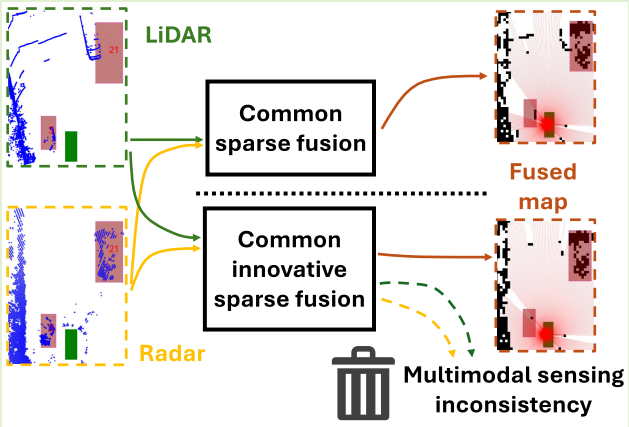
Spatial Sparsity-Aware Radar-LiDAR Fusion for Occupancy Grid Mapping in Automotive Driving

Peiyuan Zhai, *Graduate Student Member, IEEE*, Geethu Joseph, *Senior Member, IEEE*, Nitin Jonathan Myers, *Member, IEEE*, Çağan Önen, *Member, IEEE*, and Ashish Pandharipande, *Senior Member, IEEE*

***Abstract*—We address the problem of estimating a binary occupancy grid map by fusing point cloud data from radar and LiDAR sensors for automotive driving perception. To achieve this, we introduce two measurement models for fusion and formulate occupancy mapping as sparse vector reconstruction from the set of radar and LiDAR measurements. The first model, called common sparse fusion, jointly estimates a common map from all sensor measurements. The second model, called common innovative sparse fusion, assumes a shared map and an innovation component (error collector) for each sensor modality's measurements. This approach enhances the robustness of occupancy map estimation against potential sensor mismatch and calibration errors, and inconsistencies between the two modalities. We use the pattern-coupled sparse Bayesian learning (PCSBL) algorithm to recover maps, leveraging the inherent sparsity and spatial dependencies in automotive occupancy maps. Numerical experiments on the public RADIATE dataset show that our feature-level fusion models outperform single-modality mapping and decision-level fusion models in detecting drivable areas and targets. Furthermore, statistical results with corrupted LiDAR data establish that our common innovative sparse fusion model is robust against unreliable sensor data.**



***Index Terms*— Multimodal sensing, pattern-coupled sparse Bayesian learning (PCSBL), point cloud measurement fusion, RADIATE dataset, sparse fusion models.**

I. Introduction

IN ASSISTED and autonomous driving, occupancy grid mapping (OGM) is used to generate a representation of the vehicle's surroundings. These maps are essential for subsequent driving tasks, such as object tracking and path planning [1], [2]. To construct these maps, environmental data are acquired through on-board sensors such as LiDAR and radar. We address the sensor fusion problem for OGM, focusing on combining point cloud data from LiDAR and radar sensors to improve the detection of targets and identify drivable areas compared to single-modality mapping and decision-level fusion of maps. To establish the context and background of our work, we start with a literature review on OGM algorithms.

A. Related Work

State-of-the-art occupancy mapping algorithms focus either on single sensing modalities or high-level fusion of maps generated by different sensing modalities. A common single-modality occupancy mapping algorithm is the inverse sensor model, where the occupancy probability of each cell is updated recursively based on measurements [3]. The algorithm has also been extended to multimodal sensor cases, where data are fused at the feature or decision level [4], [5]. However, this algorithm often struggles with conflicting measurements and does not adequately explore the spatial relationship between occupancy grid cells. Another approach for OGM is the kernel-based Gaussian process OGM [6] that reconstructs the occupancy map by learning a Gaussian process classification model in the mapping area. This approach

Received 15 April 2025; revised 12 July 2025; accepted 17 July 2025. Date of publication 29 July 2025; date of current version 3 September 2025. The associate editor coordinating the review of this article and approving it for publication was Dr. Tao Tang. *(Corresponding author: Geethu Joseph.)*

Peiyuan Zhai and Geethu Joseph are with the Signal Processing Systems Group, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: p.zhai@tudelft.nl; g.joseph@tudelft.nl).

Nitin Jonathan Myers is with Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: N.J.Myers@tudelft.nl).

Çağan Önen and Ashish Pandharipande are with NXP Semiconductors, 5656 AG Eindhoven, The Netherlands (e-mail: cagan.onen_1@nxp.com; ashish.pandharipande@nxp.com).

Digital Object Identifier 10.1109/JSEN.2025.3592023

captures spatial correlations in the map, albeit with high computational complexity. In [7], a Gaussian process implicit surface problem involving multimodal sensor data is studied. This work combines data from LiDAR and radar using either feature-level fusion, which merges training data points, or decision-level fusion, which combines multiple Gaussian processes. Nevertheless, kernel-based methods only exploit the spatial dependencies in the map but cannot account for the underlying sparsity of the occupancy status values of the map. Furthermore, in typical automotive applications, occupancy grid maps are sparse because the obstacles in the field of view (FOV) constitute small continuous areas of the entire environment and are measured at their boundaries. The sparsity-aware OGM model is introduced in [8], where a block sparse occupancy map is estimated using the pattern-coupled sparse Bayesian learning (PCSBL) algorithm [9]. However, the algorithm in [8] is limited to single-modality sensors, either LiDAR or radar, and does not address issues of data unreliability or calibration errors. In [10], camera information is extracted as prior information on occupancy status and fused with LiDAR or radar data in the sparsity-aware OGM model. However, this work does not consider the fusion of LiDAR and radar data. In contrast, in this article, we introduce a sparsity-aware model-based OGM framework that fuses LiDAR and radar data, leveraging the complementary characteristics of both sensing modalities.

Besides model-based algorithms, deep learning-based algorithms have also been explored to solve the OGM problem. Using large scale datasets [11], [12], deep learning-based models can be trained to infer occupancy grid maps with classification or semantic information, either with single-modality [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23] or multimodal sensors [24], [25], [26], [27], [28]. Deep learning models, however, require large annotated datasets, struggle to generalize, and often need retraining when datasets or sensor types change. In addition, they often fail to generate occupancy maps of varying sizes and resolutions, estimate confidence in the estimated maps, or handle sensor unreliability. Some works address these issues partially. For example, multiscale maps are generated using coarse-to-fine query structures [29], [30], [31] but require full-size feature computation. Other approaches estimate uncertainty alongside occupancy mapping [13], [22], [32]. Also, some works explore LiDAR-radar fusion [24], [27], [28], assuming that radar is reliable. To the best of our knowledge, no approach provides a unified framework for robust multimodal fusion while accounting for arbitrary-area and uncertainty-aware mapping. In this work, we develop a physics-informed sensor fusion that generalizes across LiDAR and radar data without training or annotated data.

B. Contributions

In this work, we consider OGM problem using multimodal sensor data from LiDAR and radar sensors. We account for the underlying sparsity and spatial dependencies of the occupancy grid maps while ensuring robustness against unreliable sensor data. To effectively utilize information from diverse sensors, we introduce feature-level fusion algorithms rather than high-level fusion [33]. Our main contributions are given as follows.

1) *Fusion Models:* We present two feature-level LiDAR-radar sensor fusion models for OGM, namely, common sparse and common-innovation sparse (CIS) fusion models. Both models rely on features extracted from the sensor data in the form of linear measurements of the occupancy status obtained from the point cloud data. While the common sparse model directly combines the linear measurements from the two sensors, the CIS model further includes innovation components for each sensor modality to account for sensor inconsistencies across different modalities and unreliable sensor data.
2) *Sparsity-Aware OGM Algorithms:* We devise PCSBL-based algorithms for the two OGM fusion models that can recover sparse occupancy grid maps while exploiting spatial dependencies. The common sparse model uses PCSBL under possibly different noise variances at the radar and LiDAR. The CIS fusion model adapts PCSBL with an extended prior to account for the sparsity of the innovation components.
3) *Real-World Dataset-Based Validation:* We evaluate our algorithm using the public real-world RADIATE dataset [34] and compare our algorithms with single-modality sparse OGM [8] and two high-level fusion algorithms based on single-modality maps. The results show that our algorithms outperform existing OGM methods. Particularly, in high-disparity scenarios, our common sparse and CIS fusion achieve high detection rates while enhancing drivable area detection by 9%–33% and 15%–37%, respectively, compared to the baseline models.

In summary, we introduce a multimodal fusion OGM approach that employs flexible sparsity models for feature-level fusion and utilizes Bayesian sparse recovery methods to adaptively handle unreliable sensor data.

We advance our earlier work [35] in several directions. We provide a detailed derivation of the newly developed PCSBL-based algorithms. We also introduce a new benchmarking algorithm based on the Bayesian fusion of single-modality occupancy probability maps obtained via PCSBL [8], making our performance evaluation more comprehensive. Furthermore, we present statistical results with corrupted LiDAR data across various scenes from the dataset, demonstrating the superior performance of the fusion models. They also show that our CIS model dynamically adjusts to unreliable sensor data, thereby improving accuracy and robustness against sensor errors.

The organization of the article is given as follows. Section II presents two fusion models: common sparse and common innovative sparse fusion. Subsequently, Sections III and IV introduce two PCSBL-based OGM algorithms designed for each fusion model, respectively. Section V evaluates these models through a series of experiments using real-world datasets, demonstrating their enhanced performance and robustness against multimodal sensor disparities. Finally, Section VI presents concluding remarks.

II. Sensor Data Processing and Models

This section introduces the OGM problem, outlining the required sensor data processing steps and the OGM models for radar and LiDAR measurements.

