the sensors and the transceiver's transmitted power strength. From which the combined channel quality of different beam pairs is generated by ray-tracing software. We adopted YOLOv5 algorithms for object detection and recognition of video image data. The image data algorithm is pre-trained based on the Coco dataset [45], which contains various types of objects, i.e., cars, buses, trucks, motorcycles, bicycles, pedestrians, etc. We train the proposed deep neural network (DNN) architecture for 20 epochs, where distinct training dynamics lead to a range of convergence times and more precise values. The data obtained by GPS and MMW radar are processed by the algorithm parsing the seven types of state information of the objects, which include location information, length, width, longitudinal distance, transverse distance, longitudinal velocity, transverse velocity, and possible categories. There are several categories of

objects that display obvious characteristics, including people, cars, bicycles, and buses. Data points are formed when GPS and radar data of the object are combined with the pixel coordinates of the bounding box. We calculate the ensemble mean by averaging 1,000 samples from channels with identical distributions. To assess the performance of the SMTL-based AI/ML D-engine, we measure the average of the last 20 iterations out of the total M = 200 iterations. The training dataset for DNNs is divided into 100 percent training and 20 percent testing.

D. RESULTS AND DISCUSSIONS

In this section, we provide the results of our methodology for evaluating the efficiency of the proposed "Task_1" approach to multimodal raytracing. We use five evaluation metrics that capture performance from different perspectives. These metrics include top-M validation accuracy, beam selection accuracy, throughput ratio, beam alignment as a fraction of throughput, and access time. In Fig. 7(a), we plot the evolution of the accuracy metrics, averaging over 5 repetitions of the proposed learning procedures. During convergence time, the H-RAN learning procedures quickly plateaued at higher accuracy levels as illustrated in Fig. 7(a). Network accuracy is evaluated using test data, and our proposed network achieves about 97.2% accuracy, whereas the beam selection network SMTL demonstrates about 90.8% efficiency at Epoch 15. We observed that SMTL becomes more efficient as the number of inputs increases. Therefore, the optimal number of epochs to train decreases as the input layer neurons increase. We can observe that activation functions converge after sufficient data training epochs, resulting in high accuracy levels. The reason for this is that the SMTL is trained to predict the optimal beam direction for a given communication scenario based on input features such as channel state information (CSI), and environmental factors.

In the next evaluation figures-of-merit, our model consistently outperforms the 5G-NR-based EBS scheme in terms of top-M accuracy and throughput ratio metrics, which are reported in Fig. 7 (b), and (c) for the recommended beam candidate $\in [0, 20]$. We observe the gap in accuracy between the two approaches in Fig. 7(b), we can see the stability in the growth of beam selection accuracy in our approach after 5 beams because the best beam was chosen from among the 5 candidate beams. In contrast, we notice that the accuracy of EBS increases gradually with the increase of the beams used until it reaches 20 beams, after which it continues to increase. Nevertheless, even with 20 beams, EBS is still less accurate than the proposed approach.

This result can be attributed to the fact that EBS suffers from accuracy degradation due to several factors, including low antenna gain, insufficient spatial resolution, and an abundance of possible beam directions. In contrast, SMTL can achieve higher accuracy, especially in scenarios with low antenna gain and challenging propagation conditions. By learning patterns and correlations from measurement data, SMTL can make more informed decisions.

In Fig. 7(c), we analyze the impact of the sweep beam candidate's accuracy on the throughput ratio. We observed how the triplet throughput ratio and average selected sweep beam candidate decreased. Intuitively, increasing the accuracy of the selected sweep beam candidate gives more weight to the algorithm to be faster and choose the optimal beam which results in higher QoS. Interestingly, we observe that for beams = 5, the maximum average selected optimal beam equals 92.4%. In this scenario, the objective is to maximize alignment probability.

