COPILOT: Cooperative Perception using Lidar for Handoffs between Road Side Units

Suyash Pradhan *, Debashri Roy†, Batool Salehi*, and Kaushik Chowdhury*

*Institute for the Wireless Internet of Things, Northeastern University and †The University of Texas Arlington
Email: pradhan.suy@northeastern.edu, debashri.roy@uta.edu, {bsalehihikouei, krc}@ece.neu.edu

***Abstract*—This paper presents COPILOT, a ML-based approach that allows vehicles requiring ubiquitous high bandwidth connectivity to identify the most suitable road side units (RSUs) through proactive handoffs. By cooperatively exchanging the data obtained from local 3D Lidar point clouds within adjacent vehicles and with coarse knowledge of their relative positions, COPILOT identifies transient blockages to all candidate RSUs along the path under study. Such cooperative perception is critical for choosing RSUs with highly directional links required for mmWave bands, which majorly degrade in the absence of LOS. COPILOT proposes three modules that operate in an inter-connected manner: (i) As an alternative to sending raw Lidar point clouds, it extracts and transmits low-dimensional intermediate features to lower the overhead of inter-vehicle messaging; (ii) It utilizes an attention-mechanism to place greater emphasis on data collected from specific vehicles, as opposed to nearest neighbor and distance-based selection schemes, and (iii) it experimentally validates the outcomes using an outdoor testbed composed of an autonomous car and Talon AD7200 60GHz routers emulating the RSUs, accompanied by the public release of the datasets. Results reveal COPILOT yields upto 69.8% and 20.42% improvement in latency and throughput compared to traditional reactive handoffs for mmWave networks, respectively.**

***Index Terms*—AP association, mmWave, road side units, co-operative perception.**

I. Introduction

Autonomous vehicles equipped with a multitude of sensors may need to continuously relay large volumes of data to a central cloud for a number of tasks like cooperative path planning, situational awareness, and safety-related actions for itself along with neighboring vehicles and pedestrians [1]. Additionally, as augmented and virtual reality technologies start to become commonplace [2], [3], there will be an increased demand from passengers for real-time access to high-definition content. To address these requirements, 3GPP has prioritized vehicle-to-everything (V2X) communication in its next-generation (nextG) RAN Technical Specification Groups. One promising recommendation of this group is to leverage the millimeter-wave (mmWave) band within the frequency range of 57-72 GHz for V2X connectivity [4], enabling high data rate communication.

• **Problem of blockage prediction for mmWave V2X links.** To realize the full promise of V2X mmWave communication, the road side units (RSUs) that provide the connectivity to the vehicles are strategically situated in close proximity to one another. As the vehicle moves, handoffs between these RSUs at opportune times help to establish robust and uninterrupted mmWave band connectivity [1].

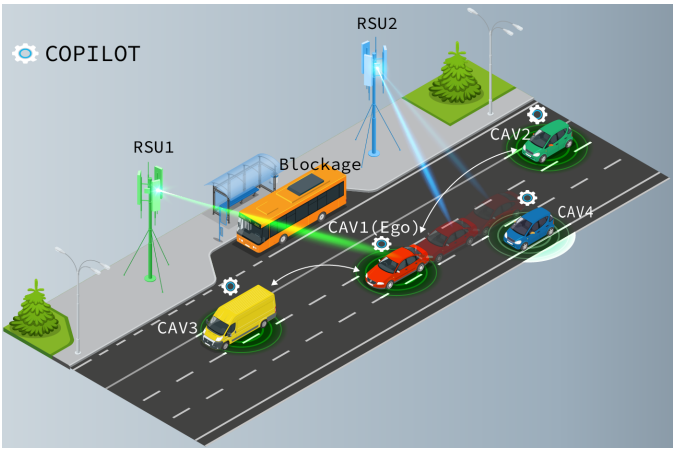

Fig. 1: The V2X mmWave scenario of interest and COPILOT framework. The ego vehicle requires to perform a proactive handoff from RSU2 to RSU1 leveraging the relevant contextual information from neighboring CAVs (CAV2-CAV4). The COPILOT framework selectively uses CAV2 and CAV3 for predicting that decision, rather than relying on the nearest neighboring vehicles CAV3 and CAV4.

Currently, handoff events are triggered when an impeding blockage is anticipated, based on evaluating metrics such as received power, throughput, or channel state information (CSI) associated with the mmWave link. Firstly, since these metrics need to be collected and averaged over time, there is an inevitable latency that impacts scenarios involving mobility. As shown in Fig. 1, a fast-moving connected and autonomous vehicle CAV1 must switch from RSU2 to RSU1, which needs to be done in a proactive manner to maintain a stable communication link. Furthermore, to establish new directional links upon breakage, probe packets are sent in different sectors, which further increases association time.

• **Co-operative perception for blockage prediction.** Recently, there has been a surge of works using vision and location based sensor data to predict blockage for improving mmWave link performance [5], [6] as well as perform proactive handoffs [7], [8]. However, these works focus on observing the environment from a single point of view, which limits their utility since highly discriminative features may remain undetected just beyond the sensor coverage region. As opposed to this, co-operative perception [9] merges mean-

ingful information obtained from spatially distributed nodes (see vehicles CAV2 and CAV3 in Fig. 1) to obtain a richer representation of the environment within the given node, here CAV1. In the domain of wireless communication, co-operative sensing using multi-camera systems have identified blockages to initiate handoffs [1]. Different from these approaches, our proposed COPILOT uses Lidar data collected at multiple vehicles and then fuses them together to provide precise location and obstacle information. We opt for Lidar because RGB camera images cannot provide depth cues, whereas depth camera images require additional processing to obtain a 3D map of the environment [10]. An example showing the benefit of co-operative perception using Lidar is presented in Fig. 2.

• **Problem setting and challenges.** Our vehicular setting is composed of two types of communication links: (a) vehicle to vehicle (V2V), which follows the 5G NR based C-V2X standard operating at $5.9GHz$ band [11], and (b) vehicle to infrastructure (V2I) operating in the mmWave band. As shown in Fig. 1, autonomous vehicles CAV1-CAV3 each collect Lidar data locally. Considering one such vehicle CAV1, which we refer to as the *ego* vehicle [12][13], the problem we address in this paper is: *which RSU from the set* $\{RSU_1, RSU_2\}$ *must the ego vehicle associate with?* To solve this problem, we solve three challenges given below:

(C1) *Compressing Lidar data:* Since the Lidar fusion occurs at each vehicle, transmitting raw Lidar data for spatial fusion will saturate the bandwidth, given a single capture is 3.076 MB. There is also a need to study the performance-bandwidth trade-off of any compression method.

(C2) *Lowering V2V overhead:* Since each vehicle contributes to the control overhead due to V2V messaging, how can we select the appropriate subset of vehicles? Including more vehicles may not only increase traffic overhead but may also lower performance of RSU selection due to noise in the data or require larger inference models.

(C3) *RSU selection methodology:* How to perform the RSU selection using the co-operative perception while adhering to the standard defined association procedure.

A. Proposed solution & contributions

We propose a machine learning (ML) based approach called COPILOT, which addresses challenges **C1**-**C3** above through context-aware spatial fusion. In Fig. 1, COPILOT chooses vehicles CAV2 and CAV3, but not CAV4, for participation in the data fusion process at the ego vehicle. This is non-intuitive since CAV4 is closer to the ego vehicle than CAV2, but CAV 2 can provide better supplementary information for enabling the optimal RSU prediction, making our supporting CAV selection different from distance based metrics. COPILOT locally quantizes the Lidar point clouds in a given vehicle, extracts semantic information using local feature extractor to reduce the overhead and identifies the suitable supporting CAVs that should participate in data sharing. Finally, the given ego vehicle CAV1 fuses the perception from these selected CAVs using a deep learning architecture with an attention mechanism to focus on the most relevant portions of the fused data for RSU selection.