A. Sensor Data Preprocessing

The first preprocessing step removes clutter from LiDAR and radar data. LiDAR data consist of point cloud measurements with 3-D coordinates and reflection intensity values, where each point represents a potential obstacle at that location. The clutter removal for LiDAR aims to remove ground scatter that can mislead the mapping algorithm into detecting drivable areas as obstacles. Although complex adaptive ground segmentation algorithms are available in the literature [36], we remove ground points using height thresholding to reduce computation. The radar data are typically available as high-resolution 360° range–azimuth maps. Clutter removal uses adaptive threshold filtering, such as the constant false alarm rate detector, to extract stable target detection points [37].

After the removal of clutter, points outside the mapping area are excluded from both LiDAR and radar point cloud measurements to reduce computation [8]. Let M_{L} and M_{R} be the number of LiDAR and radar point cloud measurements after preprocessing, respectively. We next explain the measurement models for the LiDAR and radar measurements.

B. Single-Modality Linear Measurement Models

To introduce the model, we first divide the area of interest into N cells and present a linear measurement model that relates the point cloud to the unknown occupancy status of cells. In this model, each sensor measurement is mapped to two linear measurements, capturing possible occupied locations and free space along the trajectory of the sensor beam [8]. To identify free and occupied cells, each LiDAR measurement uses a thin beam connecting the ego vehicle and the reflection point, implemented using Bresenham's line algorithm [38]. The cells along the beam's path are marked as free, while the cell containing the reflection point is considered occupied. In contrast, radar, due to its relatively lower angular resolution, uses a circular sector with its vertex at the ego vehicle and the detected cell as the endpoint. Cells within a specified angle and distance around the reflection point are classified as occupied, while the remaining cells within the sector are considered free. For more details and an example of free and occupied cells for the LiDAR and radar sensor models, we refer to [8].

The mth measurement point of a sensor (LiDAR or radar) is related to the occupancy probability map $\boldsymbol{x}_{\mathrm{sensor}} \in [0,1]^N$ as follows:

$$\begin{bmatrix} y_{\mathrm{free}} \\ y_{\mathrm{occ}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{a}_{2m-1}^{\mathrm{T}} \\ \boldsymbol{a}_{2m}^{\mathrm{T}} \end{bmatrix} \boldsymbol{x}_{\mathrm{sensor}}$$

where $\boldsymbol{a}_{2m-1}^{\mathrm{T}}$ and $\boldsymbol{a}_{2m}^{\mathrm{T}}$ are the $(2m-1)$th and $2m$th rows of the measurement matrix $\boldsymbol{A}_{\mathrm{sensor}}$, with 1s indicating free and occupied cells, respectively. Then, $\boldsymbol{a}_{2m-1}^{\mathrm{T}}\boldsymbol{x}_{\mathrm{sensor}}$ and $\boldsymbol{a}_{2m}^{\mathrm{T}}\boldsymbol{x}_{\mathrm{sensor}}$ are the sum of probabilities of free and occupied cells corresponding to the mth measurement, respectively, mapped to y_{free} (a small value close to zero) and y_{occ} (large value). Thus, the M point cloud measurements can be used to construct $2M$ linear equations, given by $\boldsymbol{y}_{\mathrm{sensor}} = \boldsymbol{A}_{\mathrm{sensor}}\boldsymbol{x}_{\mathrm{sensor}}$.

With the above sensor model, the single-modality linear measurement models for LiDAR and radar can be written as follows:

$$\boldsymbol{y}_{\mathrm{L}} = \boldsymbol{A}_{\mathrm{L}}\boldsymbol{x}_{\mathrm{L}} + \boldsymbol{n}_{\mathrm{L}} \tag{1}$$

$$\boldsymbol{y}_{\mathrm{R}} = \boldsymbol{A}_{\mathrm{R}}\boldsymbol{x}_{\mathrm{R}} + \boldsymbol{n}_{\mathrm{R}} \tag{2}$$

where $\boldsymbol{x}_{\mathrm{L}} \in [0,1]^N$ is the vectorized LiDAR occupancy probability map representing the occupancy probabilities of N cells and $\boldsymbol{A}_{\mathrm{L}} \in \{0,1\}^{2M_{\mathrm{L}}\times N}$ is the measurement matrix for LiDAR. Similarly, $\boldsymbol{x}_{\mathrm{R}} \in [0,1]^N$ is the vectorized radar map, and $\boldsymbol{A}_{\mathrm{R}} \in \{0,1\}^{2M_{\mathrm{R}}\times N}$ is its measurement matrix. In addition, $\boldsymbol{n}_{\mathrm{L}} \in \mathbb{R}^{2M_{\mathrm{L}}}$ and $\boldsymbol{n}_{\mathrm{R}} \in \mathbb{R}^{2M_{\mathrm{R}}}$ are the model discrepancies in single-modality LiDAR and radar sensors, respectively. Building on this model, we formulate our fusion model next.

C. Sensor Fusion Models and OGM Problem Formulation

In a static environment or a short frame of dynamic environment, given the LiDAR and radar measurements $\boldsymbol{y}_{\mathrm{L}}$ and $\boldsymbol{y}_{\mathrm{R}}$ and the measurement matrices $\boldsymbol{A}_{\mathrm{L}}$ and $\boldsymbol{A}_{\mathrm{R}}$, our goal is to recover an unknown occupancy probability map $\boldsymbol{x} \in [0,1]^N$. This requires new fusion models that can effectively combine and explain the linear measurements from both sensors. We introduce two fusion models to address this problem, namely, the common sparse fusion model and the CIS fusion model.

The common sparse fusion model assumes that the single-modality maps $\boldsymbol{x}_{\mathrm{L}}$ and $\boldsymbol{x}_{\mathrm{R}}$ are identical to a common map $\boldsymbol{x}$, i.e., $\boldsymbol{x}_{\mathrm{L}} = \boldsymbol{x}_{\mathrm{R}} = \boldsymbol{x}$. This leads to a new model that concatenates the LiDAR and radar measurements as follows:

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_{\mathrm{L}} \\ \boldsymbol{y}_{\mathrm{R}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_{\mathrm{L}} \\ \boldsymbol{A}_{\mathrm{R}} \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} \boldsymbol{n}_{\mathrm{L}} \\ \boldsymbol{n}_{\mathrm{R}} \end{bmatrix} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n} \tag{3}$$

where the measurements $\boldsymbol{y}_{\mathrm{L}}$ and $\boldsymbol{y}_{\mathrm{R}}$, and measurement matrices $\boldsymbol{A}_{\mathrm{L}}$ and $\boldsymbol{A}_{\mathrm{R}}$ are defined as in (1) and (2) for LiDAR and radar, respectively. Also, $\boldsymbol{y} = \left[\boldsymbol{y}_{\mathrm{L}}^{\top}\ \boldsymbol{y}_{\mathrm{R}}^{\top}\right]^{\top}$, $\boldsymbol{A} = \left[\boldsymbol{A}_{\mathrm{L}}^{\top}\ \boldsymbol{A}_{\mathrm{R}}^{\top}\right]^{\top}$, and $\boldsymbol{n} = \left[\boldsymbol{n}_{\mathrm{L}}^{\top}\ \boldsymbol{n}_{\mathrm{R}}^{\top}\right]^{\top}$. Therefore, the OGM problem using the common sparse model is equivalent to estimating $\boldsymbol{x}$ from $\boldsymbol{y}$ and $\boldsymbol{A}$ in (3). Jointly using the measurements, the model can effectively resolve conflicts due to inconsistent measurements to get a better map than single-modality maps.

We note that the sparse fusion approach makes a strong assumption that both LiDAR and radar measurements correspond to all obstacles in the environment. However, radar may miss certain obstacles that are detected by LiDAR, and vice versa. In addition, conflicts can arise between the two sensor measurements, particularly when one of the sensors is unreliable. For example, poor performance can arise from adverse weather conditions (especially for LiDAR) or due to sensor misalignment. Therefore, our second model, the CIS fusion model, softly fuses LiDAR and radar measurements based on the reliability of the sensor. To accommodate any possible errors in the single-modality maps, we assume that

$$\boldsymbol{x}_{\mathrm{L}} = \boldsymbol{x}_{\mathrm{c}} + \boldsymbol{x}_{\Delta\mathrm{L}}, \quad \text{and} \quad \boldsymbol{x}_{\mathrm{R}} = \boldsymbol{x}_{\mathrm{c}} + \boldsymbol{x}_{\Delta\mathrm{R}} \tag{4}$$

where $\boldsymbol{x}_{\mathrm{c}}$ is the common part signal assumed to be the true underlying occupancy grid map, while $\boldsymbol{x}_{\Delta\mathrm{L}}$ and $\boldsymbol{x}_{\Delta\mathrm{R}}$ are the sparse error collectors for LiDAR and radar, respectively.

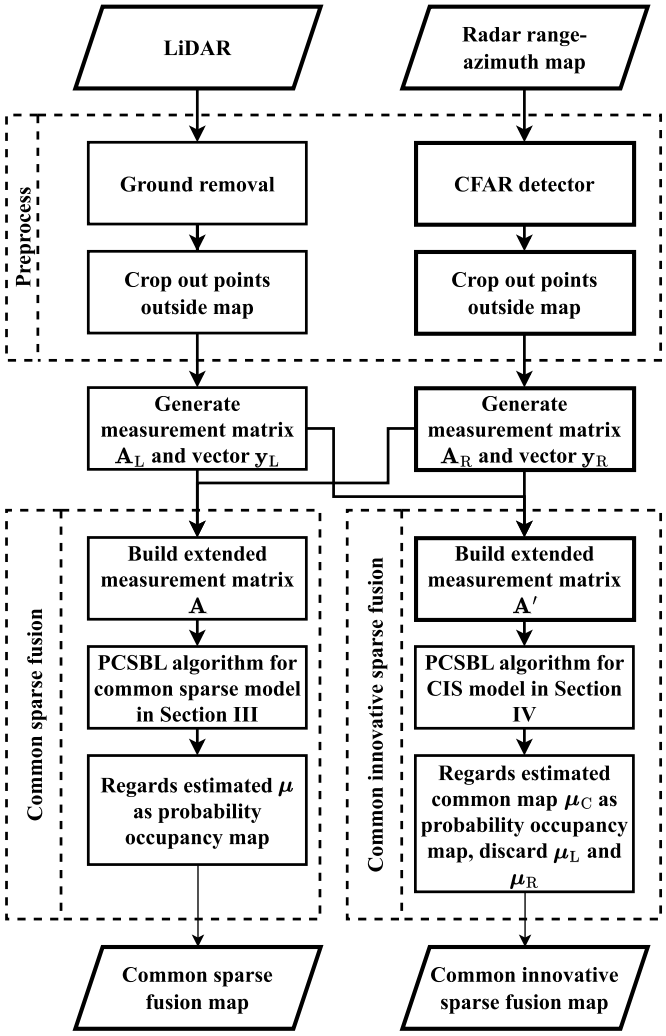


Fig. 1. Signal processing pipeline of the two fusion approaches.

