LiDAR-Based Occupancy Grid Map Estimation Exploiting Spatial Sparsity

Önen, Çağan; Pandharipande, Ashish; Joseph, Geethu; Myers, Nitin Jonathan

DOI
10.1109/SENSORS56945.2023.10325050

Publication date
2023

Document Version
Final published version

Published in
Proceedings IEEE Sensors 2023

LiDAR-based Occupancy Grid Map Estimation Exploiting Spatial Sparsity

Çağan Önen, Ashish Pandharipande
NXP Semiconductors
High Tech Campus, Eindhoven, Netherlands
Email: {cagan.onen, ashish.pandharipande}@nxp.com

Geethu Joseph, Nitin Jonathan Myers
Delft University of Technology
Mekelweg 5, 2628 CD Delft, Netherlands
Email: {g.joseph, n.j.myers}@tudelft.nl

***Abstract*—The problem of estimating occupancy grids to support automotive driving applications using LiDAR sensor point clouds is considered. We formulate the problem as a sparse binary occupancy value reconstruction problem. Our proposed occupancy grid estimation method is based on pattern-coupled sparse Bayesian learning and exploits the inherent sparsity and spatial occupancy dependencies in LiDAR sensor measurements. The proposed method demonstrates enhanced detection capabilities compared to commonly used benchmark methods, as observed through testing on scenes from the nuScenes dataset.**

***Index Terms*—LiDAR point clouds, Occupancy grids, Automotive driving, Bayesian learning.**

I. Introduction

Occupancy grid mapping is the problem of constructing a grid-based estimate of the local environment using sensor data [1]. These maps form an essential building block of automotive perception [2]. Our work considers the occupancy map estimation problem using point cloud data derived from a LiDAR (Light Detection and Ranging) sensor to support assisted/autonomous driving applications.

Map building is a well-studied topic in robotics. The popular occupancy map estimation methods use the inverse-sensor model [3], [4] and kernel-based approaches [5], [6]. The inverse sensor model approach is computationally simple but often suffers from occupancy estimation conflicts when combining multiple measurements with partially overlapping fields of view. It also overlooks spatial correlation across the occupancies, leading to less accurate occupancy map estimates and sensitivity to sensor noise. This problem is handled by using a Gaussian Process to model the spatial dependencies at the expense of increased computational complexity [5]. Recently, an improved version of this approach, Bayesian Generalized Kernel-based mapping (BGK) in [6], addressed the complexity issue. However, the above mapping algorithms [3]–[6] developed for robotic applications do not exploit contextual information and the underlying sparse structure in LiDAR sensor measurements when estimating occupancy maps for automotive perception. The sparse structure arises because, even in the presence of obstacles, only the borders or a portion of objects are captured by sensor measurements. To this end, we present an occupancy mapping method that exploits the sparsity and spatial correlation in the occupancy map.

Our approach relies on the pattern-coupled sparse Bayesian learning (PC-SBL) framework [7]. In the PC-SBL framework, the sparse structure is captured via a sparsity-promoting prior distribution on the map, and the spatial dependencies are exploited by assuming a block-sparse structure for the map. This assumption is valid since real-world obstacles span multiple occupancy grid cells for typical LiDAR resolutions. PC-SBL estimates block-sparse vectors without assuming the block size, making it promising for automotive driving applications where prior knowledge of the object sizes is unavailable.

We evaluate our method using LiDAR point cloud data from the nuScenes dataset [8] and compare it with the occupancy map estimation methods in [3], [6]. Our approach outperforms these methods by accurately resolving obstacles and effectively eliminating road reflections.

II. LiDAR sensor signal model

We consider a LiDAR sensor mounted on an ego vehicle, making range measurements by a 360° scan of the environment. We exclude points that fall outside the interest range and those below and above the desired height. Let M be the number of remaining LiDAR points, referred to as reflection points, representing potential occupancy locations inside the region of interest. Each measurement consists of the 3D coordinates of a reflection point, which is represented in the global coordinate system with the ego vehicle's position as the origin. We aim to build a binary occupancy grid map using the point cloud, indicating the grid cell occupancies.

