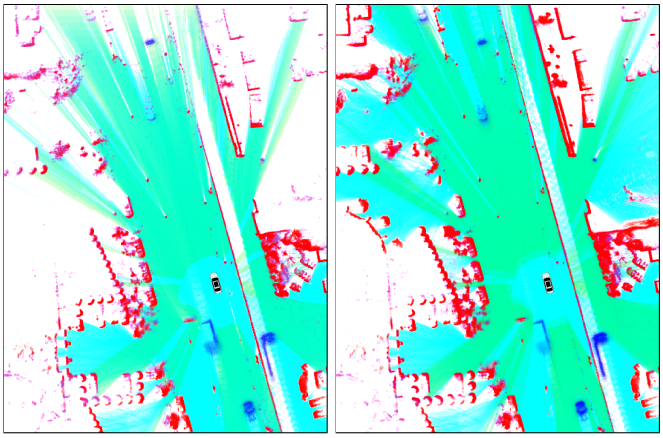
Offline Dynamic Grid Generation for Automotive Environment Perception Using Temporal Inference Methods

Egon Ye*, Gerald Würsching*, Sascha Steyer, and Matthias Althoff

***Abstract*—The validation of online perception algorithms in automotive systems requires a large amount of ground-truth data. Since manual labeling is inefficient and error-prone, an automatic generation of accurate and reliable reference data is desirable. We present a post-processing approach based on a particle-based dynamic occupancy grid representation of the environment. In contrast to existing online dynamic grid algorithms, our estimation additionally utilizes future measurements by applying offline smoothing algorithms. Our proposed concept uses a two-filter procedure for smoothing the occupancy states of the grid cells. We further introduce two methods based on particle reweighting and two-filter smoothing to improve the velocity estimates. We show that our approach enhances the situational awareness and thus provides a more precise environment model. We demonstrate these benefits using lidar data from real-world experiments.**

***Index Terms*—Mapping, intelligent transportation systems, sensor fusion**



(a) Filtered map. (b) Smoothed map.

Fig. 1. Our smoothing approach results in more confident estimates indicated by a higher color intensity, which leads to better situational awareness of the road ahead. This is particularly visible in the upper left corner of both maps, which represents an area that has not yet been fully observed during the online filtering algorithm.

I. Introduction

GROUND-TRUTH data is required for evaluating online perception such as object tracking. One of the methods to generate ground-truth data is to post-process sensor data [1], which gained attention in the last few years due to its scalability in comparison to approaches including recordings from differential global positioning systems (DGPS) [2] or manual labeling [3]. However, previous research mainly focuses on smoothing of state estimates at the object level, which requires object assumptions and data association methods. Instead, our paper proposes an offline smoothing algorithm based on a low-level dynamic grid representation, which already improves the accuracy of occupancies and velocities before extracting objects (see Fig. 1).

A. Related Work

Offline Post-processing: In contrast to online methods, offline post-processing of sensor data makes it possible to use precise methods to generate ground-truth data since no real-time capability is required and future measurements can be exploited for information fusion. A popular post-processing approach on the object level is to initialize objects individually at a well observable time step and to track them for both past and future times. This is shown for classical fitting of L-, I-, and U-shapes in [4] and for computer-aided design (CAD) models in [5].

Alternative offline methods apply the same procedure as online, but add subsequent improvements, e.g., by applying updated shape information to past time steps to refine object poses and by detecting objects in time steps before their online initialization. Such an approach is pursued in [6] with classical shape-fitting on lidar data and in [7] based on filtered dynamic occupancy grid sequences. Likewise, the work in [8] presents a concept using the random finite set framework with extended object shape estimation.

A further promising post-processing approach is proposed in [9], which fuses raw sensor data from multiple vehicles to reduce the drawbacks of occlusions and decreasing sensor resolution with distance when considering sensors of a single vehicle. However, this approach mainly benefits from offline computing and does not consider future measurements.

Occupancy Grids: Static occupancy grids [10], [11] are widely used in robotics for mapping the environment by

Manuscript received: October 15, 2020; Revised January 10, 2021; Accepted February 5, 2021. This paper was recommended for publication by Editor Sven Behnke upon evaluation of the Associate Editor and Reviewers' comments.

*Egon Ye and Gerald Würsching have contributed equally to this work.