The error collectors account for possible mismatches and unreliability of the sensors. Substituting (4) into (1) and (2)

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_L \\ \boldsymbol{y}_R \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_L & \boldsymbol{A}_L & \boldsymbol{0} \\ \boldsymbol{A}_R & \boldsymbol{0} & \boldsymbol{A}_R \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_c \\ \boldsymbol{x}_{\Delta L} \\ \boldsymbol{x}_{\Delta R} \end{bmatrix} + \begin{bmatrix} \boldsymbol{n}_L \\ \boldsymbol{n}_R \end{bmatrix} = \boldsymbol{A}'\boldsymbol{x}' + \boldsymbol{n} \tag{5}$$

where $\boldsymbol{A}' \in \mathbb{R}^{2(M_L+M_R)\times 3N}$ is the extended measurement matrix and $\boldsymbol{x}' = \left[\boldsymbol{x}_c^\top \; \boldsymbol{x}_{\Delta L}^\top \; \boldsymbol{x}_{\Delta R}^\top\right]^\top \in \mathbb{R}^{3N}$ is the extended map including the error collector components. Therefore, the OGM problem using the CIS fusion model is to estimate $\boldsymbol{x}_c$ from (5).

In certain scenarios, some part of the map is not necessary to be estimated, such as the known nondrivable area (buildings or vegetation) from digital maps [8]. The proposed fusion models can be extended to account for this case by simply removing the corresponding indices cells and columns from the unknown map vector $\boldsymbol{x}$ or $\boldsymbol{x}'$ and the measurement matrix $\boldsymbol{A}$ or $\boldsymbol{A}'$, respectively. Furthermore, in the OGM problem, the underlying map, i.e., $\boldsymbol{x}$ for the common sparse fusion model and $\boldsymbol{x}_c$ for the CIS fusion model, is assumed to be a block sparse signal with the unknown sparsity level and block structure [8]. This assumption comes from the observation that obstacles usually occupy small continuous portions of the environment and are likely detected at their boundaries. Therefore, we need to devise an OGM algorithm that can exploit the block sparse nature of the underlying occupancy probability map. Once the occupancy probability map $\hat{\boldsymbol{x}}$ is estimated, using either of the two models, the binary occupancy grid map $\hat{\boldsymbol{x}}_b$ can be obtained by simple thresholding, i.e., $\hat{\boldsymbol{x}}_b = \mathbf{1}_{\hat{\boldsymbol{x}}>\tau}$, where $\mathbf{1}_{\cdot}$ is an elementwise indicator function, which returns 1 if an element of $\hat{\boldsymbol{x}}$ is larger than a predefined threshold τ and 0 otherwise. Since $\hat{\boldsymbol{x}}$ is block sparse, the resulting binary map $\hat{\boldsymbol{x}}_b$ also exhibits a block sparse structure. Hence, we use block sparsity-aware occupancy mapping algorithms, based on PCSBL, for the common sparse and CIS fusion models. The signal processing pipeline of our approach including sensor data preprocessing and fusion algorithms for common sparse fusion and CIS fusion is shown in Fig. 1. Sections III and IV derive the sparse recovery algorithms.

III. PCSBL-Based OGM Algorithm for Common Sparse Fusion Model

We solve the occupancy probability map from the common model fusion model in (3) using a modified version of the PCSBL algorithm [9]. The PCSBL framework inherently accommodates the unknown block-sparse nature of the map $\boldsymbol{x}$, making it a suitable choice for our algorithm development.

To derive the OGM algorithm, the PCSBL framework imposes a hypothetical two-layer hierarchical prior on the unknown map $\boldsymbol{x}$. The first layer is a zero mean Gaussian prior with unknown hyperparameter $\boldsymbol{\alpha} \in \mathbb{R}^N$

$$p(\boldsymbol{x}|\boldsymbol{\alpha}) = \prod_{n=1}^{N} \mathcal{N}\left(x[n]|0, (\delta[n])^{-1}\right) \tag{6}$$

where the precision value of the nth cell

$$\delta[n] = \alpha[n] + \beta \sum_{m\in\mathbb{N}_n} \alpha[m] \tag{7}$$

depends on the nth hyperparameter $\alpha[n]$ and hyperparameters of its direct neighbors denoted by set $\mathbb{N}_n$. For the nth cell, the $\delta[n]$ value close to zero leads to a large variance, thus allowing $x[n]$ to deviate from the zero mean, indicating an occupied grid. On the contrary, a large $\delta[n]$ value leads to a small variance, thus enforcing $x[n]$ to be zero, indicating a free grid. The positions of the nonzero entries of $\boldsymbol{x}$ are decided by the precision $\delta[n]$. Also, $\beta \geq 0$ is a hyperparameter that controls how much the neighbors influence the precision of the cell. When $\beta = 0$, the existence of a nonzero value at the nth cell is independent of the neighbors, which is equivalent to the classical sparse Bayesian learning model [39] where the block structure is not considered. A positive value of β enforces the spatial correlation via the shared hyperparameters within the prior distribution in (6). Specifically, if $\alpha[n]$ is large, the precision $\delta[m]$ of the neighboring cells surrounding the nth cell also increases, driving the corresponding $x[m]$ values to zero. This coupled structure promotes block sparsity in $\boldsymbol{x}$, imposing a prior that encourages multiple adjacent occupied cells within the same neighborhood. The second layer is a Gamma hyperprior model with

parameters a and b

$$p(\boldsymbol{\alpha}) = \prod_{n=1}^{N} \text{Gamma}(\alpha[n]|a,b) = \prod_{n=1}^{N} \Gamma(a)^{-1} b^{a} (\alpha[n])^{a-1} e^{-b\alpha[n]} \tag{8}$$

where the gamma function $\Gamma(a) = \int_0^\infty t^{a-1}e^{-t}dt$. A larger hyperparameter a increases the likelihood of nonzero $\alpha[n]$ values, which, in turn, leads to larger $\delta[n]$ values and smaller corresponding entries in $x[n]$ and vice versa.

Furthermore, the model discrepancies are assumed to follow independent Gaussian distributions, i.e., $\boldsymbol{n}_{\text{L}} \sim \mathcal{N}(\boldsymbol{n}_{\text{L}}|\boldsymbol{0}, \sigma_{\text{L}}^2 \boldsymbol{I}_{M_{\text{L}}})$ and $\boldsymbol{n}_{\text{R}} \sim \mathcal{N}(\boldsymbol{n}_{\text{R}}|\boldsymbol{0}, \sigma_{\text{R}}^2 \boldsymbol{I}_{M_{\text{R}}})$. We choose different variances for the two sensing modalities to account for their distinct characteristics. The variances of measurement noise σ_{L}^2 and σ_{R}^2 are unknown, and their inverse are assumed to follow Gamma prior distributions

$$\sigma_{\text{L}}^{-2} \sim \text{Gamma}\left(\sigma_{\text{L}}^{-2}|c,d\right) \tag{9}$$

$$\sigma_{\text{R}}^{-2} \sim \text{Gamma}\left(\sigma_{\text{R}}^{-2}|c,d\right) \tag{10}$$

where c and d are chosen to be small positive values, leading to a noninformative prior.

With the prior model assumed, an occupancy probability map $\boldsymbol{x}$ can be estimated by solving a type-II maximum a posterior estimation problem, i.e.,

$$\hat{\boldsymbol{x}} = \arg\max_{\boldsymbol{x}} p\left(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2\right). \tag{11}$$

Based on the linear measurement model, we have

$$p\left(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2\right) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{A}\boldsymbol{x},\boldsymbol{C})$$

where the noise covariance matrix is

$$\boldsymbol{C} = \begin{bmatrix} \sigma_{\text{L}}^2 \boldsymbol{I}_{2M_{\text{L}}} & \boldsymbol{0} \\ \boldsymbol{0} & \sigma_{\text{R}}^2 \boldsymbol{I}_{2M_{\text{R}}} \end{bmatrix}.$$

Furthermore, since $p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2)$ and $p(\boldsymbol{x}|\boldsymbol{\alpha})$ are assumed to be Gaussian, the posterior distribution $p(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2) \propto p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2)p(\boldsymbol{x}|\boldsymbol{\alpha})$ is also Gaussian. Its posterior mean and variance are given as follows:

$$\begin{aligned} \boldsymbol{\mu}_{\boldsymbol{x}} &= \boldsymbol{\Sigma}_{\boldsymbol{x}} \boldsymbol{A}^{\top} \boldsymbol{C}^{-1} \boldsymbol{y} \\ \boldsymbol{\Sigma}_{\boldsymbol{x}} &= \left(\boldsymbol{A}^{\top} \boldsymbol{C}^{-1} \boldsymbol{A} + \text{diag}(\boldsymbol{\delta})\right)^{-1} \end{aligned} \tag{12}$$

where $\boldsymbol{\delta}$ is the vector of precision values in (7). However, from (11) and (12), the map can be immediately obtained only if the hyperparameters $\boldsymbol{\alpha}$ and noise variances σ_{L}^2 and σ_{R}^2 are available. Their estimates can be obtained from the measurements by maximizing the posterior distribution

$$\left(\hat{\boldsymbol{\alpha}}, \hat{\sigma}_{\text{L}}^2, \hat{\sigma}_{\text{R}}^2\right) = \arg\max_{\boldsymbol{\alpha},\sigma_{\text{L}}^2\sigma_{\text{R}}^2} p\left(\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2|\boldsymbol{y}\right).$$

Since the closed-form solution of the above optimization problem is unavailable, one strategy is to use the expectation–maximization (EM) algorithm [8], [40] to infer the posterior distribution and estimate the hyperparameters recursively. The kth EM iteration is given by

$$\textbf{E-step: } Q\left(\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2\right) = \mathbb{E}\left\{\log\left[p\left(\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2|\boldsymbol{y},\boldsymbol{x}\right)\right]\right\}$$

$$\textbf{M-step: } \left(\boldsymbol{\alpha}^{(k+1)}, \left(\sigma_{\text{L}}^2\right)^{(k+1)}, \left(\sigma_{\text{R}}^2\right)^{(k+1)}\right) = \arg\max_{\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2} Q\left(\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2\right)$$

where the Q-function is the expected value of the log-likelihood function with respect to the posterior distribution of $\boldsymbol{x}$ given the measurements and the current estimates of the hyperparameters, i.e., $p(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\alpha}^{(k)},(\sigma_{\text{L}}^2)^{(k)},(\sigma_{\text{R}}^2)^{(k)})$.