We next develop a new signal model using the point cloud measurements to derive our mapping algorithm. The first step is to include the contextual information from the digital maps [9] in the model by defining a *focus area*. It comprises the cells of interest, i.e., the cells on the ego vehicle's road and the walkways around [10]. The area beneath the walkways is not crucial for driving scenarios. This approach reduces the dimensionality of the occupancy estimation problem, and now our goal is to estimate the occupancy map over the focus area.

Once the focus area is defined, it is partitioned into N two-dimensional cells of equal size. Let $\mathbf{f} \in \mathbb{R}^N$ denote the unknown map vector comprising the occupancy probabilities of the N cells, representing the occupancy grid map. To estimate the map vector $\mathbf{f}$, we construct a new measurement model using the LiDAR point cloud. Each LiDAR measurement implies that the corresponding reflection point (discretized to the nearest grid cell) is occupied, and the cells along the line

2023 IEEE SENSORS | 979-8-3503-0387-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/SENSORS56945.2023.10325050

connecting the ego vehicle and reflection point are unoccupied. Therefore, we obtain two linear map measurements from every LiDAR measurement. If the cell corresponding to the nth entry of $\mathbf{f}$ is a reflection point, we have $\mathbf{e}_n^\mathsf{T}\mathbf{f} = 1$ and $\sum_{k\in\mathcal{F}_n} \mathbf{e}_k^\mathsf{T}\mathbf{f} = 0$. Here, $\mathbf{e}_k \in \{0,1\}^N$ is the kth standard basis vector, and $\mathcal{F}_n$ denote the cells sampled along the line from the ego vehicle to the nth cell. The measurement vector $\mathbf{y} \in \{0,1\}^{2M}$ resulting from M reflection points is

$$\mathbf{y} = \mathbf{C}\mathbf{f} + \mathbf{w}, \tag{1}$$

where $\mathbf{w} \in \mathbb{R}^{2M}$ is the additive Gaussian measurement noise with zero mean and unknown variance σ^2 and $\mathbf{C} \in \{0,1\}^{2M\times N}$ is the selection matrix. Specifically, if the mth reflection point corresponds to the nth entry of $\mathbf{f}$, we get $y[2m-1] = 1$ and $y[2m] = 0$. Also, the $(2m-1)$th and $2m$th rows of $\mathbf{C}$ are $\mathbf{e}_n^\mathsf{T}$ and $\sum_{k\in\mathcal{F}_n} \mathbf{e}_k^\mathsf{T}$, respectively. Hence, our occupancy mapping problem reduces to estimating the map vector $\mathbf{f}$ from the linear measurement vector $\mathbf{y}$ by exploiting sparsity and spatial correlation in the map.

III. Occupancy grid estimation

Our estimation algorithm reconstructs the unknown map vector $\mathbf{f}$ from measurement vector $\mathbf{y}$ in (1) by accounting for its block sparse structure via PC-SBL. Motivated by the PC-SBL framework, we impose a fictitious Gaussian hierarchical prior on sparse vector $\mathbf{f}$,

$$p(\mathbf{f}|\boldsymbol{\alpha}) = \prod_{n=1}^{N} p(f[n]|\boldsymbol{\alpha}) = \prod_{n=1}^{N} \mathcal{N}(0, (\alpha[n] + \beta \sum_{j\in\mathcal{L}_n} \alpha[j])^{-1}). \tag{2}$$

Here, $\boldsymbol{\alpha} \in \mathbb{R}^N$ denotes the unknown hyperparameters, $\beta \in [0,1]$ is the prefixed coupling parameter, and $\mathcal{L}_n$ denotes the set of immediate neighbors (the adjacent left, right, above, and below cells) of the nth cell. Due to the shared hyperparameters of the prior distributions, the entry $\mathbf{f}[n]$ goes to zero if its hyperparameter $\boldsymbol{\alpha}[n]$ or any of its neighboring hyperparameters in $\boldsymbol{\alpha}_{\mathcal{L}_n}$ go to infinity. So, this structure encourages block sparsity. The SBL hierarchical model [11] also uses Gamma prior distribution on the hyperparameters $\boldsymbol{\alpha}$ and the inverse of the noise variance $\gamma = \sigma^{-2}$ with prior parameters a, b, c, $d > 0$ as follows,

$$p(\boldsymbol{\alpha}) = \prod_{n=1}^{N} \Gamma\left(\alpha[n] \mid a, b\right) = \prod_{n=1}^{N} \Gamma(a)^{-1} b^a \alpha[n]^a e^{-b\alpha[n]} \tag{3}$$

$$p(\gamma) = \Gamma(\gamma \mid c, d) = \Gamma(c)^{-1} d^c \gamma^c e^{-d\gamma}. \tag{4}$$