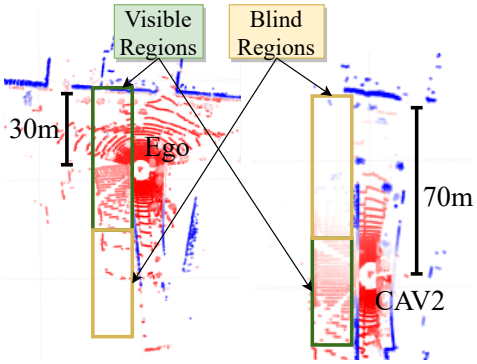

Fig. 2: A Lidar point cloud example showing the invisible regions in one frame when ego vehicle is $30m$ distant from a reference point (shown in *left*). At the same time, that invisible region is visible to CAV2 which is $70m$ distance from the same reference point (shown in *right*). Co-operative perception is useful for such cases as blind regions for one vehicle is in the visible regions for another one.

• **Summary of contributions.**

1) We study the performance tradeoffs along the accuracy-bandwidth axes for DL based RSU selection by varying the dimensionality of latent embeddings from our feature extractor. Using these results, we propose a bandwidth-adaptive feature sharing strategy to transmit the features based on available resource constraints. *Solution to* ***(C1)***
2) We propose a method that allows distributed selection of a subset of supporting CAVs for fusing the perception data and compare it with traditional distance-based selection in terms of model performance. We introduce an attention mechanism that assigns weights to focus on vehicles that provide more discriminative inputs to the overall inference process. *Solution to* ***(C2)***
3) We propose a deep learning model that predicts the best RSU for association by leveraging the fused Lidar sensor data from multiple viewpoints. We show that the spatial fusion approach results in 14.11% performance improvement over using local perception.
4) Using an experimental testbed of an autonomous car and four 802.11ad-enabled mmWave Talon routers as RSUs, we rigorously analyze the end-to-end latency in COPILOT. We also compare COPILOT with the standard association procedure for an example mmWave link and show 20.42% increase in the throughput with 69.8% of latency improvement. *Solution to* ***(C3)***
5) We publish the first (to the best of our knowledge) dataset for RSU selection in urban environments collected via an actual autonomous vehicle. This dataset [14] is of 164 GB can be used for vehicular channel estimation, power control of RSUs, link quality prediction, and thus, it has longevity beyond the scope of this paper.

II. Related Work

• **Handoff in mmWave links.** Palacio *et al.* present a location based access point (AP) selection with predictive handoff [15]. The authors measure the CSI that is used to estimate the relative positioning of APs and presence of blockages. Polese *et al.* implement a dual connectivity protocol providing cellular users 4G and 5G connectivity simultaneously, enabling them to instantly switch to the other standard in case of failure in any one link [16]. The authors monitor the UE channel quality and uplink control signals to detect link failures and perform the handoffs using a local co-ordinator. Mezzavilla *et al.* design a markov decision process-based framework that jointly considers channel conditions and dynamic load for providing handoff decisions [17]. Sun *et al.* propose a reinforcement learning-based approach by taking into account both mmWave channel characteristics and QoS requirements of users to reduce the number of handoffs [18].

• **Leveraging sensor modalities for proactive handoff.** Non-RF sensors are being increasingly used to gain contextual information about the environment and allows for proactive handover. Charan *et al.* utilize both RGB images and beamforming vectors and leverage deep learning models like Convolutional Neural Networks (CNNs) to perform proactive blockage prediction and user handoff [19]. Nishio *et al.* demonstrate that future received power can be predicted by leveraging spatio-temporal visual information obtained by depth-cameras and advanced ConvLSTM models, which in turn is used for performing proactive handoffs [7] . On similar lines, Koda *et al.* use camera images to predict the data rate degradation and apply a deep reinforcement learning model to decide the handover timing [8]. In another work, Koda *et al.* introduce co-operative sensing for enabling proactive handovers for mmWave links through multiple cameras [1] . They learn an optimal mapping of spatial location to handoff region using a deep reinforcement learning algorithm for identifying blockages and finally controlling the actions of the selected base station.

• **Innovation opportunity.** Prior works narrowly focus on demonstrating handoff outcomes with synthetic data, indoor setups and/or human blockages. To accurately model the challenging V2X environment, there is a need to perform real-world experiments involving data collected from an actual autonomous car that operates in a typical urban space with pedestrians, other vehicles and building-generated reflections.

III. System Model and Problem Formulation

A. Problem formulation

We denote the vehicle of interest as the *'ego' vehicle* ν_e, which is surrounded by $\mathcal{V}$ other connected and autonomous vehicles (CAVs) $V = \{\nu_i\}_{i=1}^{\mathcal{V}}$. We consider V2V links between them given by sub-6GHz C-V2X standard [11]. Furthermore, we assume directional mmWave V2I links to multiple RSUs $\{\tau_j\}_{j=1}^{\mathcal{R}}$ placed alongside the roadside to allow transferring high bandwidth sensor data. Both the ego vehicle and CAVs are equipped with Lidar sensors. The ego vehicle ν_e receives multiple perceptions of the environment from the surrounding

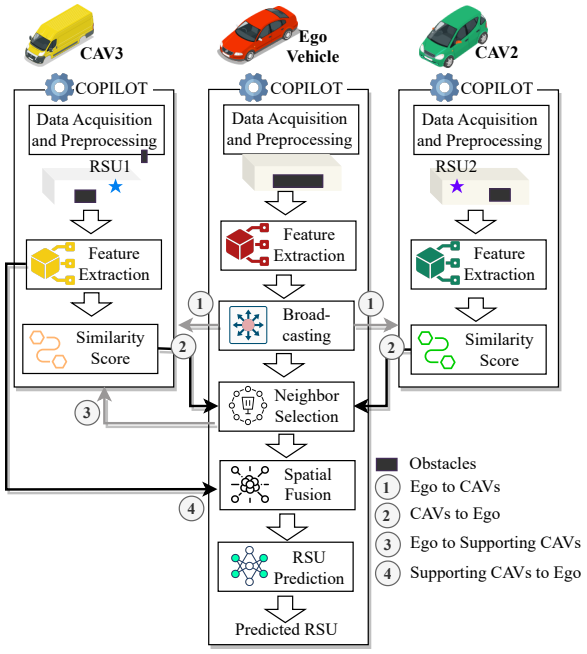

Fig. 3: An instance of COPILOT running at the 'ego' vehicle. Among the candidate options $CAV2$ and $CAV3$, the latter is actually chosen by the ego vehicle based on the similarity scores (details in Sec. III-C).

CAVs in form of Lidar feature maps through V2V links. Following this, it locally performs data fusion and analysis to select the optimal RSU for association. Similar to prior works [20], while the above procedure involves decentralized execution, COPILOT uses centralized training for the RSU selection model. We formulate our objective as:

$$\tau^* = \underset{1 \le m \le \mathcal{R}}{\arg\max}\ y_{\tau_m}^{\nu_e}, \tag{1}$$

with $y_{\tau_m}^{\nu_e}$ being the observed received signal strength at the ego vehicle ν_e when the mmWave radio is associated with the m^{th} RSU, i.e., τ_m.

Notations: We denote the ith CAV as ν_i, and jth RSU as τ_j. We define the data matrices for the Lidar data at ego vehicle and CAV ν_i as: $X_{\text{L}}^{\nu_e} \in \mathbb{R}^{d_0^{\nu_e} \times d_1^{\nu_e} \times d_2^{\nu_e}}$ and $X_{\text{L}}^{\nu_i} \in \mathbb{R}^{N_t^{\nu_i} \times d_0^{\nu_i} \times d_1^{\nu_i} \times d_2^{\nu_i}}$, respectively, where $(d_0^{\nu_e} \times d_1^{\nu_e} \times d_2^{\nu_e})$ and $(d_0^{\nu_i} \times d_1^{\nu_i} \times d_2^{\nu_i})$ are the dimensionality of Lidar data at the ego vehicle ν_e and CAV vehicle ν_i, respectively. Note that such Lidar data is generated for all the CAVs.