Based on the Bayes rule and the fact that LiDAR and radar measurements are acquired independently with different systems, we derive

$$p\left(\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2|\boldsymbol{y},\boldsymbol{x}\right) \propto p(\boldsymbol{x}|\boldsymbol{\alpha})\,p(\boldsymbol{\alpha}) \times p\left(\boldsymbol{y}_{\text{L}}|\boldsymbol{x},\sigma_{\text{L}}^2\right) p\left(\sigma_{\text{L}}^2\right) p\left(\boldsymbol{y}_{\text{R}}|\boldsymbol{x},\sigma_{\text{R}}^2\right) p\left(\sigma_{\text{R}}^2\right)$$

which leads to a separable M-step optimization given by

$$\boldsymbol{\alpha}^{(k+1)} = \arg\max_{\boldsymbol{\alpha}} \mathbb{E}\left\{\log\left[p(\boldsymbol{x}|\boldsymbol{\alpha})\,p(\boldsymbol{\alpha})\right]\right\} \tag{13}$$

$$\left(\sigma_{\text{L}}^2\right)^{(k+1)} = \arg\max_{\sigma_{\text{L}}^2} \mathbb{E}\left\{\log\left[p\left(\boldsymbol{y}_{\text{L}}|\boldsymbol{x},\sigma_{\text{L}}^2\right) p\left(\sigma_{\text{L}}^2\right)\right]\right\} \tag{14}$$

$$\left(\sigma_{\text{R}}^2\right)^{(k+1)} = \arg\max_{\sigma_{\text{R}}^2} \mathbb{E}\left\{\log\left[p\left(\boldsymbol{y}_{\text{R}}|\boldsymbol{x},\sigma_{\text{R}}^2\right) p\left(\sigma_{\text{R}}^2\right)\right]\right\}. \tag{15}$$

For the optimization problem in (13), the closed-form update equation is not available since the hyperparameters are entangled together. A suboptimal solution $\boldsymbol{\alpha}_{\star}$ can be obtained by setting first derivative $(\partial\, Q(\boldsymbol{\alpha})/\partial\, \boldsymbol{\alpha}) = \boldsymbol{0}$, as discussed in [9]. The optimal solutions of (14) and (15) are available in closed form [9], and the update equations corresponding to (13)–(15) are given by

$$\alpha^{(k+1)}[n] = \frac{2a}{w^{(k)}[n] + \sum_{m\in\mathbb{N}_n} w^{(k)}[m] + 2b} \tag{16}$$

$$\left(\sigma_{\text{L}}^2\right)^{(k+1)} = \frac{2d + \left\|\boldsymbol{y}_{\text{L}} - \boldsymbol{A}_{\text{L}}\boldsymbol{x}\right\|_2^2 + \text{tr}\left(\boldsymbol{A}_{\text{L}}^{\top}\boldsymbol{A}_{\text{L}}\boldsymbol{\Sigma}_x^{(k)}\right)}{2c + 2M_{\text{L}}} \tag{17}$$

$$\left(\sigma_{\text{R}}^2\right)^{(k+1)} = \frac{2d + \left\|\boldsymbol{y}_{\text{R}} - \boldsymbol{A}_{\text{R}}\boldsymbol{x}\right\|_2^2 + \text{tr}\left(\boldsymbol{A}_{\text{R}}^{\top}\boldsymbol{A}_{\text{R}}\boldsymbol{\Sigma}_x^{(k)}\right)}{2c + 2M_{\text{R}}} \tag{18}$$

where $\boldsymbol{\mu}^{(k)}$ and $\boldsymbol{\Sigma}_x^{(k)}$ are given by (12) with $(\boldsymbol{\alpha},\sigma_{\text{L}}^2,\sigma_{\text{R}}^2) = (\boldsymbol{\alpha}^{(k)},(\sigma_{\text{L}}^2)^{(k)},(\sigma_{\text{R}}^2)^{(k)})$, and we define $w^{(k)}[n] = (\mu_x^{(k)}[n])^2 + \Sigma_x^{(k)}[n,n]$. Also, $\mu_x^{(k)}[n]$ and $\Sigma_x^{(k)}[n,n]$ represent the nth entry and the nth diagonal element of $\boldsymbol{\mu}^{(k)}$ and $\boldsymbol{\Sigma}_x^{(k)}$, respectively. The complete algorithm for the common sparse fusion model is summarized in Algorithm 1.

IV. PCSBL-Based OGM Algorithm for CIS Fusion Model

To develop the PCSBL algorithm for the CIS model in (5), we adopt the following prior model. We recall that the common part signal $\boldsymbol{x}_{\text{c}}$ is designed to gather the sparse supports of the true underlying map that contains the obstacles.

Algorithm 1 Common Sparse LiDAR-Radar Fusion OGM

Input: M_{L} LiDAR points, M_{R} radar points, N grid cells, maximum number of iterations K, map threshold τ

Output: Occupancy map $\hat{\boldsymbol{x}}$

Parameters: a, b

Generate $\boldsymbol{y}_{\mathrm{L}}$, $\boldsymbol{y}_{\mathrm{R}}$ and $\boldsymbol{A}_{\mathrm{L}}$, $\boldsymbol{A}_{\mathrm{R}}$ based on point cloud using (1) and (2)

Set noise hyperparameters $c, d = 1 \times 10^{-4}$

Initialize $k = 0$, $\boldsymbol{\alpha}^{(0)} = \mathbf{0}$, $(\sigma_{\mathrm{L}}^2)^{(0)} = (\sigma_{\mathrm{R}}^2)^{(0)} = 0.5$

repeat

  Compute $\boldsymbol{\mu}_{\boldsymbol{x}}^{(k)}$ and $\boldsymbol{\Sigma}_{\boldsymbol{x}}^{(k)}$ according to (7) and (12)

  Update $\boldsymbol{\alpha}^{(k+1)}$ according to (16)

  Update $(\sigma_{\mathrm{L}}^2)^{(k+1)}$ and $(\sigma_{\mathrm{R}}^2)^{(k+1)}$ as per (17) and (18)

  Set $k \leftarrow k+1$

until convergence or $k > K$

Set $\boldsymbol{\mu}^{(k-1)}$ as the estimated probabilistic map $\hat{\boldsymbol{x}}$

Obtain the binary map $\hat{\boldsymbol{x}}$ by thresholding $\hat{\boldsymbol{x}}_{\mathrm{b}} = \mathbf{1}_{\hat{\boldsymbol{x}}>\tau}$

Therefore, it is assumed to be a block sparse vector following the hierarchical prior distribution of the PCSBL model, i.e., described by (6), with $\boldsymbol{\alpha} = \boldsymbol{\alpha}_{\mathrm{c}}$, and (8), with hyperparameter $a = a_{\mathrm{c}}$ and $b = b_{\mathrm{c}}$. On the contrary, the error collectors are expected to capture false positives that are not necessarily spatially correlated and, thus, are assumed to have a general sparse structure. Setting $\beta = 0$ in (7) eliminates the spatial correlation, and the prior model is equivalent to the general sparse Bayesian learning model [39]. Therefore, we define hyperparameter $\boldsymbol{\alpha}' = \left[\boldsymbol{\alpha}_{\mathrm{c}}^{\top}\ \boldsymbol{\alpha}_{\Delta\mathrm{L}}^{\top}\ \boldsymbol{\alpha}_{\Delta\mathrm{R}}^{\top}\right]^{\top}$, using which the first layer of the CIS prior model is given by

$$p\left(\boldsymbol{x}'|\boldsymbol{\alpha}'\right) = \prod_{n'=1}^{3N} \mathcal{N}\left(x'\left[n'\right]|0, \left(\delta\left[n'\right]\right)^{-1}\right)$$

where the precision value of the n'th cell is defined as follows:

$$\delta'\left[n'\right] = \begin{cases} \alpha_{\mathrm{c}}\left[n'\right] + \beta \sum\limits_{m\in\mathbb{N}_{n'}} \alpha_{\mathrm{c}}\left[m\right], & \text{if } n' \le N \\ \alpha_{\Delta\mathrm{L}}\left[n'-N\right], & \text{if } N < n' \le 2N \\ \alpha_{\Delta\mathrm{R}}\left[n'-2N\right], & \text{if } 2N < n' \le 3N \end{cases} \tag{19}$$

for common part signal, and LiDAR and radar error collectors, respectively.