Our model yields a striking 82.9% throughput ratio, harnessing a significant portion of the available rate without any search procedure. The EBS baseline yields only 44.8% throughput ratios for the 5 beams. At the same time, to ensure a 96.7% expected throughput ratio, our model needs to sweep no more than 5 beams, greatly reducing beam search overhead. As a comparison, the 5G-NR-based EBS scheme requires sweeping more than 20 beams to reach 80%. When it comes to system performance, our model outperforms alternative 5G-NR by providing an average top-1 throughput ratio, improving upon the two baselines by 37.9%. Moreover, our proposed model achieves a tighter confidence interval than the 5G-NR standard for smaller beam candidates. Indeed, this is very beneficial since it ensures more reliable performance guarantees for various instances of training and deployment.

In Fig. 7(d), we compare the above beam selection schemes and report the fraction of throughput obtained as we select the top-recommended beam candidate list versus what is obtained as the EBS sweep progresses. As expected, the throughput ratio increases significantly by moving from a lower resolution to a higher resolution. Overall, the delay caused by sensor data-driven methods and decision-level fusion is very modest, and, at the same time, utilizing sensor information yields high throughput beam pairs with minimal search. For this reason, despite the initial offset due to sensor data side information processing and the transmission of the predicted best beam directions, the proposed method outperforms the EBS method. In this scenario, our proposed approach has almost one-quarter of the search time compared to EBS by achieving an implied throughput ratio of 92.7% at a search time of 1.5 ms, whereas EBS achieves 22. 1% at the same search time. Ultimately, the recommended beam candidate's parameter provides a trade-off between throughput performance, obtained by selecting the optimal beam, and latency, as a larger number of beam candidates increases processing time to search among the candidate options. Thus, our approach offers a means for appropriately determining beam candidates, with the boundary condition representing the selection of the optimal beam. In general, this approach enables the network to adjust the beam pair in response to their specific constraints on establishing ultra-reliable communication and low latency.

Finally, Fig. 7(e) illustrates the evaluation metric with time delay for the initial access procedure as a function of beam search. This is referred to as the duration needed to determine the most effective directions that result in the strongest signal level during the control plane, e.g., when the UE transitions from sleep to active mode, which is defined as

$$T_{\text{acc}} = \text{MT}_{\text{ts}}\text{d}_{\text{tx}}. \tag{37}$$

where M indicates the total number of measured codebooks, T_{ts} denotes the number of time slots occupied during control signals exchanged between the SRU and UE, and d_{tx} refers to the transmitted signal duration reference. Fig. 7(e) plots the aggregated access times for sector prediction and beam association against beamforming (combining) directions. As described in Fig. 7(e), the proposed scheme features a significant reduction in access times compared to existing EBS access methods. Overall, the proposed scheme yields at least a 87.1% reduction in access times by searching among as few as 5 beams, whereas the EBS method involves systematically evaluating each possible beam direction within the scanning space to determine the optimal beam for transmission. As the scanning space expands and more candidate beams are considered, the algorithm must perform additional calculations and measurements for each candidate, which gradually increases the access time. Overall, the reduction in access times achieved through efficient beam selection methods holds substantial promise for enabling ultra-reliable low-latency communications (URLLC) over the air interface in H-RAN. By minimizing the time required for beam selection and configuration, these methods contribute to reducing overall latency and improving the reliability of communication links.