The set of output labels are the RSUs within the environment: $\mathcal{L} = \{\tau_1, \cdots, \tau_{\mathcal{R}}\}$, where $\mathcal{R}$ is the total number of RSUs in the environment. The label matrix $Y_\tau \in \{0,1\}^{\mathcal{R}}$ represents the one-hot encoding of $\mathcal{R}$ RSUs, where the optimum RSU is set to 1, and rest are set to 0 as per Eq. (1).

B. Offline centralized training of COPILOT

Training dataset: We collect a custom dataset (presented in Sec. V) from a real Lidar sensor equipped autonomous car with multiple mmWave RSUs in a typical urban road environment. Existing datasets for autonomous driving [21], [22] provide Lidar sensor data but do not include wireless signal measurements. Wireless-focused datasets like DeepSense

6G [23] and FLASH [24] include Lidar sensor data and mmWave signal measurements but are limited to the use case of beam prediction and also do not include multiple RSUs in the testbed.

Training objective: Training the model in COPILOT requires labeled data from Lidar sensors installed in multiple CAVs and the ego vehicle, where the ground truth indicates the best RSU for each sample data point. In COPILOT, the model $p^{\text{L}}_{\theta^{\nu_e}}$ within the ego vehicle ν_e is parameterized by θ^{ν_e}, i.e., a neural network with weights θ^{ν_e}. The empirical loss of the model parameters θ^{ν_e} on dataset is defined as $\mathcal{L}(\theta^{\nu_e}) := \frac{1}{n_l}\sum_{j=1}^{n_l}[\ell(p^{\text{L}}_{\theta^{\nu_e}}(X^{\nu_e}_{\text{L}}(j), X^{\nu_1}_{\text{L}}(j), \cdots, X^{\nu\nu}_{\text{L}}(j)), Y_\tau(j))]$, where n_l is the total training samples, ℓ is a loss function measuring the discrepancy between predicted and true labels. The deep learning training approach finds a model that minimizes the loss across all of the training samples by solving: $\min_{\theta^{\nu_e}} \mathcal{L}(\theta^{\nu_e})$ over multiple training epochs.

C. Decentralized execution of COPILOT

COPILOT consists of seven main modules as follows (see Fig. 3):

- **Data acquisition and preprocessing** *(at ego vehicle and CAVs)*: The vehicles record Lidar sensor data. For Lidar preprocessing, we employ a quantization technique that marks RSU and vehicle positions in the point clouds and the remaining detected objects as obstacles; see Sec. IV-A.
- **Local feature extraction** *(at ego vehicle and CAVs)* : The ego vehicle and CAVs perform high level Lidar feature extraction; see Sec. IV-B.
- **Broadcasting ego feature map** *(from ego vehicle to CAVs)*: The ego vehicle broadcasts its extracted feature map to all the CAVs; see Sec. IV-C.
- **Similarity score computation** *(at the CAVs)*: The CAVs calculate similarity scores between the ego feature map and their own feature map. The similarity scores are transmitted to the ego vehicle using V2V links; see Sec. IV-D.
- **Neighbor selection and requesting the feature maps** *(at the ego vehicle)*: Based on the received similarity scores the ego vehicle selects the supporting CAVs and requests corresponding feature maps; see Sec. IV-E.
- **Spatial fusion** *(at ego vehicle only)*: The ego vehicle performs the spatial fusion combining its own feature map with those from the supporting CAVs; see Sec. IV-F.
- **RSU prediction** *(at ego vehicle only)*: The ego vehicle performs the RSU prediction from the fused features in real-time; see Sec. IV-G.

IV. COPILOT Framework

A. Data acquisition and preprocessing

In this section we describe how the Lidar point clouds are quantized locally by each CAV and then combined together.

1) Co-ordinate system: We project the Lidar point clouds to the ego vehicle's co-ordinate system. While GPS data provides latitude and longitude coordinates in degrees, the Lidar data and subsequent distance-based processing are more efficient when locations are transformed into a *Cartesian* coordinate system. We then establish a fixed origin point that serves as the reference for all the subsequent collaboration among the CAVs.

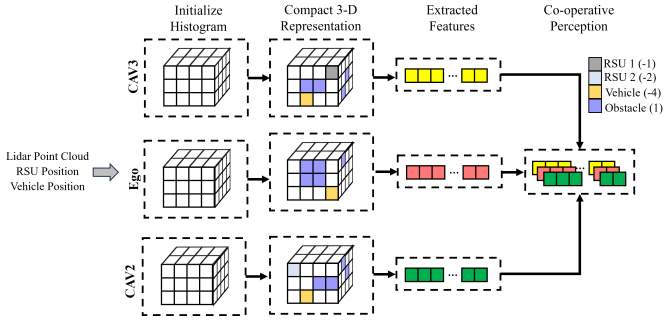

Fig. 4: The proposed Lidar point cloud acquisition and preprocessing method in COPILOT. We only show the quantization involving 2 RSUs for simplicity. The details are in Sec. IV-A2.

2) Preprocessing and quantizing Lidar data: The Lidar point clouds consist of an unstructured set of points in 3-D space. However, the permutation invariance of these points poses a challenge in leveraging popular CNN architectures. Unlike CNNs, which process ordered grid structures like image pixels, rearranging the order of Lidar points does not alter the represented scene. Therefore, we need to convert Lidar point clouds into a structured, ordered grid representation of the 3-D space through a 3-D quantized cuboid structure (see Fig. 4). Each unit within this structure is called a *voxel*, which stores the occupancy information of point clouds. Voxel values are set to 1 if they contain at least one point, indicating the presence of obstacles in that specific region. Conversely, unoccupied voxels are assigned a value of 0, while the ego vehicle's voxel is labeled -4. Similarly, voxels containing RSUs $1-3$, are labeled as -1, -2, and -3, respectively. All CAVs project their data onto a common coordinate system centered at the ego vehicle ν_e, with the only variation being the position encoding of -4 to identify the receiver's location (at ν_e), relative to surrounding RSUs. Hence, $X^{\nu_e}_{\text{L}}, X^{\nu_1}_{\text{L}}, \cdots, X^{\nu\nu}_{\text{L}}$ are generated at the ego vehicle ν_e and V CAVs as quantized Lidar data.

B. Local feature extraction

Existing studies have proven aggregating intermediate features instead of raw data as an effective strategy [25], [26]. Therefore, after preprocessing we extract feature maps locally on the vehicles. We denote the dimension of the extracted feature as d^{ν_e} and d^{ν_i} for ego vehicle ν_e and ith CAV ν_i, respectively. The feature extractors $f^{\text{L}}_{\theta^{\nu_e}}$ and $f^{\text{L}}_{\theta^{\nu_i}}$, which are pre-trained through centralized training, map each sample input to dimension d^{ν_e} and d^{ν_i}, for the ego vehicle and ith CAV, respectively. We refer to the output of these feature extractors as the latent embeddings of the Lidar data. Formally,

$$\mathbf{z}^{\nu_e}_{\text{L}} = f^{\text{L}}_{\theta^{\nu_e}}(X^{\nu_e}_{\text{L}}),\ f^{\text{L}}_{\theta^{\nu_e}} : \mathbb{R}^{N^{\nu_e}_t \times d^{\nu_e}_0 \times d^{\nu_e}_1 \times d^{\nu_e}_2} \mapsto \mathbb{R}^{d^{\nu_e}} \quad (2a)$$

$$\mathbf{z}^{\nu_i}_{\text{L}} = f^{\text{L}}_{\theta^{\nu_i}}(X^{\nu_i}_{\text{L}}),\ f^{\text{L}}_{\theta^{\nu_i}} : \mathbb{R}^{N^{\nu_i}_t \times d^{\nu_i}_0 \times d^{\nu_i}_1 \times d^{\nu_i}_2} \mapsto \mathbb{R}^{d^{\nu_i}} \quad (2b)$$

where $\mathbf{z}^{\nu_e}_{\text{L}}$ and $\mathbf{z}^{\nu_i}_{\text{L}}$ show the extracted latent embeddings for the quantized Lidar data $X^{\nu_e}_{\text{L}}$ and $X^{\nu_i}_{\text{L}}$ for ego ν_e vehicle and

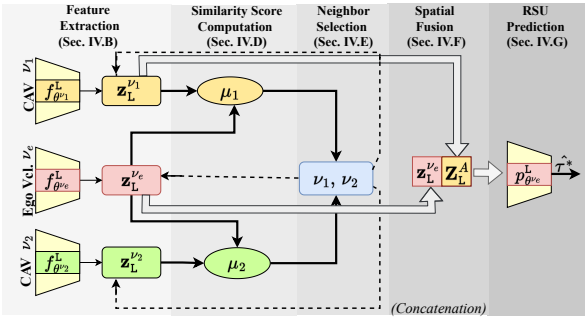

Fig. 5: Proposed attention based spatial fusion approach. The feature map from ego vehicle is broadcasted to all the CAVs for similarity score computation. The ego vehicle selects η CAVs as 'supporting CAVs' (for simplicity we show it selected only ν_1, hence, $\eta = 1$), to concatenate an extra component to its own feature map.