For the second layer, we use

$$p\left(\boldsymbol{\alpha}'\right) = \prod_{n'=1}^{3N} \mathrm{Gamma}\left(\alpha'\left[n'\right]|a\left[n'\right], b\left[n'\right]\right)$$

where the hyperparameters $a[n']$ and $b[n']$ are given by

$$\left(a\left[n'\right], b\left[n'\right]\right) = \begin{cases} \left(a_{\mathrm{c}}, b_{\mathrm{c}}\right), & \text{if } n' \le N \\ \left(a_{\mathrm{L}}, b_{\mathrm{L}}\right), & \text{if } N < n' \le 2N \\ \left(a_{\mathrm{R}}, b_{\mathrm{R}}\right), & \text{if } 2N < n' \le 3N \end{cases}$$

which correspond to the common part signal, and the LiDAR and radar error collectors, respectively.

The hyperparameters $a_{\mathrm{c}}, b_{\mathrm{c}}$ are set the same as a and b in the common sparse model. For each error collector, a larger a value makes the corresponding error collector more likely to be sparse, hence collecting fewer false positives, and vice versa. Therefore, the CIS fusion model provides a flexible and soft choice between the LiDAR and radar measurements by changing the prior model of the error collectors. Reliability information for sensors can be supplemented by other sensors, such as cameras or temperature sensors, which provide additional context, such as weather data. If both sensors are reliable, we choose high a_{L} and a_{R} values, enforcing $\boldsymbol{x}_{\Delta\mathrm{L}} = \boldsymbol{x}_{\Delta\mathrm{R}} = \mathbf{0}$. The resulting algorithm is similar to the common sparse fusion algorithm. If LiDAR (or radar) is known to be less reliable, the corresponding hyperparameter a_{L} (or a_{R}) is set to smaller than a_{R} (or a_{L}) to allow the error to be captured by $\boldsymbol{x}_{\Delta\mathrm{L}}$ (or $\boldsymbol{x}_{\Delta\mathrm{R}}$). Finally, similar to the common sparse model, the model discrepancies follow a Gaussian distribution with mean zero and unknown noise variances σ_{L}^2 and σ_{R}^2, and variances follow a Gamma prior distribution as given by (9) and (10).

To estimate the occupancy map, the posterior distribution of the extended map $\boldsymbol{x}'$ is inferred. With Gaussian measurement noises and first layer prior distributions, the posterior distribution of $\boldsymbol{x}'$ is also Gaussian with its posterior mean and variance given by

$$\begin{aligned} \boldsymbol{\mu}_{\boldsymbol{x}'} &= \boldsymbol{\Sigma}_{\boldsymbol{x}'} \boldsymbol{A}'^{\top} \boldsymbol{C}^{-1} \boldsymbol{y} \\ \boldsymbol{\Sigma}_{\boldsymbol{x}'} &= \left(\boldsymbol{A}'^{\top} \boldsymbol{C}^{-1} \boldsymbol{A}' + \mathrm{diag}\left(\boldsymbol{\delta}'\right)\right)^{-1}. \end{aligned} \tag{20}$$

To compute the sparse map $\boldsymbol{x}_{\mathrm{c}}$ from (5), we resort to the EM-based solver to estimate the unknown hyperparameters $\{\boldsymbol{\alpha}', \sigma_{\mathrm{L}}^2, \sigma_{\mathrm{R}}^2\}$. As shown in Algorithm 1, in the E-step of each iteration, the posterior mean and variance are calculated. The M-step update equation of $\boldsymbol{\alpha}'$ is given by

$$\alpha'^{(k+1)}\left[n'\right] = \frac{2a\left[n'\right]}{w'^{(k)}\left[n'\right] + \beta \sum_{m\in\mathbb{N}_{n'}} w'^{(k)}\left[m\right] + 2b\left[n'\right]} \tag{21}$$

where $w'^{(k)}[n'] = (\mu_{\boldsymbol{x}'}^{(k)}[n'])^2 + \Sigma_{\boldsymbol{x}'}^{(k)}[n', n']$ and $\mathbb{N}_{n'} = \emptyset$ for $N < n' \le 3N$.

The update equations for the noise variances are given by

$$\begin{aligned} \left(\sigma_{\mathrm{L}}^2\right)^{(k+1)} = \frac{1}{2c+2M_{\mathrm{L}}} \Big(&2d + \left\|\boldsymbol{y}_{\mathrm{L}} - \boldsymbol{A}_{\mathrm{L}}\left(\boldsymbol{x}_{\mathrm{c}} + \boldsymbol{x}_{\Delta\mathrm{L}}\right)\right\|_2^2 \\ &+ \mathrm{tr}\left(\boldsymbol{A}'^{\mathsf{T}} \boldsymbol{A}' \boldsymbol{\Sigma}_{\boldsymbol{x}'}^{(k)}\right)\Big) \end{aligned} \tag{22}$$

$$\begin{aligned} \left(\sigma_{\mathrm{R}}^2\right)^{(k+1)} = \frac{1}{2c+2M_{\mathrm{R}}} \Big(&2d + \left\|\boldsymbol{y}_{\mathrm{R}} - \boldsymbol{A}_{\mathrm{R}}\left(\boldsymbol{x}_{\mathrm{c}} + \boldsymbol{x}_{\Delta\mathrm{R}}\right)\right\|_2^2 \\ &+ \mathrm{tr}\left(\boldsymbol{A}'^{\mathsf{T}} \boldsymbol{A}' \boldsymbol{\Sigma}_{\boldsymbol{x}'}^{(k)}\right)\Big). \end{aligned} \tag{23}$$

The pseudocode for the CIS fusion model is summarized in Algorithm 2. This approach does not discard less reliable sensors but instead selectively incorporates their measurements by adjusting the sparsity of the error collectors, outperforming single-modality maps.

Complexity Analysis: Given M linear measurements and N cells in the occupancy map, the complexity of solving the PCSBL-based recovery algorithm in L iterations

Algorithm 2 Common Innovative Sparse LiDAR-Radar Fusion OGM

Input: M_{L} LiDAR points, M_{R} radar points, N grid cells, maximum number of iterations K, map threshold τ

Output: Occupancy map $\hat{\boldsymbol{x}}$

Parameters: $(a_{\mathrm{c}}, b_{\mathrm{c}})$, $(a_{\mathrm{L}}, b_{\mathrm{L}})$, $(a_{\mathrm{R}}, b_{\mathrm{R}})$
Generate $\boldsymbol{y}_{\mathrm{L}}$, $\boldsymbol{y}_{\mathrm{R}}$ and $\boldsymbol{A}_{\mathrm{L}}$, $\boldsymbol{A}_{\mathrm{R}}$ based on point cloud using (1) and (2)
Set noise hyperparameters $c, d = 1 \times 10^{-4}$
Initialize $k = 0$, $\boldsymbol{\alpha'}^{(0)} = \mathbf{0}$, $(\sigma_{\mathrm{L}}^2)^{(0)} = (\sigma_{\mathrm{R}}^2)^{(0)} = 0.5$
repeat
    Compute $\boldsymbol{\mu}_{\boldsymbol{x}'}^{(k)}$ and $\boldsymbol{\Sigma}_{\boldsymbol{x}'}^{(k)}$ according to (19) and (20)
    Update $\boldsymbol{\alpha'}^{(k+1)}$ according to (21)
    Update $(\sigma_{\mathrm{L}}^2)^{(k+1)}$ and $(\sigma_{\mathrm{R}}^2)^{(k+1)}$ as per (22) and (23)
    Set $k \leftarrow k + 1$
until convergence or $k > K$
Set $\boldsymbol{\mu}_{\boldsymbol{x}_{\mathrm{c}}}^{(k-1)}$ as the estimated probabilistic map $\hat{\boldsymbol{x}}$
Obtain the binary map $\hat{\boldsymbol{x}}_{\mathrm{b}}$ by thresholding $\hat{\boldsymbol{x}}_{\mathrm{b}} = \mathbf{1}_{\hat{\boldsymbol{x}} > \tau}$

TABLE I
ORDER OF COMPLEXITY COMPARISON

Algorithm	Computational complexity	Memory complexity
single-modality LiDAR	$\mathcal{O}(LN^3 + M_{\mathrm{L}}N^2)$	$\mathcal{O}(N^2)$
single-modality radar	$\mathcal{O}(LN^3 + M_{\mathrm{R}}N^2)$	
Fusion of single-modality maps	$\mathcal{O}(LN^3 + \max\{M_{\mathrm{L}}, M_{\mathrm{R}}\}N^2)$	
Common sparse and CIS fusion	$\mathcal{O}(LN^3 + (M_{\mathrm{L}} + M_{\mathrm{R}})N^2)$	

is $\mathcal{O}(LN^3 + MN^2)$, $N < M$, or $\mathcal{O}(LM^3 + NM^2)$, when $M < N$. In our setting, we find that there are generally more measurements than the number of cells, that is, $M_{\mathrm{L}}, M_{\mathrm{R}} > N$, and the complexity of the different OGM algorithms are summarized in Table I. Here, the fusion of single-modality maps refers to fusion approaches that generate the single-modality LiDAR and radar maps independently using the PCSBL algorithm in [8] and fuse them either before or after thresholding (for example, see baseline algorithms in Section V). The analysis shows that all the algorithms examined have a computational complexity of cubic order with respect to the map size N.

V. Performance Evaluation

We now present experimental results comparing our fusion algorithms with existing OGM algorithms.

A. Experimental Setting

This section outlines the experimental setup for our tests, including dataset specifications, evaluation metrics, benchmarking algorithms, and choices for algorithm hyperparameters.

The RADIATE dataset [34] is used for the experiments, which adapts a 32-channel LiDAR (Velodyne HDL-32E) covering a 40° vertical FOV and a 360° horizontal FOV, with a maximum range of 100 m, a range accuracy of 20 mm, and an angular resolution of 1.33°. The radar (Navtech CTS350-X) operates at 77–78 GHz with a bandwidth of 1 GHz, providing a range resolution of approximately 0.175 m and an angular resolution of approximately 1.8°.