Egon Ye and Sascha Steyer are with BMW Group, D-85716 Unterschleissheim, Germany. {egon.ye, sascha.steyer}@bmw.de

Gerald Würsching and Matthias Althoff are with the Department of Informatics, Technical University of Munich, D-85748 Garching, Germany. {gerald.wuersching, althoff}@tum.de

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the author.

Digital Object Identifier (DOI): see top of this page.

estimating occupancy probabilities of grid cells. Dynamic occupancy grids also estimate the velocity of the cells, thus enabling the distinction between static and dynamic occupancy. Initial dynamic grid approaches [12], [13] use a four-dimensional state (i.e., two dimensions for each, position and velocity) per grid cell and estimate the cell velocities using discrete velocity histograms. More efficient and robust methods apply particle filtering to estimate the velocity distribution and occupancy probability of each cell [14].

Grid-Based Tracking and Mapping: In recent works [15]–[17], the particle-based approach was extended towards an evidential grid representation using the Dempster-Shafer theory (DST). In grid-based tracking and mapping (GTAM), an evidential mapping problem and a probabilistic tracking problem are solved simultaneously. Therein, the evidences for cell occupancy are estimated by a temporal filtering in the Dempster–Shafer framework, whereas the cell velocities are estimated by the filtering procedure of the particles. Overall, GTAM provides a robust environment model without object assumptions by only combining a uniform grid representation with a low-level velocity estimation using particles and an explicit static/dynamic occupancy classification using DST.

Based on the dynamic grid, an object-level representation can be obtained by clustering dynamic cells for object extraction as presented in [18], [19]. Hence, given more accurate cell velocity estimates and static/dynamic classifications, a higher robustness of the object extraction is achieved.

B. Contributions

This paper presents a post-processing approach to increase the accuracy and reliability of dynamic occupancy grids for generating ground-truth data. In contrast to the online dynamic grid estimation in [17], we do not only exploit past but also future measurements using offline smoothing algorithms [20] and thus obtain a more precise environment model. Our concept uses a two-filter recursion for smoothing the occupancy states of the grid cells, which benefits from an easy integration of existing online occupancy grid algorithms. Furthermore, we propose two methods to enhance the velocity estimation: One uses a particle reweighting scheme for smoothing the cell velocity estimates, the other applies a two-filter approach to the velocity grid.

As a result, enhanced input data is provided for subsequent perception tasks, such as the extraction of objects from dynamic grids. We evaluate our approach using real-world lidar data and demonstrate the improved performance of our concept by comparing the offline smoothing results with those of online filtering.

This paper is structured as follows: Section II introduces the necessary preliminaries of dynamic grid mapping and defines the smoothing problem of this work. In section III, we describe our two-filter recursion for smoothing the occupancy states. Section IV explains the two variants for smoothing the velocity estimates. We present our experimental results in section V and give concluding remarks in section VI.

II. Preliminaries & Problem Formulation

This section briefly reviews the basic concept of the GTAM approach proposed by [16], [17], which serves as the foundation for this work. Additionally, we define the offline post-processing problem for dynamic grids.

Grid-Based Tracking and Mapping (GTAM): In the GTAM concept proposed in [16], [17], the environment is represented as an evidential occupancy grid estimated using the Dempster–Shafer Theory (DST) [21], [22]. Therein, the so-called frame of discernment Θ is defined by the singleton hypotheses for free space (F), static (S), and dynamic (D) occupancy. Thus, Θ and its powerset 2^Θ, which contains all considered hypotheses, are

$$\begin{aligned} \Theta &= \{F, S, D\}, \\ 2^\Theta &= \{\emptyset, \{F\}, \{S\}, \{D\}, \{F,S\}, \{F,D\}, \{S,D\}, \Theta\}, \end{aligned} \tag{1}$$

for which unclassified occupancy $\{S, D\}$, passable area $\{F, D\}$ and the unknown state Θ are explicitly modeled. The set $\{F, S\}$ is neglected as it is always conflicting.

In DST, hypotheses are estimated using a mass function $m : 2^\Theta \to [0, 1]$, which assigns a basic belief mass $m(A)$ to every set $A \in 2^\Theta$, such that [23]

$$\sum_{A \subseteq \Theta} m(A) = 1, \qquad m(\emptyset) = 0. \tag{2}$$

To indicate cell-related variables, we use the index $c \in \mathcal{C}$ with $\mathcal{C}$ denoting the set of all grid cells.