CAV ν_i, respectively. It is to be noted that at this stage, the latent embeddings from all the $\mathcal{V}$ CAVs are extracted, which is denoted as: $\mathbf{Z}_{\text{L}}^{\mathcal{V}} = \{\mathbf{z}_{\text{L}}^{\nu_i}\}_{i=1}^{\mathcal{V}}$.

C. Broadcasting ego feature map

In order to establish connections between the ego vehicle and CAVs, the simplest approach is to select the nearest neighbors or utilize pre-defined distance based approaches. However, these methods are not scalable when it comes to widespread deployment [20]. As opposed to this, COPILOT chooses the CAVs that ego vehicle should connect with for co-operative perception by computing a similarity metric between the sensed features of the ego vehicle and other CAVs. We note that broadcasting features from all the CAVs will likely saturate the V2V channel between the vehicles. Therefore, we follow the multi-stage handshaking method proposed in [20], followed by our distributed method of selecting the best CAVs using an attention mechanism. The handshaking is initiated once all the CAVs and ego vehicle extract their feature maps locally in the vehicle, i.e., after the ego vehicle ν_e broadcasts its own extracted feature map $\mathbf{z}_{\text{L}}^{\nu_e}$ to all the V CAVs.

D. Similarity score computation at the CAVs

Each CAV calculates a similarity score between the received ego feature map and its own feature map, and sends this score back to the ego vehicle using the V2V link. The similarity score μ_i is calculated at the CAV ν_i from the received ego feature map $\mathbf{z}_{\text{L}}^{\nu_e}$ and it's own feature map $\mathbf{z}_{\text{L}}^{\nu_i}$ as: $\mu_i = \zeta(\mathbf{z}_{\text{L}}^{\nu_e}, \mathbf{z}_{\text{L}}^{\nu_i}), \quad \forall \nu_i \in \{\nu_1, \nu_2, \cdots, \nu_{\mathcal{V}}\}$, where $\zeta(.)$ is the similarity function which uses *general attention* mechanism [27]. The similarity function $\zeta(.)$ is represented as: $\zeta(\mathbf{z}_{\text{L}}^{\nu_e}, \mathbf{z}_{\text{L}}^{\nu_i}) = \mathbf{z}_{\text{L}}^{\nu_e} \cdot \mathbf{W} \cdot \mathbf{z}_{\text{L}}^{\nu_i}$. Here, $\mathbf{W}$ is a learnable parameter that allows $\mathbf{z}_{\text{L}}^{\nu_e}$ and $\mathbf{z}_{\text{L}}^{\nu_i}$ to be of different sizes. Once the similarity score μ_i is calculated at the CAV ν_i, it is sent to the ego vehicle ν_e using the V2V link. All the $\mathcal{V}$ CAVs follow the same process.

E. Neighbor selection and requesting the feature maps

The ego vehicle ν_e receives $\mathcal{V}$ similarity scores from the neighboring CAVs, where μ_i is a representative similarity score from CAV ν_i. We calculate a probability vector Θ for the similarity scores received from $\mathcal{V}$ CAVs using *Softmax* operation, denoted as: $\Theta = \sigma([\mu_1; \cdots; \mu_{\mathcal{V}}])$ $\sigma : \mathbb{R} \mapsto \{0,1\}^{\mathcal{V}}$. Formally, Θ is represented as: $\{\theta_i\}_{i=1}^{\mathcal{V}} \in \mathbb{R}^{(0,1)}$. COPILOT performs a threshold based selection of suitable neighbours, with the resulting subset referred to as *supporting CAVs* for that ego vehicle ν_e. If θ_i is greater than a pre-defined threshold ϕ, we select ν_i as a supporting CAV for fusion with the ego vehicle ν_e. We define the set of supporting CAVs as $V_\eta = \{\nu_i^s\}_{i=1}^{\eta}$, where V_η is a subset of of the set of all CAVs V and $|V_\eta| = \eta$ with similarity scores $[\mu_1^s; \cdots; \mu_\eta^s]$. Formally,

$$V_\eta = \underset{V' \subset V, |V'|=\eta}{\arg\max} \sum_{i \in V'} (\theta_i - \phi). \tag{3}$$

Finally, the ego vehicle ν_e requests for the feature maps from the selected V_η supporting CAVs.

F. Spatial fusion

The ego vehicle ν_e receives the feature maps from the η supporting CAVs $V_\eta = \{\nu_i^s\}_{i=1}^{\eta}$, which are represented as $\mathbf{Z}_{\text{L}}^{V_\eta} = \{\mathbf{z}_{\text{L}}^{\nu_i^s}\}_{i=1}^{\eta}$. The proposed fusion approach has two components, (i) a weighted aggregation of the feature maps $\mathbf{Z}_{\text{L}}^{V_\eta}$ from the supporting CAVs V_η, and (ii) the feature map $\mathbf{z}_{\text{L}}^{\nu_e}$ from the ego vehicle ν_e. The steps for fusing these two components are:

- The weighted aggregation is performed through element-wise multiplication between the θ_i and corresponding $\mathbf{z}_{\text{L}}^{\nu_i^s}$ from supporting CAV ν_i^s. The aggregated vector $\mathbf{Z}_{\text{L}}^{A}$ is derived by: $\mathbf{Z}_{\text{L}}^{A} = \sum_{i=1}^{\eta} (\theta_i * \mathbf{z}_{\text{L}}^{\nu_i^s})$.
- To achieve cooperative perception, we combine the ego feature map $\mathbf{z}_{\text{L}}^{\nu_e}$ and aggregated feature map $\mathbf{Z}_{\text{L}}^{A}$ from the ego vehicle ν_e and supporting η CAVs V_η given by $\mathbf{z}_{\text{L}}^{\nu_e} \in \mathbb{R}_{\text{L}}^{d^{\nu_e}}$ and $\mathbf{Z}_{\text{L}}^{A} \in \mathbb{R}^{d^{\nu_1^s} + \cdots + d^{\nu_\eta^s}}$, respectively. The final spatially fused feature $\mathbf{z}_{\text{L}}$ representing the co-operative perception is generated by concatenation of the $\mathbf{z}_{\text{L}}^{\nu_e}$ and $\mathbf{Z}_{\text{L}}^{A}$. Formally, $\mathbf{z}_{\text{L}} = [\mathbf{z}_{\text{L}}^{\nu_e}; \mathbf{Z}_{\text{L}}^{A}] \in \mathbb{R}^{d^{\nu_e} + d^{\nu_1^s} + \cdots + d^{\nu_\eta^s}}$.