1) Performance Metrics: To evaluate OGM algorithm performance, we generate ground truth data by extracting obstacle bounding box locations and dimensions from the dataset. Two error metrics, intersection over bounding box (IoBB), and angular scan normalized mean square error (AS-NMSE) [8], are used to quantify algorithm accuracy. IoBB measures the overlap between detected and true occupied cells for each obstacle, with values greater than zero indicating successful detection. While IoBB effectively quantifies detection accuracy, it does not account for false alarms. To address this, AS-NMSE is used to evaluate an algorithm's ability to accurately detect obstacle boundaries while minimizing false positives. AS-NMSE is computed by projecting lines from the ego vehicle at predetermined angles until they intersect obstacles or environment boundaries. The scan error for each angle is calculated as the distance difference between the first obstacle in the estimated and true maps. AS-NMSE is the normalized squared sum of these errors across all angles, indicating the drivable area detection.

2) Benchmarking Algorithms: The baseline algorithms are the single-modality LiDAR OGM [8], single-modality radar OGM [8], and two kinds of decision-level fusion algorithms of these single-modality maps, referred to as the logical OR operator-based fusion and Bayesian fusion.

The single-modality mapping algorithm from [8] provides an estimate of the occupancy probability maps and posterior variances of LiDAR and radar, denoted as $\{\hat{\boldsymbol{x}}_{\mathrm{L}}, \boldsymbol{\Sigma}_{\mathrm{L}}\}$ and $\{\hat{\boldsymbol{x}}_{\mathrm{R}}, \boldsymbol{\Sigma}_{\mathrm{R}}\}$. The logical OR fusion is a simple decision-level fusion strategy that tries to maximize detection, i.e., the fused probability map is obtained by

$$\hat{\boldsymbol{x}}_{\mathrm{OR}} = \max\left(\hat{\boldsymbol{x}}_{\mathrm{L}}, \hat{\boldsymbol{x}}_{\mathrm{R}}\right)$$

where max is the elementwise maximum operator, which is equivalent to logical OR operation on the single-modality binary maps. Another baseline algorithm, Bayesian fusion, is a more adaptive decision-level fusion model that fuses the single-modality maps by weighted summation such that the occupancy probability of the nth cell in the fused map is

$$\hat{x}_{\mathrm{B}}[n] = \frac{\Sigma_{\mathrm{R}}[n,n]\,\hat{x}_{\mathrm{L}}[n] + \Sigma_{\mathrm{L}}[n,n]\,\hat{x}_{\mathrm{R}}[n]}{\Sigma_{\mathrm{L}}[n,n] + \Sigma_{\mathrm{R}}[n,n]}.$$

The Bayesian fusion makes use of the posterior variances provided by the single-modality maps and possesses a better false alarm rate compared to the OR fusion.

For the single-modality and the common-sparse fusion, the hyperparameters are set to $a = 0.5$, $b = 10^{-4}$, and $\beta = 1$. For CIS fusion OGM, when both sensors are reliable, we choose $a_{\mathrm{c}} = 0.5$ and $a_{\mathrm{L}} = a_{\mathrm{R}} = 1.3$. In scenes where LiDAR (or radar) is considered less reliable, we change $a_{\mathrm{L}} = 0.54$ (or $a_{\mathrm{R}} = 0.54$). In both cases, β for the common part signal is set to $\beta = 1$. To generate binary maps from the probabilistic maps, the threshold is set as $\tau = 0.3$ for all algorithms.

Using the above experimental setup, we present two sets of results. The first set includes occupancy maps and error metrics

TABLE II
RESULTS OF SCENE CITY-3-0 FRAME 275

Metric	ID	Methods					
		LiDAR PCSBL	Radar PCSBL	OR	Bayesian	Common Sparse	CIS
IoBB	118	0.088	0.418	0.429	0.253	0.385	0.374
	159	0.233	0.209	0.395	0.326	0.349	0.372
	162	0.000	0.765	0.765	0.765	0.706	0.706
	163	0.241	0.185	0.426	0.204	0.204	0.204
	166	0.038	0.397	0.405	0.160	0.336	0.336
	167	0.185	0.148	0.296	0.204	0.222	0.222
	168	0.042	0.104	0.146	0.104	0.083	0.083
	169	0.000	0.204	0.204	0.082	0.102	0.102
	170	0.000	0.250	0.250	0.192	0.192	0.192
	172	0.020	0.040	0.060	0.060	0.020	0.020
	173	0.000	0.000	0.000	0.000	0.019	0.019
	175	0.000	0.000	0.000	0.000	0.000	0.000
	176	0.000	0.139	0.139	0.139	0.139	0.139
Detected Target		7 / 13	11 / 13	11 / 13	11 / 13	**12** / 13	**12** / 13
AS-NMSE		0.052	0.210	0.197	0.072	0.044	0.044

of all algorithms for two example scenes from the RADIATE dataset [34], City-3-0 frame-275, and City-3-1 frame-43 for visual comparison. The second set consists of statistical plots of error metrics averaged over multiple scenes, revealing the patterns and trends in the performance of different algorithms.

B. Comparison on Scene City-3-0 Frame-275

This scene represents a regular city road with 13 vehicles either parked (targets 162, 163, 167–170, 172, 173, and 175) or moving (targets 118, 159, 166, and 176), as shown in Fig. 2(a). The LiDAR and radar point cloud are shown in Fig. 2(b) and (c), respectively. The maps are shown in Fig. 2(d)–(i), and IoBB and AS-NMSE results are shown in Table II.

Our common sparse and CIS fusion OGM algorithms have the best AS-NMSE value compared to the baseline PCSBL-based algorithms. While our methods match the IoBB values of the OR fusion, they also eliminate false positives in front of the ego vehicle via the feature-level LiDAR-radar fusion. This leads to a lower AS-NMSE value, indicating superior drivable area detection performance. The low false alarm rate may have led to the detection of target 173 using its LiDAR point cloud measurements with a small IoBB value. In contrast, this target was missed by both the single-modality LiDAR PCSBL OGM and the OR fusion OGM. Consequently, our common sparse and CIS fusion approaches detect the most targets (12 out of 13). In addition, our methods show similar performances because the CIS fusion algorithm uses large hyperparameter values ($a_{\mathrm{L}} > 1$ and $a_{\mathrm{R}} > 1$) to make $\boldsymbol{x}_{\Delta\mathrm{L}}$ and $\boldsymbol{x}_{\Delta\mathrm{R}}$ close to zero, thereby making (3) and (5) equivalent.

C. Comparison on Scene City-3-1 Frame-43

This scene consists of four vehicles, two of which (targets 1 and 10) are parked at a crossroad, one (target 3) is moving along the ego vehicle, and the other (target 21) is moving on the opposite side of the road, as shown in Fig. 3(a). In this scene, the LiDAR is assumed to be unreliable compared to the radar, leading to an angular error. The error could arise from the displacement of the sensor mounting mechanism due

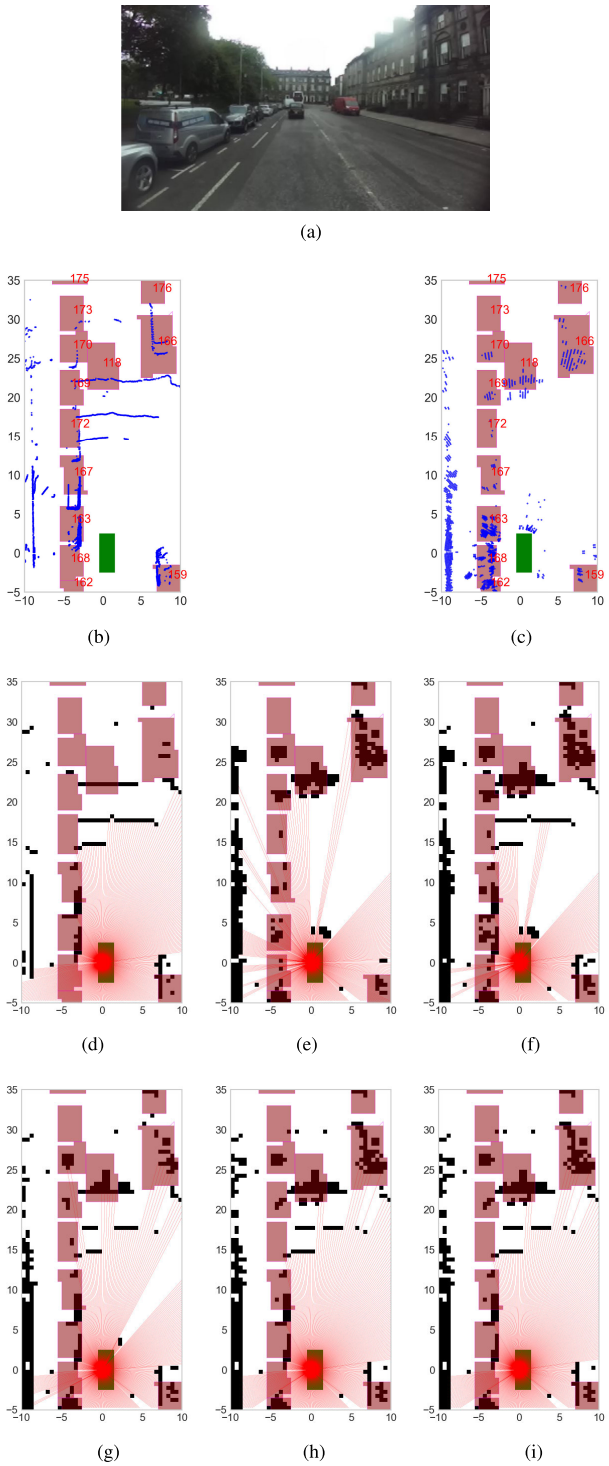


Fig. 2. Visualization of the scene City-3-0 frame 275 through (a) right camera view, (b) LiDAR point cloud measurements, and (c) radar point cloud measurements with points in blue and target indices labeled in red numbers. It also illustrates the reconstructed binary occupancy grid maps (in black) based on (d) single-modality LiDAR PCSBL, (e) single-modality radar PCSBL, (f) OR fusion, (g) Bayesian fusion, (h) common sparse fusion, and (i) CIS fusion algorithms, where the red boxes represent the ground truth and the red lines show AS-NMSE scan lines of the free space.