Additionally, dynamic occupancy is estimated using particle filtering. Each particle $i \in \mathcal{I}_t$ in the set $\mathcal{I}_t$ of N_t particles at time t represents a hypothesis of dynamic occupancy at position $[s_{x,t}^{(i)},\ s_{y,t}^{(i)}]^T$ in a local world-fixed 2D coordinate system with velocity components $[v_{x,t}^{(i)},\ v_{y,t}^{(i)}]^T$. This results in the particle state vector

$$\boldsymbol{x}_t^{(i)} = [s_{x,t}^{(i)}, s_{y,t}^{(i)}, v_{x,t}^{(i)}, v_{y,t}^{(i)}]^T. \tag{3}$$

Furthermore, a filtered occupancy value $o_t^{(i)} \in [0, 1)$ is assigned to every particle, which can be interpreted as the particle weight. For each cell $c \in \mathcal{C}$, the particle weights are normalized to the filtered dynamic mass $m(D_t^c)$ of the cell and an estimate of the cell velocity is calculated as the weighted mean of the particle velocities in the cell [17].

Temporal filtering consists of predicting the prior map, for which the dynamic occupancy is predicted by the particles using a dynamic motion model, and updating the predicted map with an evidential measurement grid. For fusing independent evidences, the DST framework provides several combination rules [23], from which we apply the conjunctive rule $\oplus_\mathrm{c}$ defined as [23]

$$m_1(A) \oplus_\mathrm{c} m_2(A) = \sum_{B \cap C = A} m_1(B) m_2(C), \quad \forall A, B, C \subseteq \Theta, \tag{4}$$

for two individual mass functions m_1 and m_2. Therein, the conflicting combinations (i.e., $B \cap C = \emptyset$) can be distributed individually as presented in [17].

Problem Formulation: The approaches for the temporal inference problem in this work are twofold. Let Θ_t^c be the frame of discernment for cell c at time step t. In the DST framework, the estimation problem for occupancy smoothing is to find the smoothed basic belief mass of the individual hypotheses $X_t^c \subseteq \Theta_t^c$ conditioned on all measurements $z_{0:T} = \{z_0, ..., z_T\}$ over a fixed time interval:

$$m(X_t^c|z_{0:T}), \quad \forall c \in \mathcal{C}, \quad t = 0, ..., T. \tag{5}$$

Particle smoothing is defined by Bayesian inference methods, which estimates the particle states by maximizing over $p(\boldsymbol{x}_t^{(i)}|z_{0:T})$ as the conditional probability distribution of $\boldsymbol{x}_t^{(i)}$ given the measurements $z_{0:T}$. Thus, the smoothed states are

$$\boldsymbol{x}_t^{(i)*} = \arg\max_{\boldsymbol{x}_t^{(i)}} p(\boldsymbol{x}_t^{(i)}|z_{0:T}), \quad \forall i \in \mathcal{I}_t, \quad t = 0, ..., T. \tag{6}$$

Our solution to this problem is presented in the next two sections.

III. Two-Filter Occupancy Smoothing

To estimate the smoothed grid occupancy, we use the two-filter approach based on [24], which fuses two independent paths, i.e., forward and backward recursions, using DST combination rules. Fig. 2 visualizes our two-filter occupancy smoothing concept. Therein, $\mathcal{M}_t^f = \{m(X_t^c|z_{0:t})\}_{c\in\mathcal{C}}$ is the forward filtered map, $\mathcal{M}_t^b = \{m(X_t^c|z_{t+1:T})\}_{c\in\mathcal{C}}$ is the backward predicted map, and $\mathcal{M}_t^s = \{m(X_t^c|z_{0:T})\}_{c\in\mathcal{C}}$ is the smoothed map with conditioned mass functions for forward filtering $m(X_t^c|z_{0:t})$, backward prediction $m(X_t^c|z_{t+1:T})$, and smoothing $m(X_t^c|z_{0:T})$, respectively. In the following considerations, the equations are defined for all $X_t^c \subseteq \Theta_t$ and we drop the cell index c for simplicity.