G. RSU prediction at the ego vehicle

The final step within COPILOT is the real time prediction of the RSUs at each location of the ego vehicle ν_e. The model trained in COPILOT $p_{\theta^{\nu_e}}^{\text{L}}$ (see Sec. III-B) predicts the probability of each class using the fused features $\mathbf{z}_{\text{L}}$. The neural network transformation of $p_{\theta^{\nu_e}}^{\text{L}}$ is represented as:

$$\hat{Y}^{\tau} = \sigma(p_{\theta^{\nu_e}}^{\text{L}}(\mathbf{z}_{\text{L}})) \qquad p_{\theta^{\nu_e}}^{\text{L}} : \mathbb{R}^{d^{\nu_e} + d^{\nu_1^s} + \cdots + d^{\nu_\eta^s}} \mapsto \{0,1\}^{\mathcal{R}}, \tag{4}$$

where σ is *Softmax* activation and $\hat{Y}^{\tau}$ is the generated probability vector. Eq. (4) can also be represented in terms of the original input data and the trained model $p_{\theta^{\nu_e}}^{\text{L}}$ as:

$$\hat{Y}^{\tau} = \sigma(p_{\theta^{\nu_e}}^{\text{L}}(X_{\text{L}}^{\nu_e}, X_{\text{L}}^{\nu_1}, \cdots, X_{\text{L}}^{\nu_{\mathcal{V}}})). \tag{5}$$

The predicted RSU $\hat{\tau*}$ from the generated probability vector, $\hat{Y}^{\tau}$ is defined as:

$$\hat{\tau*} = \underset{0 \leq \tau \leq len(\mathbb{I})}{\arg\max} \ (\hat{Y}^{\tau}) \tag{6}$$

An example of the similarity score calculation, neighbor selection, and spatial fusion is shown in Fig. 5.

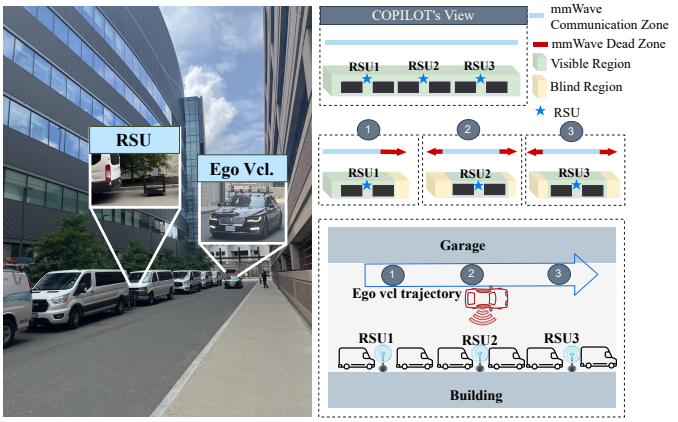

Fig. 6: Real-world data collection setup for COPILOT (*left*) showing a partial view of the road with one RSU and the ego vehicle, for Category 3 (see Table. I). The *bottom right* figure shows the overall setup with 3 RSUs and ego vehicle moving left to right. Three positions of the ego vehicle are marked as ①, ②, and ③. The *middle right* figure shows individual Lidar perception that is susceptible to blind regions. The *top right* figure shows the proposed method of co-operative perception in the ego vehicle, which eliminates blind regions through spatial fusion.

V. Experimental Setup for Validating COPILOT

We validate COPILOT using a real-world dataset collected in an autonomous car equipped with Lidar and GPS.

A. Scenario selection and sensors

The data collection campaign was conducted on days having dry, low humid weather conditions in a two-way urban street surrounded by a multi-storied brick parking garage and an office building with a mixed facade of glass, brick and metal, as shown in Fig. 6. Both the buildings are located at least 4ft (1.2m) from either side of the road with small trees and bushes on the sidewalk along with parked vehicles by the roadside. We leverage these features to create varying scenarios observed in a typical V2X network. We use a Lincoln Mkz Hybrid autonomous car equipped with a GPS system and a state-of-the-art Ouster OS1-64 channel Lidar that captures a panoramic 360-degree field of view. These sensors are integrated with an on-board computer running robot operating system (ROS) for logging location, data storage and synchronization.

B. Configuring radios

We utilize Talon TP-Link AD7200 tri-band routers, equipped with Qualcomm QCA 9500 Wi-Fi chips, to establish connectivity in the 60 GHz mmWave frequency band. One of these routers is mounted on the roof of our autonomous vehicle to serve as a receiver, while the remaining three function as RSUs, separated by $20m$ along a straight line. We leverage the open-source Linux Embedded Development Environment (LEDE) along with Nexmon firmware patches [28], [29] to extract physical layer information. We configure the radio attached to the vehicle's roof in the `managed` (client) mode and set the RSUs to `access point` mode following the IEEE 802.11ad standard. We record time-synchronized RF ground truth data, including data transmission rates and received signal strength indication (RSSI), to establish associations with the RSUs. We record all corresponding RSSI values from candidate RSUs but select the RSU with the highest RSSI as the associated RSU.

TABLE I: Summary of different categories in COPILOT dataset. The total number of samples in the dataset is 11068.

Category	Blockage	Scenario	# Episodes	# Samples
1	No obstacle	LOS	10	3066
2	Pedestrian	Static	10	2826
3	Vehicle	Static	10	2384
4	Static vehicle and pedestrian	Left to right	5	1425
		Right to left	5	1367

C. Data collection and pre-processing

1) Categorization: We collect the COPILOT dataset replicating the real-world vehicular network scenarios, categorized in: (a) Line of Sight (LOS), (b) Non-Line of Sight (NLOS) with a pedestrian in front of the RSU, (c) NLOS surrounded by vehicles around the RSU, (d) NLOS surrounded by vehicles and a pedestrian walking around the RSU. For each category, we collect 10 episodes, or trials, with episode duration of approximately 10 seconds. We limit the vehicle's speed to 15 mph, which is typical for inner-city roads. Additional details about the datasets in given in Table I.

2) Synchronization: The GPS and Lidar sensors have different sampling rates of 0.5 Hz and 20 Hz, respectively. Therefore, we prepare a synchronized dataset along with the optimal RSU associated at that time as the ground truth label. We consider Lidar frequency as our reference sampling rate and up-sample the GPS and RF data accordingly.

VI. Performance Evaluation

In this section, we rigorously analyze four key features of COPILOT, as given below:

1) Co-operative perception: We evaluate the performance of COPILOT by fusing multiple viewpoints $(X_{\text{L}}^{\nu_e}, X_{\text{L}}^{\nu_1}, \cdots, X_{\text{L}}^{\nu_\mathcal{V}})$ under a variety of scenarios and compare each case with a single perception (only $X_{\text{L}}^{\nu_e}$) RSU prediction (notations from Sec. IV-A).

2) Communication aware feature sharing: We analyze the impact of using different dimensions of the ego and CAV feature maps ($\mathbf{z}_{\text{L}}^{\nu_e}$ and $\mathbf{Z}_{\text{L}}^{\mathcal{V}}$ respectively) to observe the effect on RSU prediction performance (notations from Sec. IV-B).

3) Neighbor Selection: The strategy to select the supporting CAVs (V_η) for co-operative perception proposed in COPILOT is compared with a pre-defined distance selection by observing model performance (notations from Sec. IV-E).

4) Spatial fusion strategies: We compare the aggregation method (details in Sec. IV-F) proposed in COPILOT with concatenation [30] and averaging [31].