Using the above hierarchical prior model, we use type II maximum likelihood estimation of $\mathbf{f}$ where we first estimate the parameters $\boldsymbol{\alpha}$ and γ from the measurement $\mathbf{y}$. This estimation relies on the iterative expectation-maximization method with $\mathbf{f}$ being the unobserved latent variable. The tth iteration of the EM algorithm is given by

$$\alpha_t[n] = \frac{a}{0.5\hat{\nu}_t[n] + \beta \sum_{j\in\mathcal{L}_n} \hat{\nu}_t[j] + b} \tag{5}$$

$$\gamma_t = \frac{\|\boldsymbol{y} - \boldsymbol{C}\hat{\boldsymbol{\mu}}_t\|_2^2 + \gamma_{t-1}^{-1} \sum_n \hat{\Phi}_t[n,n] D_t[n,n] + 2d}{M + 2c}, \tag{6}$$

where $\hat{\boldsymbol{\nu}}_t \in \mathbb{R}^N$ with the nth entry as $\hat{\nu}_t[n] = \hat{\mu}_t[n]^2 + \hat{\Phi}_t[n,n]$. Also, we define

$$\hat{\boldsymbol{\mu}}_t = \gamma_{t-1} \hat{\boldsymbol{\Phi}}_t \mathbf{C}^T \mathbf{y},\ \hat{\boldsymbol{\Phi}}_t = (\gamma_{t-1} \mathbf{C}^T \mathbf{C} + \mathbf{D}_t)^{-1}, \tag{7}$$

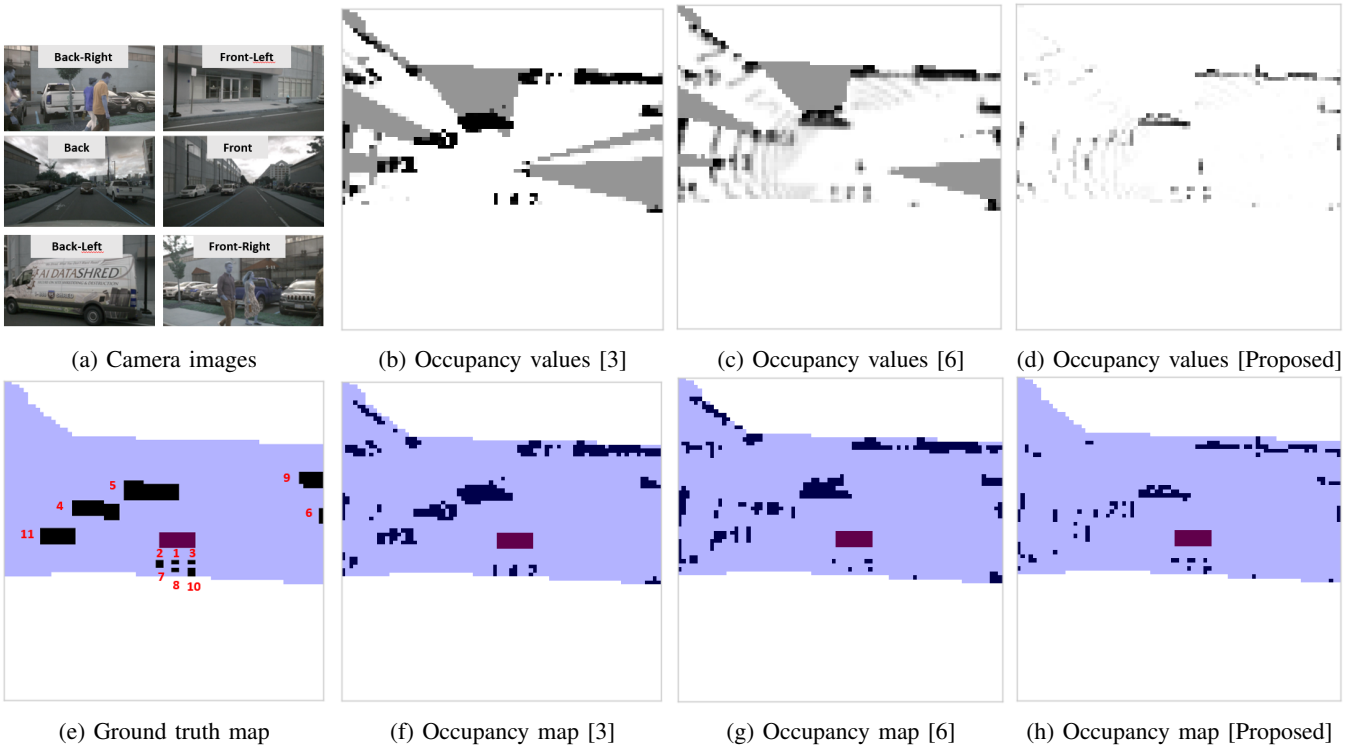
and $\mathbf{D}_t$ is a diagonal matrix with $\mathbf{D}_t[n,n] = \boldsymbol{\alpha}_{t-1}[n] + \beta \sum_{j\in\mathcal{L}_n} \boldsymbol{\alpha}_{t-1}[j]$. For a detailed derivation of the above steps, please refer to [7]. The EM updates in (5) and (6) are repeated until convergence, and the final $\hat{\boldsymbol{\mu}}_t$ is the estimate of $\mathbf{f}$. Lastly, we apply thresholding to this MAP estimate to obtain a binary occupancy grid map.