E. CONCLUSION AND FURTHER WORKS

There is no doubt that the massive revolution in MMW/THz systems, sensors, and AI/ML technologies has led to the emergence of a critical role for these technologies in a wide range of modern applications. In addition, the convergence of these technologies opens up opportunities for innovative innovations, applications, and use cases. Integrating a communications and sensor network into a single network with mutual assistance between the two functionalities aims to create a fully perceptible network. Therefore, the compact network can leverage the synergy between the two functionalities. Communication systems can provide connectivity, data transmission, and infrastructure support for sensing devices while sensing capabilities can enhance the perception and awareness of the network by collecting data from the environment. Meanwhile, by harnessing the power of data-driven intelligence, AI/ML can unlock new capabilities, and the ability to automatically adjust network operations in response to changing conditions. Unlike the traditional MMW RAN for selecting beamforming vectors from the LoS codebook, which relies on a limited vision and non-aware system that does not consider continuous changes in the communications environment such as a blockage. The proposed "Task_1" approach is designed with the full perception of the surrounding environment. Therefore, H-RAN continuously adapts to changes in the communication environment and selects the optimal solutions from a list of recommended tasks according to a real-time scenario. In the "Task_1" approach, the optimal beamforming vectors are selected from the LoS codebook according to the real-time environmental scenario. H-RAN networks are designed to provide a comprehensive understanding of the network environment. Furthermore, H-RAN utilizes AI/ML techniques and collaborative approaches to enable intelligent decision-making. To do so, SRUs are placed close to UEs, thus ensuring a fully perceptible network, cooperative-based overlapping FoVs, improving network capacity, reducing power consumption, etc. In this study, the proposed "Task_1" approach introduced several innovations as a component of the H-RAN vision, illustrating how H-IA can be implemented. This approach uses the latent embeddings from each unimodal feature model and observes approximately a 20-25% increase in top-M accuracy, achieves a 90-97% decrease in beam selection time, and reveals a significant reduction in overhead compared to the EBS defined by the 5G-NR standard. It has been demonstrated through simulation results that the Task_1 framework is effective in dynamic scenarios and exhibits robustness against feedback delay and imperfect reciprocity. We intend to address "Task_2'' as part of our future work, where the SMTL model maps between user information on online sensing and an optimal solution from recommended solutions (e.g., dedicated blockage codebooks and adaptive strategies).

REFERENCES

[1] R. I. Abd, D. J. Findley, and K. S. Kim, "Hydra-RAN perceptual networks architecture: Dual-functional communications and sensing networks for 6G and beyond," *IEEE Access*, vol. 12, pp. 2162–2185, 2024.

[2] O-RAN Working Group 1, *O-RAN Architecture Description 5.00*, ORAN.WG1.O-RAN-Architecture-Description-v05.00 Technical Specification, Jul. 2021. [Online]. Available: https://www.o-ran.org/blog/o-ran-allianceintroduces-48-new-specifications-released-since-july-2021

[3] O-RAN Working Group 2, *O-RAN AI/ML Workflow Description and Requirements 1.03*, O-RAN.WG2.AIML-v01.03 Technical Specification, Jul. 2021. [Online]. Available: https://www.o-ran.org/blog/o-ran-alliance-introduces-48-new-specifications-released-since-july-2021

[4] L. Wei, Q. Li, and G. Wu, "Exhaustive, iterative and hybrid initial access techniques in mmWave communications," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2017, pp. 1–6, doi: 10.1109/WCNC.2017.7925666.

[5] M. Qurratulain Khan, A. Gaber, P. Schulz, and G. Fettweis, "Machine learning for millimeter wave and terahertz beam management: A survey and open challenges," *IEEE Access*, vol. 11, pp. 11880–11902, 2023.

[6] F. Liu, Y. Cui, C. Masouros, J. Xu, T. X. Han, Y. C. Eldar, and S. Buzzi, "Integrated sensing and communications: Toward dual-functional wireless networks for 6G and beyond," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 6, pp. 1728–1767, Jun. 2022.

[7] I. Tamim, A. Saci, M. Jammal, and A. Shami, "Downtime-aware O-RAN VNF deployment strategy for optimized self-healing in the O-cloud," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.

[8] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Netw.*, vol. 34, no. 3, pp. 134–142, May 2020.

[9] L. Bonati, S. D'Oro, M. Polese, S. Basagni, and T. Melodia, "Intelligence and learning in O-RAN for data-driven NextG cellular networks," *IEEE Commun. Mag.*, vol. 59, no. 10, pp. 21–27, Oct. 2021.

[10] C. Fiandrino, G. Attanasio, M. Fiore, and J. Widmer, "Toward native explainable and robust AI in 6G networks: Current state, challenges and road ahead," *Comput. Commun.*, vol. 193, pp. 47–52, Sep. 2022.

[11] W. Azariah, F. A. Bimo, C.-W. Lin, R.-G. Cheng, N. Nikaein, and R. Jana, "A survey on open radio access networks: Challenges, research directions, and open source approaches," *Sensors*, vol. 24, no. 3, p. 1038, Feb. 2024.

[12] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges," 2022, *arXiv:2202.01032*.

[13] U. Challita, H. Ryden, and H. Tullberg, "When machine learning meets wireless cellular networks: Deployment, challenges, and applications," *IEEE Commun. Mag.*, vol. 58, no. 6, pp. 12–18, Jun. 2020.