to a collision or maintenance misalignment. In the experiment, the error is introduced synthetically by rotating the LiDAR point cloud by 10° counterclockwise. The erroneous LiDAR point cloud is shown in Fig. 3(b), and the radar point cloud

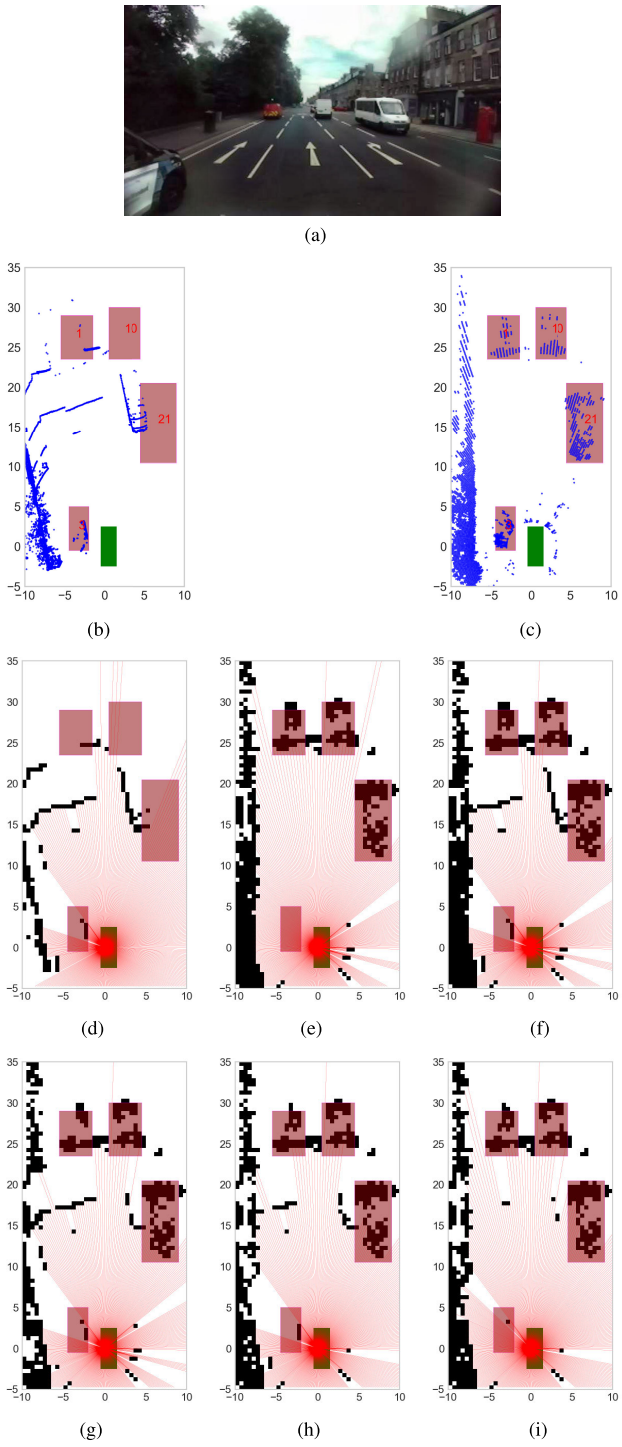


Fig. 3. Visualization of the scene City-3-1 frame 43 through (a) right camera view, (b) LiDAR point cloud measurements, and (c) radar point cloud measurements with points in blue and target indices labeled in red numbers. It also illustrates the reconstructed binary occupancy grid maps (in black) based on (d) single-modality LiDAR PCSBL, (e) single-modality radar PCSBL, (f) OR fusion, (g) Bayesian fusion, (h) common sparse fusion, and (i) CIS fusion algorithms, where the red boxes represent the ground truth, and the red lines show AS-NMSE scan lines of the free space.

is shown in Fig. 3(c). Since the LiDAR here is a less reliable sensor, we set $a_{\mathrm{L}} = 0.54$ and keep $a_{\mathrm{R}} = 1.3$ and $a_{\mathrm{c}} = 0.5$ for the CIS fusion model. The mapping results are given in Fig. 3(d)–(i), and IoBB and AS-NMSE are listed in Table III.

TABLE III
RESULTS OF SCENE CITY-3-1 FRAME 43

Metric	ID	Methods					
		LiDAR PCSBL	Radar PCSBL	OR	Bayesian	Common Sparse	CIS
IoBB	1	0.034	0.409	0.432	0.330	0.398	0.386
	3	0.091	0.000	0.091	0.073	0.073	0.018
	10	0.000	0.404	0.404	0.404	0.404	0.404
	21	0.017	0.428	0.444	0.428	0.417	0.417
Detected Target		3 / 4	3 / 4	**4** / 4	**4**/ 4	**4** / 4	**4** / 4
AS-NMSE		0.107	0.093	0.104	0.103	0.083	**0.079**

For the single-modality maps reconstructed using PCSBL, the LiDAR misses target 10 due to the angular error, and the radar misses target 3 due to background clutter.

The fusion results show that all the fusion algorithms perform better than the single-modality cases such that they detect all the targets. Compared to the OR fusion and Bayesian fusion, the common sparse fusion reduces the false positives, leading to a lower AS-NMSE value. However, obvious false positives from the LiDAR ground scatters and ill-registered points from target 21 are still present in the common sparse fusion map. With the knowledge that the LiDAR is less reliable, the CIS fusion model effectively captures the LiDAR measurement inconsistencies via its error collector, leading to a significant reduction in the LiDAR false positives and a better AS-NMSE value.

D. Statistical Results for Corrupted LiDAR Measurements

In this section, we analyze the statistical performance of algorithms when one sensor is unreliable, focusing on LiDAR measurements corrupted by angle errors. As outlined in Section V-C, these angle errors can arise from calibration issues leading to ill-registered LiDAR point cloud and large disparities between LiDAR and radar sensors. For each scene, we rotate the LiDAR point cloud across 11 angles, from $-10°$ to $10°$ in $2°$ increments. If the absolute angular error is less than $2°$, the point cloud is considered correctly registered, with the LiDAR error collector set to $a_{\mathrm{L}} = 1.3$. For errors exceeding $2°$, this collector is set to $a_{\mathrm{L}} = 0.54$ to accommodate decreased sensor reliability. We evaluate the performance of all the algorithms on 44 driving scenes from the RADIATE dataset. We record the average AS-NMSE and detection rates for each angular error, which are presented in Fig. 4(a) and (b), respectively. Here, the detection rate refers to the ratio of detected obstacles (i.e., IoBB > 0) to the total number of obstacles in the scene.

We first note that the AS-NMSE and detection values for the single-modality radar maps are unaffected by angular errors, as these errors only impact the LiDAR measurements. In contrast, for all other algorithms, the AS-NMSE reaches its minimum when the angular error is close to zero, while the detection rate is maximized under similar conditions, as expected.

When comparing single-modality LiDAR maps to our fusion models, we observe that LiDAR maps lead to a significantly high AS-NMSE value and a low detection

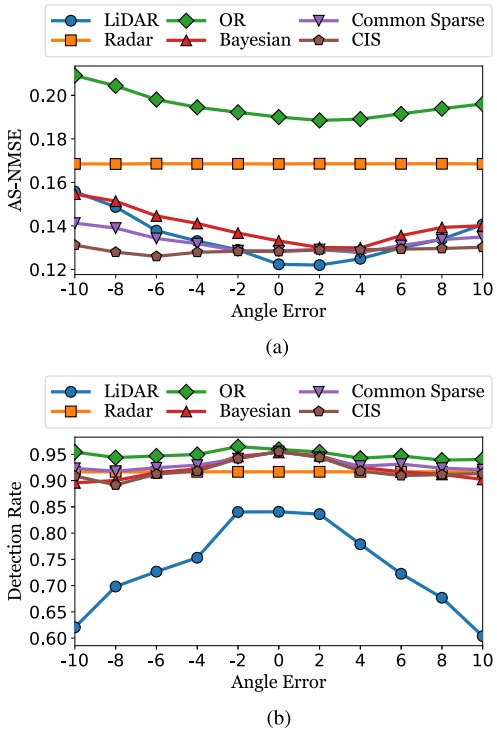


Fig. 4. Curves of (a) AS-NMSE and (b) detection rate of different algorithms as a function of LiDAR angular error, averaged over 44 scenes from RADIATE.

rate when the point cloud is corrupted by angular errors. This suggests that relying solely on one sensing modality can compromise both obstacle detection accuracy and the overall performance of the mapping algorithm when the sensor is unreliable.

The OR fusion method achieves the highest detection rate among all algorithms. However, it also suffers from a high AS-NMSE value, indicating an increased rate of false alarms. This tradeoff underscores the limitations of OR fusion, particularly in scenarios where sensor reliability is compromised. The Bayesian fusion model, another baseline, shows improved performance in terms of AS-NMSE while maintaining a detection rate comparable to that of the OR fusion method. This observation indicates that, compared to OR fusion, Bayesian fusion is more effective at reducing false positives while still reliably detecting obstacles.

Our common sparse method does not account for measurement errors, so it aligns with the CIS fusion model only when the angular error is small, and its performance deteriorates as the angular error increases. Nonetheless, the common sparse fusion model outperforms the OR fusion in AS-NMSE and achieves a detection rate on par with OR fusion. Furthermore, it excels over the Bayesian fusion model in both metrics, demonstrating its effectiveness in maintaining detection accuracy and minimizing false alarms.

Finally, in the presence of significant angular errors in LiDAR data, the CIS fusion model offers substantial improvements. It effectively reduces AS-NMSE to stable values around 0.13 compared to the common sparse fusion model while maintaining a comparable detection rate. Experiments with radar point cloud corrupted by the same setting of angular errors are omitted since statistical performance is similar. Thus, we infer that the CIS fusion model exhibits greater robustness to unreliable sensors, making it a more reliable choice in challenging conditions.