A. Experiment settings

We partition our dataset into 72% training, 18% validation and 10% test dataset for hyper-parameter tuning. The overall dataset contains around 8275 and 1660 local training and validation and 1104 test samples, respectively. For the Lidar

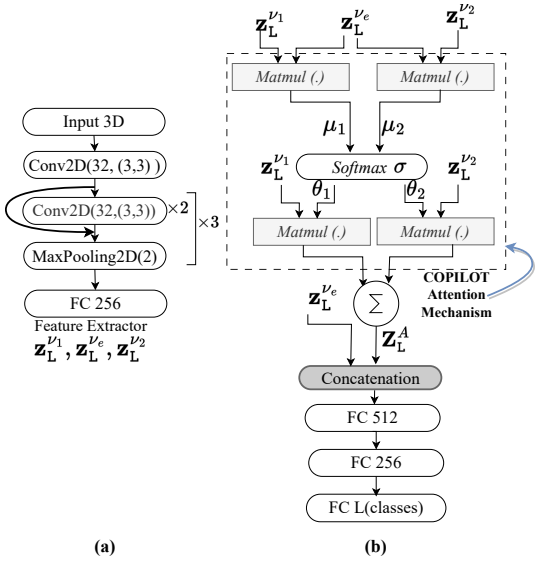

Fig. 7: Proposed neural network architectures for (a) Lidar feature extractor (at CAVs ν_1 and ν_2 and ego vehicle ν_e) and (b) the spatial fusion network with specialized attention on the extracted features (at ego vehicle ν_e).

quantization module, we select a $20m$ radius around each CAV and quantize each axis to a $(20, 20, 4)$ grid, with each voxel set to size $(2, 2, 1)$. We use NVIDIA DGX A100 computing clusters to train our models with CUDA based GPU acceleration. The details of the neural network models are presented in Fig. 7. We use categorical cross-entropy loss for training with learning rate of 0.0001, batch size of 8 for 50 epochs, and optimize using Adam optimizer with the first and second moment terms set as $\beta = (0.9, 0.999)$.

We evaluate the proposed models based on the validation and $top-1$ testing accuracy for predicting the optimal RSU at any location of the ego vehicle ν_e. We analyze the latency in terms of computational delay of the neural network modules which execute on ego vehicle and CAVs independently and also account for the communication delay due to V2V transmissions considering 5G NR based C-V2X standard [11], operating at $5.9GHz$ band. We compute the throughput gain for our deep learning based approach with respect to the *reactive handoff* [19] as a representative classical method.

B. Results and observations

1) Co-operative perception: We first verify the benefit of co-operative perception as compared to the decision made by individual perception only at the ego vehicle ν_e. We adopt a curriculum learning [32] strategy to ensure robust training of our model, wherein we expose the model easier LOS samples at the start and gradually increase the difficulty by introducing categories 1-4 as shown in Table I, in that order.

Observation 1. *Fusing multiple perceptions outperforms single perception view by 14.11% at the ego vehicle, in terms of testing accuracy (see Fig. 8, validates Contribution 3).*

2) Communication aware feature sharing: In COPILOT, the *z-axis*, denoted as $d_2^{\nu_e}$ for ego vehicle ν_e and $d_2^{\nu_i}$ for ith CAV (notation details in Sec. III-A and Sec. IV-B) of the

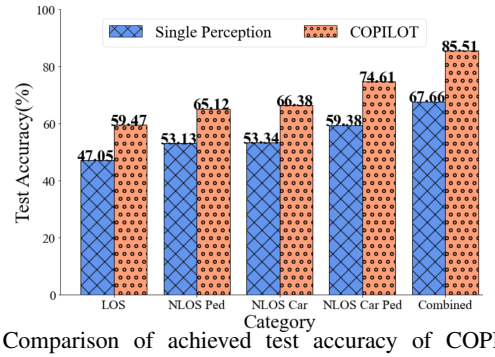

Fig. 8: Comparison of achieved test accuracy of COPILOT vs. single perception for RSU prediction (details in Sec. VI-B1, *validates Contribution 3).*

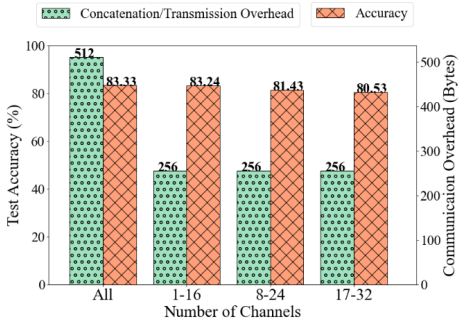

Fig. 9: Performance of communication aware feature sharing in COPILOT *(details in Sec. VI-B2, validates Contribution 1).*

pre-processed Lidar is denoted as *channels*. These channels capture semantic information such as the heights of the RSUs, receiver location and obstacles in the vicinity. In our experiment, at the end of the feature extraction process, the feature maps of both ego vehicle and CAVs have 32 channels. However, the semantic information contained within these 32 channels can be redundant (like buildings not obstructing the LOS), repetitive, or may not contribute significantly to neighbor selection and final prediction. Therefore, we explore transmitting a subset of channels both from the ego vehicle for similarity score computation as well as feature maps from supporting CAVs to reduce control traffic overhead.

Since COPILOT permits CAV feature maps of different sizes (see Sec. IV-D), divide the 32 channel intermediate feature map into 3 subsets: initial $(0-16)$, mid $(8-24)$ and final $(17-32)$ to selectively utilize these channels. Fig. 9 highlights the test accuracy with respect to these selective channels transmitted and processed in subsequent layers. We observe that test accuracy only drops by $1-3\%$ when using any of these 3 subsets (the last three bars in Fig. 9) as opposed to using 'All' channels (the first bar in Fig. 9), with 50% decrease in communication overhead.

Observation 2. *Using subset of channels from the extracted Lidar feature maps yields to similar performance as compared to using all the channels while decreasing the communication overhead significantly (see Fig. 9, validates Contribution 1).*

Observation 2 suggests an exciting area of further research involving adaptive Lidar feature sharing depending on the available network resources. How to optimize this sharing level for a particular scenario is left for future work.

TABLE II: Performance of COPILOT based attention mechanism compared to nearest neighbor baseline and prior knowledge benchmark. The COPILOT provides optimal performance without any required prior knowledge *(validates Contribution 2)*.

Neighbor Selection Approach	Test Accuracy (%)	Prior knowledge?
Nearest neighbor baseline	84.01%	No
Spatial fusion of COPILOT	**88.28%**	**No**
Expert knowledge benchmark	90.09%	Yes

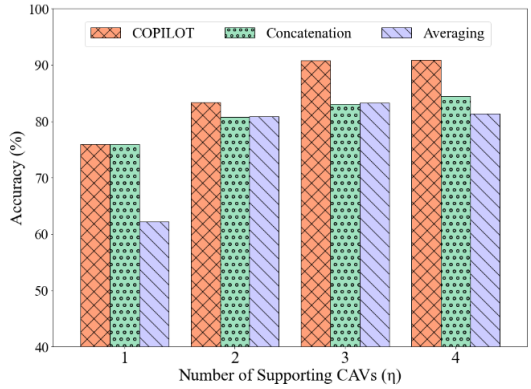

Fig. 10: Comparing the performance of proposed spatial fusion in COPILOT with concatenation [30] and averaging [31], for different number of supporting CAVs (η) *(validates Contribution 2)*.

3) Neighbor selection: In this set of experiments, we demonstrate the performance of the attention based neighbor selection approach in COPILOT by comparing it with two competing methods: (a) **nearest neighbor** and (b) **prior knowledge-based**. The *nearest neighbor* serves as the baseline that involves selecting the nearest neighbors for performing the fusion. Here, we consider a triplet composed of the ego vehicle along with two co-operative CAVs in front and back of the former, separated by $7\,\mathrm{m}$ distance. This distance takes into consideration the dimension of the vehicle itself along with the a buffer safety distance between two vehicles. We show that selecting the nearest neighbors as supporting CAVs may not provide new information. For the *prior knowledge-based* benchmark, we assume that we know the position of each vehicle in the environment by fusing the relevant feature maps from the CAVs. Here, we consider a triplet of the ego vehicle along with the two supporting CAVs separated by a fixed distance. We select $20\mathrm{m}$ distance for our experiments by analyzing the spatial extent of our test environment, such that all the RSUs are observable, which is an idealized decision. However, in a practical scenario we will require a central entity having prior knowledge of optimal distance of vehicle separation, which is intractable in a dynamic vehicular environment. In Table. II, the testing accuracy of COPILOT is compared with these competing methods.