[14] B. Brik, K. Boutiba, and A. Ksentini, "Deep learning for B5G open radio access network: Evolution, survey, case studies, and challenges," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 228–250, 2022.

[15] S. K. Singh, R. Singh, and B. Kumbhani, "The evolution of radio access network towards open-RAN: Challenges and opportunities," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops (WCNCW)*, Apr. 2020, pp. 1–6.

[16] Y. Heng, J. G. Andrews, J. Mo, V. Va, A. Ali, B. L. Ng, and J. C. Zhang, "Six key challenges for beam management in 5.5G and 6G systems," *IEEE Commun. Mag.*, vol. 59, no. 7, pp. 74–79, Jul. 2021.

[17] A. N. Uwaechia and N. M. Mahyuddin, "A comprehensive survey on millimeter wave communications for fifth-generation wireless networks: Feasibility and challenges," *IEEE Access*, vol. 8, pp. 62367–62414, 2020.

[18] R. I. Abd and K. S. Kim, "Continuous steering backups of NLoS-assisted mmWave networks to avoid blocking," in *Proc. 14th Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2023, pp. 11–13.

[19] R. I. Abd and K. S. Kim, "Protocol solutions for IEEE 802.11bd by enhancing IEEE 802.11ad to address common technical challenges associated with mmWave-based V2X," *IEEE Access*, vol. 10, pp. 100646–100664, 2022.

[20] A. Fatani, A. Dahou, M. A. A. Al-qaness, S. Lu, and M. A. Abd Elaziz, "Advanced feature extraction and selection approach using deep learning and Aquila optimizer for IoT intrusion detection system," *Sensors*, vol. 22, no. 1, p. 140, Dec. 2021.

[21] K. Gunasekaran, V. V. Kumar, A. C. Kaladevi, T. R. Mahesh, C. R. Bhat, and K. Venkatesan, "Smart decision-making and communication strategy in industrial Internet of Things," *IEEE Access*, vol. 11, pp. 28222–28235, 2023.

[22] B. Salehi, G. Reus-Muns, D. Roy, Z. Wang, T. Jian, J. Dy, S. Ioannidis, and K. Chowdhury, "Deep learning on multimodal sensor data at the wireless edge for vehicular network," *IEEE Trans. Veh. Technol.*, vol. 71, no. 7, pp. 7639–7655, Jul. 2022.

[23] M. Zecchin, M. B. Mashhadi, M. Jankowski, D. Gündüz, M. Kountouris, and D. Gesbert, "LiDAR and position-aided mmWave beam selection with non-local CNNs and curriculum training," *IEEE Trans. Veh. Technol.*, vol. 71, no. 3, pp. 2979–2990, Mar. 2022.

[24] A. Graff, Y. Chen, N. González-Prelcic, and T. Shimizu, "Deep learning-based link configuration for radar-aided multiuser mmWave vehicle-to-infrastructure communication," *IEEE Trans. Veh. Technol*.

[25] U. Demirhan and A. Alkhateeba, "Radar aided 6G beam prediction: Deep learning algorithms and real-world demonstration," in *Proc. IEEE Trans. Wireless Common. (WCNC)*, Apr. 2022, pp. 10–13.

[26] Y. Chen, A. Graff, N. González-Prelcic, and T. Shimizu, "Radar aided mmWave vehicle-to-infrastructure link configuration using deep learning," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.

[27] M. Alrabeiah, A. Hredzak, and A. Alkhateeb, "Millimeter wave base stations with cameras: Vision-aided beam and blockage prediction," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–5.

[28] M. B. Mashhadi, M. Jankowski, T.-Y. Tung, S. Kobus, and D. Gündüz, "Federated mmWave beam selection utilizing LiDAR data," *IEEE Wireless Commun. Lett.*, vol. 10, no. 10, pp. 2269–2273, Oct. 2021.

[29] W. Xu, F. Gao, X. Tao, J. Zhang, and A. Alkhateeb, "Computer vision aided mmWave beam alignment in V2X communications," *IEEE Trans. Wireless Commun.*, vol. 22, no. 4, pp. 2699–2714, Apr. 2023.