For the application to mass functions, we first reformulate the recursive equations from [24] using the conjunctive rules for conditional masses [25, p. 7]. For the forward filtering pass, this results in

$$\begin{aligned} m(X_t|z_{0:t}) \\ &= m(X_t|z_{0:t-1}) \oplus_{\mathrm{c}} m(X_t|z_t) \\ &= \Big[\sum_{X_{t-1}\subseteq\Theta_{t-1}} m(X_{t-1}|z_{0:t-1})\, m(X_t|X_{t-1})\Big] \\ &\quad \oplus_{\mathrm{c}} m(X_t|z_t), \end{aligned} \tag{7}$$

where $m(X_t|X_{t-1})$ and $m(X_t|z_t)$ are the beliefs conditioned on transitions and observations, respectively. Since forward filtering only uses past and current measurements, the implementation for (7) is equivalent to online methods (e.g., [17]). The reformulated backward prediction recursion [24] is

$$\begin{aligned} m(X_t|z_{t+1:T}) \\ &= \sum_{X_{t+1}\subseteq\Theta_{t+1}} \big[m(X_{t+1}|z_{t+1:T})\, m(X_t|X_{t+1})\big] \\ &= \sum_{X_{t+1}\subseteq\Theta_{t+1}} \big[(m(X_{t+1}|z_{t+2:T}) \\ &\qquad \oplus_{\mathrm{c}} m(X_{t+1}|z_{t+1}))\, m(X_t|X_{t+1})\big] \end{aligned} \tag{8}$$

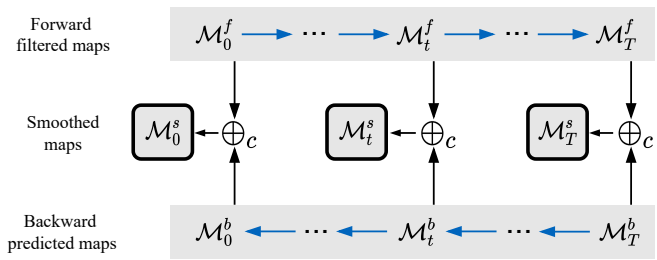


Fig. 2. Schematic overview of two-filter occupancy smoothing: Both the forward filtered maps $\mathcal{M}_t^f$ and backward predicted maps $\mathcal{M}_t^b$ are computed independently of each other. These uncorrelated maps are combined to obtain the two-filter smoothed map $\mathcal{M}_t^s$.

and the smoothed occupancy estimates of the combined two-filter concept are computed by

$$m(X_t|z_{0:T}) = m(X_t|z_{0:t}) \oplus_{\mathrm{c}} m(X_t|z_{t+1:T}) \tag{9}$$

for $t = 0, ..., T$. To compute the smoothed estimates at the interval boundaries $t = 0$ and $t = T$, we define the vacuous priors of the forward filtering and backward prediction recursions by $m(\Theta_0|z_{0:-1}) = 1$ and $m(\Theta_T|z_{T+1:T}) = 1$, respectively. This choice of the vacuous priors assigns the maximum uncertainty to the occupancy state.

The resulting conflict masses from the combination in (9) are added to the smoothed masses by heuristics based on prior knowledge (cf. Tab. I). Therein, all conflict masses arising from $m(F_t|z_{0:t})$ are assigned to $m(F_t|z_{0:T})$ since filtering includes measurements at the current time step, which is more crucial for inferring free space. Conflict masses between $m(S_t|*)$ and $m(F_t|*)$ are further distributed to $m(F_t|z_{0:T})$ to reduce erroneously assigned static mass. The remaining conflict masses involving $m(F_t|*)$ are added to $m(FD_t|z_{0:T})$ as a result of the trade-off between filtered occupancy and predicted free space. The combinations of $m(S_t|*)$ and $m(D_t|*)$ are allocated to their superset of unclassified occupancy $m(SD_t|z_{0:T})$ to preserve temporally inferred occupancies. The conflict masses between S and FD are not listed in Tab. I since they are not assigned explicitly and thus distributed by normalization. Since every recursion only needs to be processed once, the computational complexity of this algorithm is linear with the number of cells.