IV. Performance evaluation

We use LiDAR sensor data and camera data obtained from the nuScenes dataset [8] in our performance evaluation. For each scene, we use 6 camera images and the corresponding ground truth bounding box image to assess the accuracy of the estimated occupancy map. We compare the occupancy grid maps using our approach against two other algorithms [3], [6]. The assessment is done both qualitatively and using two quantitative metrics that measure the difference from the ground truth. The ground truth map is created from nuScenes using the locations and sizes of the boundary boxes within the focus driveable region.

To evaluate the accuracy of the map, we employ two metrics: Angular Scan Normalized Mean-Squared Error (NMSE), where the scan is done at the ego vehicle, and Intersection over Bounding Box (IoBB). To define NMSE, we denote $\mathbf{x}$ as a vector constructed from the ground truth, which comprises the distances from the ego vehicle to the first occupied cell along each direction of the angular scan. In particular, its ith element $x[i]$ represents the maximum distance the ego vehicle can travel in direction i before encountering an obstacle. A similar vector $\hat{\mathbf{x}}$ is defined for the estimated occupancy map. The angular scan NMSE is defined as $\text{NMSE} = \frac{\|\mathbf{x}-\hat{\mathbf{x}}\|^2}{\|\mathbf{x}\|^2}$. To define IoBB, we extend the metric in [12] as the ratio of the overlapping area between the occupied cells and the ground truth boundary boxes to the area of the ground truth boundary boxes. NMSE indicates the quality of identifying the drivable area, while IoBB evaluates the performance of obstacle detection.

The ground truth and estimated occupancy maps for scene 204 are shown in Fig. 1. We use $M = 10178$ reflection points to construct the occupancy map after removing the points below 0.2 m or above 2.5 m. The occupancy grid maps were generated with the grid size set at 0.5 m. The map spans 40 m by 40 m, leading to $N = 6400$. The ego vehicle is positioned at the center of the map. For scene 204, the computed metrics NMSE and IoBB are shown in Tables II and I, respectively. For NMSE calculation, the angular resolution is chosen as 4°. These results were obtained by optimizing the hyperparameters in the corresponding algorithms. Specifically, the free space line segments are sampled with a resolution of $1\,\text{m}$ for [6] and $0.5\,\text{m}$ for the proposed approach. We set the kernel size as $1\,\text{m}$ for [6] to exploit the spatial correlation between neighboring cells. Finally, β is set to 1 as suggested in [7].