Observation 3. *The distributed selection proposed in COPILOT outperforms the nearest neighbor baseline by improved test accuracy and shows comparable test accuracy to the prior knowledge benchmark, without intervention of a central entity (see Table. II, validates Contribution 2).*

4) Spatial fusion strategies: We next compare the COPILOT with two widely used fusion strategies: (a) **concatenation based fusion** [30] and (b) **averaging based fusion** [31]. In *concatenation based fusion*, the ego and co-operative feature maps are concatenated at the cost of high computational and communication overhead. On the other hand, *averaging based fusion* keeps a fixed-size feature vector and averages all the feature maps for further processing. This will result in modifying the primary perception from the ego vehicle. Thus, we keep the ego feature map intact and concatenate the other supporting features after performing a weighted average from supporting CAVs. Additionally, we use this experiment setup to gain insights on the selection of η (the number of supporting CAVs), which will be ideal for obtaining suitable predictions. We vary η from 1 to 4 and observe the performance in terms of overhead and spatial fusion schemes.

Observation 4. *COPILOT outperforms concatenation and averaging fusion in testing accuracy, for larger values of η. It shows maximum accuracy of 90.91% when 3 supporting CAVs are involved, i.e., $\eta = 3$. Beyond a certain threshold number of supporting vehicles, there is no significant increase in performance. (see Fig. 10, validates Contribution 2)*

C. End-to-end latency analysis of COPILOT

In 5G NR that is used for the V2V links in COPILOT , the sub-carrier spacing (SCS) varies based on the selected numerology ($\mu = \{0, 1, 2, 3, 4\}$) [4]. We consider $\mu = 1$ which corresponds to the sub-6 GHz band, where each Resource Block (RB) includes 12 sub-carriers, each with 30 KHz SCS, resulting in a 360 KHz RB. Considering 11 data symbols in a 0.5 ms slot and 256 QAM modulation, we calculate the total number of bits transmitted as 977. Assuming 100 MHz bandwidth for each user, a total of 273 RBs are available for data transmission. Overall, the throughput for the sub-6 GHz control channel is estimated as $977\ \text{bits} \times 273\ \text{RB} \times 2000\ \text{slots} \approx 63.59\ \text{MBps}$.

1) End to end latency calculation: COPILOT performs RSU selection by following the steps discussed in Sec. III-C and IV. We pass the test dataset of 1100 samples to the model and calculate average execution time as follows:

(i) Data acquisition and quantization: We assume that the time to acquire the sensor data and perform the Lidar quantization step described in Sec. IV-A incurs negligible latency, hence $\mathtt{T_i} \approx 0$.

(ii) Local feature extraction: It takes $1.995ms$ to perform the convolutional operations and extract the spatial correlations in the quantized Lidar grid, hence $\mathtt{T_{ii}} = 1.995ms$.

(iii) Broadcasting the ego feature map: The ego feature map consists of 128 tensor elements of 4 bytes each with 32 channels, which amounts to a total of 512 bytes. Using the $63.59\mathrm{MBps}$ C-V2X channel for V2V communication, the time required to broadcast the ego feature map to all the CAVs is $0.00805ms$, hence $\mathtt{T_{iii}} = 0.00805ms$.

(iv) Similarity score computation: This steps involves a matrix multiplication operation of the feature map from ego vehicle and CAV which takes around $2.14ms$. Next, all the CAVs send their respective scores which is one tensor element of 4 bytes, which takes $0.06\mu s$. We consider the worst case

scenario with all the CAVs sending the scores sequentially. Following the traffic data on an urban road of length 100 meter [33][34], we consider 17 such CAVs on the road. Hence, $\mathtt{T_{iv}} = 2.14ms + 17 \times 0.06\mu s = 2.14102ms$.

(v) Neighbor selection: Here, we compute the *Softmax* of the vector containing the received scores from CAVs to obtain a probabilistic range of scores and select the appropriate subset of supporting CAVs, which takes $2.18ms$, hence $\mathtt{T_v} = 2.18ms$.

(vi) Spatial fusion: Next, we send feature maps from 3 supporting CAVs (which is sufficient for obtaining prediction accuracy around 91% from Fig. 10), each of size 512 bytes which takes $0.012075ms$. Computing the weighted average and spatial fusion in COPILOT takes $0.14ms$, hence $\mathtt{T_{vi}} = 3 \times 0.012075ms + 0.14ms = 0.176225ms$.

(vii) RSU prediction: The computation in the final layers to select the RSU takes about $0.38ms$, hence $\mathtt{T_{vii}} = 0.38ms$.

The end to end latency is calculated as: $\mathtt{T_C} = \mathtt{T_i} + \mathtt{T_{ii}} + \mathtt{T_{iii}} + \mathtt{T_{iv}} + \mathtt{T_v} + \mathtt{T_{vi}} + \mathtt{T_{viii}} \approx 6.88$ms.

Observation 5. *The end-to-end latency of COPILOT framework is $6.88ms$ which can be further accelerated with high performance GPUs integrated within the future CAVs (validates Contribution 4).*

2) Comparison with state-of-the-art: We compare the end-to-end latency of COPILOT with the state-of-the-art proactive handoff approach proposed by Charan *et al.* [19] in Table III. We limit our comparison to this particular work because other techniques use different evaluation metrics or do not provide an end-to-end latency analysis [1], [15]. Charan *et al.* [19] propose proactive blockage prediction followed by contention-free random access that takes around $11.4ms$. After identifying a beam failure, the reactive approach takes $22.8ms$, which includes: (a) beam failure recovery [35], and (b) contention free random access [35]. Comparatively, COPILOT directly predicts the optimal RSU within $6.88ms$ with 90.91% accuracy, as mentioned in Observation 4.

TABLE III: Comparison of end-to-end latency of COPILOT with the state-of-the-art *(validates Contribution 4)*.

Papers	Strategies	Handoff Approaches	Evaluation Type	Latency (ms)	Prediction Accuracy
[36], [35]	Traditional	Reactive	Simulation	22.8	-
[19]	Deep learning	Proactive	Synthetic	11.4	**86%**
COPILOT	Deep learning	Proactive	Real world	**6.88**	**90.91%**

Observation 6. *COPILOT improves the latency for handoff mechanism by 69.8% and 39.64% than the state-of-the-art reactive and proactive approaches [19] (see Table. III, validates Contribution 4).*

D. Throughput Analysis

To highlight the effectiveness of our framework in terms of assured connectivity, we compare COPILOT framework to the conventional reactive handoff strategy similar to the state-of-the-art [19]. We design our experiments such that a handoff is triggered when the throughput drops below a threshold of $1Gbps$. The results for one episode are shown in Fig. 11 where there are two handoffs triggered along the path. In COPILOT, when the throughput is less than $1Gbps$, it is assigned to the predicted optimal RSU in 6.88 ms, whereas the reactive approach takes 22.8 ms for beam recovery and random access association [35]. We calculate the mean throughput for both these approaches in the sections where the performance differs and compute the performance improvement in COPILOT. We perform this experiment for all the samples in the selected subset of examples to obtain an empirical average of the throughput gain.

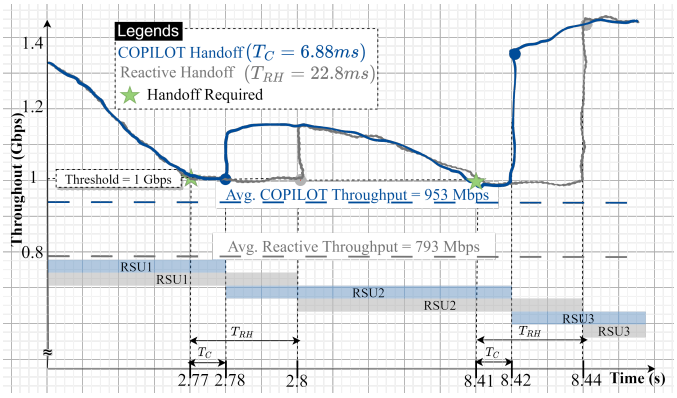

Fig. 11: Comparison of throughput variations for COPILOT based proactive handoff with the traditional reactive handoff for one episode of the COPILOT dataset. We observe the average throughput of first is 20.42% higher than the average throughput of the latter.