[30] T. Zhang, J. Liu, and F. Gao, "Vision aided beam tracking and frequency handoff for mmWave communications," in *Proc. IEEE INFOCOM Conf. Comput. Commun. Workshops*, May 2022, pp. 1–2.

[31] D. Roy, B. Salehi, S. Banou, S. Mohanti, G. Reus-Muns, M. Belgiovine, P. Ganesh, C. Dick, and K. Chowdhury, "Going beyond RF: A survey on how AI-enabled multimodal beamforming will shape the NextG standard," *Comput. Netw.*, vol. 228, Jun. 2023, Art. no. 109729, doi: 10.1016/j.comnet.2023.109729.

[32] Y. Shi, L. Lian, Y. Shi, Z. Wang, Y. Zhou, L. Fu, L. Bai, J. Zhang, and W. Zhang, "Machine learning for large-scale optimization in 6G wireless networks," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 4, pp. 2088–2132, Aug. 2023.

[33] C. Fischione, M. Chafii, Y. Deng, and M. Erol-Kantarci, "Data sets for machine learning in wireless communications and networks," *IEEE Commun. Mag.*, vol. 61, no. 9, pp. 80–81, Sep. 2023.

[34] M. Polese, M. Dohler, F. Dressler, M. Erol-Kantarci, R. Jana, R. Knopp, and T. Melodia, "Empowering the 6G cellular architecture with open RAN," *IEEE J. Sel. Areas Commun.*, vol. 42, no. 2, pp. 245–262, Feb. 2024.

[35] K. Ma, S. Du, H. Zou, W. Tian, Z. Wang, and S. Chen, "Deep learning assisted mmWave beam prediction for heterogeneous networks: A dual-band fusion approach," *IEEE Trans. Commun.*, vol. 71, no. 1, pp. 115–130, Jan. 2023.

[36] S. Rezaie, E. Carvalho, and C. N. Manchon, "A Deep learning approach to location- and orientation-aided 3D beam selection for mmWave communications," *EEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 11110–11124, Dec. 2022.

[37] A. M. Elbir, W. Shi, K. V. Mishra, and S. Chatzinotas, "Federated multi-task learning for THZ wideband channel and DOA estimation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. Workshops (ICASSPW)*, Jun. 2023, pp. 04–10.

[38] L. Chen, C. Shen, P. Zhou, and J. Xu, "Collaborative service placement for edge computing in dense small cell networks," *IEEE Trans. Mobile Comput.*, vol. 20, no. 2, pp. 377–390, Feb. 2021.

[39] M. K. Somesula, S. K. Mothku, and S. C. Annadanam, "Cooperative service placement and request routing in mobile edge networks for latency-sensitive applications," *IEEE Syst. J.*, vol. 17, no. 3, pp. 4050–4061, 2023.

[40] P. Gao, R. Guo, H. Lu, and H. Zhang, "Multi-view sensor fusion by integrating model-based estimation and graph learning for collaborative object localization," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 9228–9234.

[41] O-RAN Working Group 1, *O-RAN Operations and Maintenance Interface 4.0*, O-RAN.WG1.O1-Interface.0-v04.00 Technical Specification, Nov. 2020. [Online]. Available: https://www.o-ran.org/specifications

[42] J. Peng, P. Zhang, L. Zheng, and J. Tan, "UAV positioning based on multi-sensor fusion," *IEEE Access*, vol. 8, pp. 34455–34467, 2020.

[43] Y. Song, Z. Xie, X. Wang, and Y. Zou, "MS-YOLO: Object detection based on YOLOv5 optimized fusion millimeter-wave radar and machine vision," *IEEE Sensors J.*, vol. 22, no. 15, pp. 15435–15447, Aug. 2022.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[45] S. Jain, S. Dash, and R. Deorari, "Object detection using COCO dataset," in *Proc. Int. Conf. Cyber Resilience (ICCR)*, Oct. 2022, pp. 1–4, doi: 10.1109/ICCR56254.2022.9995808.