VI. Conclusion

In this work, we presented two feature-level sensor fusion models for sparse OGM: the common sparse and common innovative sparse fusion models. Experiments carried out on the real-world autonomous driving dataset RADIATE showed that when the sensors are reliable, both models enhance target detection compared to single-sensor approaches and improve drivable area detection relative to the baseline decision-level fusion models. By analyzing specific scenes and statistical results over multiple scenes with corrupted LiDAR data, we showed that when one sensor is less reliable, CIS surpasses other methods by providing better drivable area detection while keeping a comparable target detection rate. Despite the improved accuracy and robustness of the proposed fusion models compared to existing decision-level fusion models, they come at the cost of higher computational and memory complexity, which needs to be addressed in future research. This work also opens several future research directions, including adapting fusion models to handle sensor failures in adverse weather, joint camera-LiDAR-radar fusion models, and mapping algorithms that explore the sparse structure of the occupancy grid map.

References

[1] S. Steyer, C. Lenk, D. Kellner, G. Tanzmeister, and D. Wollherr, "Grid-based object tracking with nonlinear dynamic state and shape estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 7, pp. 2874–2893, Jul. 2020.

[2] A. Pandharipande et al., "Sensing and machine learning for automotive perception: A review," *IEEE Sensors J.*, vol. 23, no. 11, pp. 11097–11115, Jun. 2023.

[3] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, Jun. 1989.

[4] A. Elfes and L. Matthies, "Sensor integration for robot navigation: Combining sonar and stereo range data in a grid-based representataion," in *Proc. 26th IEEE Conf. Decis. Control*, Dec. 1987, pp. 1802–1807.

[5] J. Sock, J. Kim, J. Min, and K. Kwak, "Probabilistic traversability map generation using 3D-LiDAR and camera," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 5631–5637.

[6] S. T. O'Callaghan and F. T. Ramos, "Gaussian process occupancy maps," *Int. J. Robot. Res.*, vol. 31, no. 1, pp. 42–62, Jan. 2012.

[7] M. P. Gerardo-Castro, T. Peynot, and F. Ramos, *Laser-radar Data Fusion With Gaussian Process Implicit Surfaces*. Cham, Switzerland: Springer, 2015, pp. 289–302.

[8] Ç. Önen, A. Pandharipande, G. Joseph, and N. J. Myers, "Occupancy grid mapping for automotive driving exploiting clustered sparsity," *IEEE Sensors J.*, vol. 24, no. 7, pp. 9240–9250, Jul. 2024.

[9] J. Fang, Y. Shen, H. Li, and P. Wang, "Pattern-coupled sparse Bayesian learning for recovery of block-sparse signals," *IEEE Trans. Signal Process.*, vol. 63, no. 2, pp. 360–372, Jan. 2015.

[10] P. Zhai, G. Joseph, N. J. Myers, and A. Pandharipande, "Camera-aided binary prior support informed occupancy grid mapping," *IEEE Sensors J.*, early access, 2024.

[11] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[12] H. Caesar et al., "NuScenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11618–11628.

[13] D. Bauer, L. Kuhnert, and L. Eckstein, "Deep, spatially coherent inverse sensor models with uncertainty incorporation using the evidential framework," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 2490–2495.

[14] L. Sless, B. E. Shlomo, G. Cohen, and S. Oron, "Road scene understanding by occupancy grid learning from sparse radar clusters using semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 867–875.

[15] R. Weston, S. Cen, P. Newman, and I. Posner, "Probably unknown: Deep inverse sensor modelling radar," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 5446–5452.

[16] R. Cheng, C. Agia, Y. Ren, X. Li, and B. Liu, "S3CNet: A sparse semantic scene completion network for LiDAR point clouds," in *Proc. Conf. Robot Learn.*, Nov. 2021, pp. 2148–2161.

[17] L. Rold ao, R. de Charette, and A. Verroust-Blondet, "LMSCNet: Lightweight multiscale 3D semantic completion," in *Proc. Int. Conf. 3D Vis. (3DV)*, Nov. 2020, pp. 111–119.

[18] Y. Xu et al., "Sparse single sweep LiDAR point cloud segmentation via learning contextual shape priors from scene completion," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 4, pp. 3101–3109.

[19] C. B. Rist, D. Emmerichs, M. Enzweiler, and D. M. Gavrila, "Semantic scene completion using local deep implicit functions on LiDAR data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 7205–7218, Oct. 2022.

[20] M. Schreiber, V. Belagiannis, C. Gläser, and K. Dietmayer, "A multi-task recurrent neural network for end-to-end dynamic occupancy grid mapping," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2022, pp. 315–322.

[21] A. Popov, P. Gebhardt, K. Chen, and R. Oldja, "NVRadarNet: Real-time radar obstacle and free space detection for autonomous driving," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 6958–6964.

[22] A.-Q. Cao, A. Dai, and R. De Charette, "PaSCo: Urban 3D panoptic scene completion with uncertainty awareness," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 14554–14564.

[23] F. Ding, X. Wen, Y. Zhu, Y. Li, and C. X. Lu, "RadarOcc: Robust 3D occupancy prediction with 4D imaging radar," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, pp. 101589–101617.

[24] K. Qian, S. Zhu, X. Zhang, and L. E. Li, "Robust multimodal vehicle detection in foggy weather using complementary LiDAR and radar signals," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 444–453.

[25] B. Yang, R. Guo, M. Liang, S. Casas, and R. Urtasun, "RadarNet: Exploiting radar for robust perception of dynamic objects," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 496–512.

[26] Y.-J. Li, J. Park, M. O'Toole, and K. Kitani, "Modality-agnostic learning for radar-LiDAR fusion in vehicle detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 908–917.

[27] Y.-J. Li, M. O'Toole, and K. Kitani, "ST-MVDNet++: Improve vehicle detection with LiDAR-Radar geometrical augmentation via self-training," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2023, pp. 1–5.

[28] Y. Yang, J. Liu, T. Huang, Q.-L. Han, G. Ma, and B. Zhu, "RaLiBEV: Radar and LiDAR BEV fusion learning for anchor box free object detection systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 35, no. 5, pp. 4130–4143, May 2025.

[29] X. Wang et al., "OpenOccupancy: A large scale benchmark for surrounding semantic occupancy perception," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 17804–17813.

[30] J. Zhang, Y. Ding, and Z. L. Liu, "OccFusion: Depth estimation free multi-sensor fusion for 3D occupancy prediction," in *Proc. Asian Conf. Comput. Vis.*, 2024, pp. 232–249.

[31] X. Y. Tian, T. Jiang, L. Yun, Y. Wang, Y. Wang, and H. Zhao, "Occ3D: A large-scale 3D occupancy prediction benchmark for autonomous driving," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2023, pp. 64318–64330.

[32] R. Van Kempen, B. Lampe, T. Woopen, and L. Eckstein, "A simulation-based end-to-end learning framework for evidential occupancy grid mapping," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jul. 2021, pp. 934–939.

[33] D. L. Hall and J. Llinas, *Handbook of Multisensor Data Fusion* (The Electrical engineering and applied signal processing). Boca Raton, FL, USA: CRC Press, 2001.

[34] M. Sheeny, E. De Pellegrin, S. Mukherjee, A. Ahrabian, S. Wang, and A. Wallace, "RADIATE: A radar dataset for automotive perception in bad weather," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 1–7.

[35] P. Zhai, G. Joseph, N. J. Myers, Ç. Önen, and A. Pandharipande, "Sparsity-aware occupancy grid mapping for automotive driving using radar-LiDAR fusion," in *Proc. IEEE SENSORS*, Oct. 2024, pp. 1–4.

[36] T. Gomes, D. Matias, A. Campos, L. Cunha, and R. Roriz, "A survey on ground segmentation methods for automotive LiDAR sensors," *Sensors*, vol. 23, no. 2, p. 601, Jan. 2023.

[37] H. Rohling, "Radar CFAR thresholding in clutter and multiple target situations," *IEEE Trans. Aerosp. Electron. Syst.*, vols. AES–19, no. 4, pp. 608–621, Jul. 1983.

[38] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Syst. J.*, vol. 4, no. 1, pp. 25–30, Jul. 1965.

[39] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, no. 3, pp. 211–244, 2001.

[40] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Signal Process. Mag.*, vol. 25, no. 6, pp. 131–146, Nov. 2008.

Peiyuan Zhai (Graduate Student Member, IEEE) received the B.Eng. degree in information engineering from Xi'an Jiaotong University, Xi'an, China, in 2020, and the M.S. degree in electrical engineering from Delft University of Technology, Delft, The Netherlands, in 2022, where he is currently pursuing the Ph.D. degree with the Signal Processing Systems Group.

His research interests include automotive perception, multimodal sensing, and sparse Bayesian learning.

Geethu Joseph (Senior Member, IEEE) received the Ph.D. degree in electrical communication engineering from Indian Institute of Science, Bengaluru, India, in 2019.

She is currently a tenured Assistant Professor with the Signal Processing Systems Group, Delft University of Technology, Delft, The Netherlands. Her research interests include statistical signal processing, network control, and machine learning.

Nitin Jonathan Myers (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from The University of Texas at Austin, Austin, TX, USA, in 2020.

He is currently a tenured Assistant Professor with Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands. His research interests include optimization and multidimensional signal processing, with applications to communications and sensing.

Çağan Önen (Member, IEEE) received the B.Sc. degree in electrical and electronics engineering from Bilkent University, Ankara, Türkiye, in 2021, and the M.Sc. degree in electrical engineering from Delft University of Technology, Delft, The Netherlands, in 2023.

His research interests include wireless communications and signal processing.

Ashish Pandharipande (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Iowa, Iowa City, IA, USA, in 2002.

He is currently a Technical Director of NXP Semiconductors, Eindhoven, The Netherlands, and a part-time Full Professor with Eindhoven University of Technology, Eindhoven. His research interests include sensing, machine learning, data analytics, and their applications in domains such as autonomous mobility, smart lighting systems, and cognitive wireless systems.