(a) Camera images (b) Occupancy values [3] (c) Occupancy values [6] (d) Occupancy values [Proposed]

(e) Ground truth map (f) Occupancy map [3] (g) Occupancy map [6] (h) Occupancy map [Proposed]

Fig. 1: Occupancy grid mapping results for scene-204. Figures (b),(c), and (d) show the occupancy values of the grid cells for methods [3], [6], and ours, respectively. White stands for 0 and black for 1. Thresholding is applied to extract the occupied cells that are shown in subfigures (f), (g), and (h). The threshold value is 0.5 for [3], [6], and 0.25 for the proposed approach. The focus driveable area is labeled with blue and the location of the ego vehicle (headed right) is highlighted with dark red.

Object ID	Object Type	IoBB values		
		[3]	[6]	Proposed
1	Pedestrian	0	0	0
2	Pedestrian	0	0.5	1
3	Pedestrian	0	1	0.5
4	Car	0.60	0.40	0.15
5	Van	0.74	0.74	0.39
6	Car	0	0.75	1
7	Pedestrian	0.5	0	0.5
8	Pedestrian	1	0.50	0.5
9	Car	0.48	0.52	0.35
10	Pedestrian	0.25	0.25	0.75
11	Car	0.69	0.47	0.31

TABLE I: IoBB for scene-204

Methods	NMSE
[3]	0.57
[6]	0.79
Proposed	0.39

TABLE II: Angular scan NMSE for scene-204

We now consider scene-204, in which the focus area contains 11 labeled objects, including 4 cars and 6 pedestrians to the right of the ego vehicle. As two pedestrians are close to each other, they are represented with a single boundary box leading to five boundary boxes for pedestrians in Fig. 1(e). From Fig. 1, we notice that [6] and our approach successfully detect all the cars, whereas [3] does not detect the car at the right end of the road. Next, we observe that the three methods demonstrate varying performances in detecting pedestrians with IDs 1, 2, 3, 7, and 8, positioned to the right of the car. The IoBB values in Table I indicate that [3] detects two of the pedestrians, [6] detects 4 of them, and our approach detects 5 of them. The pedestrian with ID 1 remains undetected by all algorithms since the visibility of this pedestrian is obstructed by a neighboring pedestrian. The better performance of our algorithm in detecting the pedestrians and other objects in the scene is also observed in the lower NMSE value in Table II.

V. Conclusion

Our results show that the proposed approach performs better than the benchmarks in two different ways. First, the occupancy values due to road reflections are suppressed due to using a sparse prior on the occupancy map and exploiting spatial correlation. Second, the proposed method can detect smaller obstacles like pedestrians better. Finally, our method results in lower IoBB values for the detected objects than the other methods. This is because our method promotes a sparsity in the occupancy map.

Due to space limitations, we showed performance results for one specific scene. In an extended version [13], performance analysis with more scenes will be shown along with a statistical performance characterization of the algorithms on the nuScenes dataset.

REFERENCES

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

[2] A. Pandharipande, C.-H. Cheng, J. Dauwels, S. Z. Gurbuz, J. Ibanex-Guzman, G. Li, A. Piazzoni, P. Wang, and A. Santra, "Sensing and machine learning for automotive perception: A review," *IEEE Sens. J.*, vol. 23, no. 11, pp. 11 097–11 115, 2023.

[3] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents series)*. The MIT Press, 2005.

[4] E. Kaufman, T. Lee, Z. Ai, and I. S. Moskowitz, "Bayesian occupancy grid mapping via an exact inverse sensor model," in *Proc. ACC*, Jul. 2016, pp. 5709–5715.

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

[6] K. Doherty, J. Wang, and B. Englot, "Bayesian generalized kernel inference for occupancy map prediction," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2017, pp. 3118–3124.

[7] 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, Nov. 2014.

[8] 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 *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11 621–11 631.

[9] K. Irie and M. Tomono, "Localization and road boundary recognition in urban environments using digital street maps," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 4493–4499.

[10] M. Kurdej, J. Moras, V. Cherfaoui, and P. Bonnifait, "Map-aided evidential grids for driving scene understanding," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 30–41, 2015.

[11] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, Jun. 2001.

[12] V. Jiménez, J. Godoy, A. Artuñedo, and J. Villagra, "Object-wise comparison of lidar occupancy grid scan rendering methods," *Robotics and Autonomous Systems*, vol. 161, p. 104363, 2023.

[13] C. Önen, A. Pandharipande, G. Joseph, and N. J. Myers, "Occupancy grid mapping for automotive driving exploiting clustered sparsity," *IEEE Sens. J.*, Submitted.