[46] A. K. Gostar, T. Rathnayake, R. Tennakoon, A. Bab-Hadiashar, G. Battistelli, L. Chisci, and R. Hoseinnezhad, "Centralized cooperative sensor fusion for dynamic sensor network with limited field-of-view via labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 69, pp. 878–891, 2021.

[47] M. Raeisi-Varzaneh, O. Dakkak, A. Habbal, and B.-S. Kim, "Resource scheduling in edge computing: Architecture, taxonomy, open issues and future research directions," *IEEE Access*, vol. 11, pp. 25329–25350, 2023.

[48] H. Xiao, J. Huang, Z. Hu, M. Zheng, and K. Li, "Collaborative cloud-edge-end task offloading in MEC-based small cell networks with distributed wireless backhaul," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 14, pp. 4542–4557, Apr. 2023.

[49] S.-H. Park, S. Jeong, J. Na, O. Simeone, and S. Shamai (Shitz), "Collaborative cloud and edge mobile computing in C-RAN systems with minimal end-to-end latency," *IEEE Trans. Signal Inf. Process. Over Netw.*, vol. 7, pp. 259–274, Apr. 2021.

[50] A. S. Thyagaturu, P. Shantharama, A. Nasrallah, and M. Reisslein, "Operating systems and hypervisors for network functions: A survey of enabling technologies and research studies," *IEEE Access*, vol. 10, pp. 79825–79873, 2022.

[51] P. Li, E. Koyuncu, and H. Seferoglu, "Adaptive and resilient model-distributed inference in edge computing systems," *IEEE Open J. Commun. Soc.*, vol. 4, pp. 1263–1273, 2023.

[52] D. D. S. Brilhante, J. C. Manjarres, R. Moreira, L. D. O. Veiga, J. F. de Rezende, F. Müller, A. Klautau, L. L. Mendes, and F. A. P. de Figueiredo, "A literature survey on AI-aided beamforming and beam management for 5G and 6G systems," *Sensors*, vol. 23, no. 9, p. 4359, Apr. 2023.

[53] O. E. Ayach, S. Rajagopal, S. Abu-Surra, Z. Pi, and R. W. Heath, "Spatially sparse precoding in millimeter wave MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 13, no. 3, pp. 1499–1513, Mar. 2014.

[54] B. Salehi, J. Gu, D. Roy, and K. Chowdhury, "FLASH: Federated learning for automated selection of high-band mmWave sectors," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, May 2022, pp. 1719–1728.

[55] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[56] R. Mahima, M. Maheswari, S. Roshana, E. Priyanka, N. Mohanan, and N. Nandhini, "A comparative analysis of the most commonly used activation functions in deep neural network," in *Proc. 4th Int. Conf. Electron. Sustain. Commun. Syst. (ICESC)*, Jul. 2023, pp. 06–08.

[57] F. Liu, B. Zhang, G. Chen, G. Gong, H. Lu, and W. Li, "A novel configurable high-precision and low-cost circuit design of sigmoid and tanh activation function," in *Proc. IEEE Int. Conf. Integr. Circuits, Technol. Appl. (ICTA)*, Zhuhai, China, Nov. 2021, pp. 222–223, doi: 10.1109/icta53157.2021.9661606.

[58] X. Qian and D. Klabjan, "The impact of the mini-batch size on the variance of gradients in stochastic gradient descent," 2020, *arXiv:2004.13146*.

[59] E. P. Frady, D. Kleyko, and F. T. Sommer, "Variable binding for sparse distributed representations: Theory and applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 5, pp. 2191–2204, May 2023.

[60] D. Grimm, D. Tollner, D. Kraus, Á. Török, E. Sax, and Z. Szalay, "A numerical verification method for multi-class feed-forward neural networks," *Exp. Syst. Appl.*, vol. 247, Aug. 2024, Art. no. 123345, doi: 10.1016/j.eswa.2024.123345.

[61] T. T. Nguyen and K.-K. Nguyen, "A deep learning framework for beam selection and power control in massive MIMO–millimeter-wave communications," *IEEE Trans. Mobile Comput.*, vol. 22, no. 8, pp. 4374–4387, Aug. 2023.

[62] Y. Han, S. Jin, J. Zhang, J. Zhang, and K.-K. Wong, "DFT-based hybrid beamforming multiuser systems: Rate analysis and beam selection," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 3, pp. 514–528, Jun. 2018.