Observation 7. *COPILOT shows a throughput gain of 20.42% on an average compared to the reactive hand off approach. The performance gain of one such episode with two hand-offs is illustrated in Fig. 11 (validates Contribution 4)*

VII. Conclusions

In this paper, we propose COPILOT, which uses Lidar sensing data from multiple perceptions and leverages V2X communication to select the best RSU for mmWave communication with autonomous cars. COPILOT improves on single perception in an ego vehicle in the presence of transient blockages as well as performs RSU selection providing about 20.42% improved throughput than legacy handoffs within typical inner-city mobility conditions. The real-world dataset that we release and insights from experimental testing performed will enable new research on real autonomous cars, mmWave radios and RSUs in a urban settings. Our evaluations show that COPILOT outperforms the solutions with single perception by 14.11% accuracy. Moreover, COPILOT yields upto 69.8% and 39.64% improvement in latency compared to traditional reactive handoff in mmWave networks and state-of-the-art proactive handoff mechanisms, respectively. As a part of future work, more diverse channel scenarios can be evaluated to understand the impact of congested environments. The authors have provided public access to their code and data at [14].

Acknowledgement

The authors gratefully acknowledge the support from funding from the US National Science Foundation (grants CNS 2120447 and CNS 2112471).

REFERENCES

[1] Y. Koda, K. Nakashima, K. Yamamoto, T. Nishio, and M. Morikura, "Cooperative sensing in deep rl-based image-to-decision proactive handover for mmwave networks," in *2020 IEEE 17th Annual Consumer Communications Networking Conference (CCNC)*, 2020, pp. 1–6.

[2] "We believe in the future of connection in the metaverse," 2023, https://about.meta.com/metaverse/.

[3] "Introducing Apple Vision Pro: Apple's first spatial computer," 2023, https://www.apple.com/newsroom/2023/06/introducing-apple-vision-pro/.

[4] M. Harounabadi, D. M. Soleymani, S. Bhadauria, M. Leyh, and E. Roth-Mandutz, "V2x in 3gpp standardization: Nr sidelink in release-16 and beyond," *IEEE Communications Standards Magazine*, vol. 5, no. 1, pp. 12–21, 2021.

[5] M. Alrabeiah, A. Hredzak, and A. Alkhateeb, "Millimeter Wave Base Stations with Cameras: Vision-Aided Beam and Blockage Prediction," in *IEEE 91st Vehicular Technology Conference (VTC2020)*. IEEE, 2020, pp. 1–5.

[6] 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 Transactions on Vehicular Technology*, vol. 71, no. 7, pp. 7639–7655, 2022.

[7] T. Nishio, H. Okamoto, K. Nakashima, Y. Koda, K. Yamamoto, M. Morikura, Y. Asai, and R. Miyatake, "Proactive received power prediction using machine learning and depth images for mmwave networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 11, pp. 2413–2427, 2019.

[8] 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 Transactions on Cognitive Communications and Networking*, vol. 6, no. 2, pp. 802–816, 2020.

[9] J. Cui, H. Qiu, D. Chen, P. Stone, and Y. Zhu, "Coopernaut: End-to-end driving with cooperative perception for networked vehicles," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[10] E. Arnold, M. Dianati, R. de Temple, and S. Fallah, "Cooperative perception for 3d object detection in driving scenarios using infrastructure sensors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 1852–1864, 2022.

[11] "C-V2X Drives Intelligent Transportation," 2023, https://www.qualcomm.com/content/dam/qcomm-martech/dm-assets/documents/cv2x_white_paper-final_0.pdf.

[12] J. Cui, H. Qiu, D. Chen, P. Stone, and Y. Zhu, "Coopernaut: End-to-end driving with cooperative perception for networked vehicles," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 17 252–17 262.

[13] R. Xu, X. Xia, J. Li, H. Li, S. Zhang, Z. Tu, Z. Meng, H. Xiang, X. Dong, R. Song, H. Yu, B. Zhou, and J. Ma, "V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 13 712–13 722.

[14] "COPILOT Dataset," https://genesys-lab.org/copilot.

[15] J. Palacios, P. Casari, H. Assasa, and J. Widmer, "Leap: Location estimation and predictive handover with consumer-grade mmwave devices," in *IEEE Conference on Computer Communications (INFOCOM)*, 2019.

[16] M. Polese, M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, "Improved handover through dual connectivity in 5g mmwave mobile networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 9, pp. 2069–2084, 2017.

[17] M. Mezzavilla, S. Goyal, S. Panwar, S. Rangan, and M. Zorzi, "An mdp model for optimal handover decisions in mmwave cellular networks," in *2016 European Conference on Networks and Communications (EuCNC)*, 2016, pp. 100–105.

[18] Y. Sun, G. Feng, S. Qin, Y.-C. Liang, and T.-S. P. Yum, "The smart handoff policy for millimeter wave heterogeneous cellular networks," *IEEE Transactions on Mobile Computing*, vol. 17, no. 6, pp. 1456–1468, 2018.

[19] G. Charan, M. Alrabeiah, and A. Alkhateeb, "Vision-aided 6g wireless communications: Blockage prediction and proactive handoff," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10 193–10 208, 2021.

[20] Y.-C. Liu, J. Tian, C.-Y. Ma, N. Glaser, C.-W. Kuo, and Z. Kira, "Who2com: Collaborative perception via learnable handshake communication," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6876–6883.

[21] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[22] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[23] A. Alkhateeb, G. Charan, T. Osman, A. Hredzak, J. Morais, U. Demirhan, and N. Srinivas, "Deepsense 6g: A large-scale real-world multi-modal sensing and communication dataset," *IEEE Communications Magazine*, pp. 1–7, 2023.

[24] B. Salehi, J. Gu, D. Roy, and K. Chowdhury, "Flash: Federated learning for automated selection of high-band mmwave sectors," in *IEEE Conference on Computer Communications (INFOCOM)*, 2022, pp. 1719–1728.

[25] D. Qiao and F. Zulkernine, "Adaptive feature fusion for cooperative perception using lidar point clouds," in *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023, pp. 1186–1195.

[26] Q. Chen, X. Ma, S. Tang, J. Guo, Q. Yang, and S. Fu, "F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3d point clouds," in *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, 2019, pp. 88–100.

[27] X. Zhu, D. Cheng, Z. Zhang, S. Lin, and J. Dai, "An empirical study of spatial attention mechanisms in deep networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[28] D. Steinmetzer, D. Wegemer, M. Schulz, J. Widmer, and M. Hollick, "Compressive Millimeter-Wave Sector Selection in Off-the-Shelf IEEE 802.11ad Devices," *International Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, 2017.

[29] J. Palacios, D. Steinmetzer, A. Loch, M. Hollick, and J. Widmer, "Adaptive Codebook Optimization for Beam Training on Off-the-Shelf IEEE 802.11ad Devices," *International Conference on Mobile Computing and Networking (MobiCom)*, 2018.

[30] J.-M. Perez-Rua, V. Vielzeuf, S. Pateux, M. Baccouche, and F. Jurie, "MFAS: Multimodal Fusion Architecture Search," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[31] N. Shvetsova, B. Chen, A. Rouditchenko, S. Thomas, B. Kingsbury, R. S. Feris, D. Harwath, J. Glass, and H. Kuehne, "Everything at once - multi-modal fusion transformer for video retrieval," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 20 020–20 029.

[32] N. Sarafianos, T. Giannakopoulos, C. Nikou, and I. A. Kakadiaris, "Curriculum learning for multi-task classification of visual attributes," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017.

[33] "Highway Statistics," 2018, https://www.fhwa.dot.gov/policyinformation/statistics/2015/hm10m.cfm.

[34] "State Motor-Vehicle Registrations," 2018, https://www.fhwa.dot.gov/policyinformation/statistics/2015/mv1.cfm.

[35] J. Thota and A. Aijaz, "On performance evaluation of random access enhancements for 5g urllc," in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, 2019, pp. 1–7.

[36] M. Giordani, M. Mezzavilla, and M. Zorzi, "Initial access in 5g mmwave cellular networks," *IEEE Communications Magazine*, vol. 54, no. 11, pp. 40–47, 2016.