[63] Y. Koda, K. Nakashima, K. Yamamoto, T. Nishio, and M. Morikura, "Handover management for mmWave networks with proactive performance prediction using camera images and deep reinforcement learning," *IEEE Trans. Cognit. Commun. Netw.*, vol. 6, no. 2, pp. 802–816, Jun. 2020.

[64] S. G. Zadeh and M. Schmid, "Bias in cross-entropy-based training of deep survival networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 9, pp. 3126–3137, Sep. 2021.

[65] M. Mohsin, J. M. Batalla, E. Pallis, G. Mastorakis, E. K. Markakis, and C. X. Mavromoustakis, "On analyzing beamforming implementation in O-RAN 5G," *Electronics*, vol. 10, no. 17, p. 2162, Sep. 2021.

[66] M. Alrabeiah, A. Hredzak, Z. Liu, and A. Alkhateeb, "ViWi: A deep learning dataset framework for vision-aided wireless communications," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–5.

[67] Remcom. *Wireless Insite*. Accessed: Feb. 16, 2024. [Online]. Available: http://www.remcom.com/wireless-insite

RAFID I. ABD (Member, IEEE) received the B.S. degree in electronics and communications engineering from the University of Technology, Baghdad, Iraq, the M.S. degree in communications and networks, and the Ph.D. degree from the School of Electrical and Electronic Engineering, Yonsei University, Seoul, South Korea. He has three patents registered in communications and networks. His research interests include wireless communication and signal processing. He invented the Hydra Radio Access Network (H-RAN) for 6G and beyond.

DANIEL J. FINDLEY (Senior Member, IEEE) received the Ph.D. degree in civil engineering from North Carolina State University, Raleigh, NC, USA. He is currently the Associate Director of the Institute for Transportation Research and Education, NC State University. He specializes in economic impact analysis, multi-modal transportation, human behavior research, transportation engineering studies, and asset management and inventory. He served as a Principal Investigator (PI) or a Co-PI on over 80 funded projects and has conducted research for sponsors, including NCDOT, FHWA, ACRP, and ITE. He is also a Licensed Professional Engineer (P.E.) in NC, USA. He is also an Adjunct Assistant Professor with the Department of Civil, Construction, and Environmental Engineering, NC State University. He has studied many of the modes of travel including aviation, bicycle, ferry, highway, pedestrian, port, and railroad.

KWANG SOON KIM (Senior Member, IEEE) received the B.S. (summa cum laude), M.S.E., and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in February 1994, February 1996, and February 1999, respectively. From March 1999 to March 2000, he was a Postdoctoral Researcher with the Department of Electrical and Computer Engineering, University of California at San Diego, La Jolla, CA, USA. From April 2000 to February 2004, he was a Senior Member of the Research Staff with the Mobile Telecommunication Research Laboratory, Electronics and Telecommunication Research Institute, Daejeon. Since March 2004, he has been with the Department of Electrical and Electronic Engineering, Yonsei University, Seoul, South Korea, where he is currently a Professor. His research interests include signal processing, communication theory, information theory, stochastic geometry applied to wireless heterogeneous cellular networks, wireless local area networks, wireless D2D networks, wireless ad-hoc networks, and new radio access technologies for 5G. He was a recipient of the Postdoctoral Fellowship from Korea Science and Engineering Foundation (KOSEF), in 1999. He received the Outstanding Researcher Award from the Electronics and Telecommunication Research Institute (ETRI), in 2002, the Jack Neubauer Memorial Award (Best System Paper Award, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY) from IEEE Vehicular Technology Society, in 2008, and the LG R&D Award: Industry-Academic Cooperation Prize, LG Electronics, in 2013. From 2006 to 2012, he served as an Editor for the *Journal of the Korean Institute of Communications and Information Sciences* (KICS). From 2013 to 2016, he served as the Editor-in-Chief for the *Journal of the Korean Institute of Communications and Information Sciences*. Since 2008, he has been serving as an Editor of the *Journal of Communications and Networks* (JCN). From 2009 to 2014, he served as an Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.