TABLE I
Conflict mass assignment for smoothing

Added to	**Conflict masses**
$m(F_t\|z_{0:T})$	$m(F_t\|z_{0:t})m(S_t\|z_{t+1:T})$, $m(F_t\|z_{0:t})m(D_t\|z_{t+1:T})$, $m(F_t\|z_{0:t})m(SD_t\|z_{t+1:T})$, $m(S_t\|z_{0:t})m(F_t\|z_{t+1:T})$
$m(FD_t\|z_{0:T})$	$m(D_t\|z_{0:t})m(F_t\|z_{t+1:T})$, $m(SD_t\|z_{0:t})m(F_t\|z_{t+1:T})$
$m(SD_t\|z_{0:T})$	$m(S_t\|z_{0:t})m(D_t\|z_{t+1:T})$, $m(D_t\|z_{0:t})m(S_t\|z_{t+1:T})$

IV. Velocity Smoothing

The velocity in the dynamic grid representation is estimated using particles, which are weighted by their occupancy value

to obtain the velocity in each cell [17]. As mentioned above, we compare two approaches for smoothing the velocity, which we present subsequently.

A. Velocity Smoothing by Reweighting Particles

To obtain a smoothed distribution using the particle reweighting algorithm [26], a backward recursion is appended, which adapts the particle weights from the forward recursion using future measurement information. The original particle reweighting algorithm (cf. (11)) is designed for estimating a single state vector, for which the sum of smoothed weights at each time step is normalized to 1 accordingly to the particle filtering algorithm [20, pp.116–128]. Since in [17] the particle filter estimates the states separately for each cell and the particle weights in each cell are normalized to its dynamic mass, we also normalize the reweighted particles to the smoothed dynamic mass $m(D_t|z_{0:T})$ (cf. (10)) to be consistent with [17]. For a fixed time interval between 0 and T, the backward recursion is initialized at $t = T$ by setting the smoothed weights equal to the filtered weights since no future measurements exist beyond time step T, i.e., $o^{(i)}_{T|0:T} = o^{(i)}_T$, $\forall i \in \mathcal{I}_T$. The smoothed weights are computed for $t = T-1, ..., 0$ using

$$o^{(i)}_{t|0:T} = \frac{\widetilde{o}^{(i)}_{t|0:T}}{\sum_{i\in\mathcal{I}_t} \widetilde{o}^{(i)}_{t|0:T}} m(D_t|z_{0:T}), \qquad \forall i \in \mathcal{I}_t \tag{10}$$

with the unnormalized smoothed weights from particle reweighting

$$\widetilde{o}^{(i)}_{t|0:T} = \sum_{j\in\mathcal{I}_{t+1}} o^{(j)}_{t+1|0:T} \frac{o^{(i)}_t p(\boldsymbol{x}^{(j)}_{t+1}|\boldsymbol{x}^{(i)}_t)}{\sum_{l\in\mathcal{I}_t} o^{(l)}_t p(\boldsymbol{x}^{(j)}_{t+1}|\boldsymbol{x}^{(l)}_t)}. \tag{11}$$

Analogously to the filtering application [17], we obtain the smoothed velocity vector for a cell c from particle reweighting (PR)

$$\boldsymbol{v}^c_{t|0:T,PR} = \frac{\sum_{i\in\mathcal{I}^c_t} o^{(i)}_{t|0:T} \boldsymbol{v}^{(i)}_t}{\sum_{i\in\mathcal{I}^c_t} o^{(i)}_{t|0:T}}, \tag{12}$$

where $\boldsymbol{v}^{(i)}_t \in \mathbb{R}^2$ are the particle velocities and $\mathcal{I}^c_t \subset \mathcal{I}_t$ is the set of particles in cell c. In (10) – (12), we have neglected the exclusion of particles in $\mathcal{I}_t$ moving out of the grid bounds and those newly sampled particles in $\mathcal{I}_{t+1}$.

In our approach, the particle smoother only adjusts the weights of the particles inside the cell, whereas the two-filter occupancy smoother (cf. section III) provides the smoothed dynamic mass of the cell to rate the reliability between cells. The computational complexity of the particle reweighting algorithm is $O(N^2_t)$, which results from the sum over all particle combinations in the transition step [20, p. 171].

B. Two-Filter Velocity Map Smoothing

Our second approach for velocity estimation exploits the velocity estimations on the cell level, which are computed using all particles in each cell. To this end, we apply a two-filter smoother (cf. section III) on each grid cell using the independently estimated particle velocities from both the past and future measurements. Fig. 3 schematically depicts the principle of the two-filter velocity estimation for an accelerating vehicle.

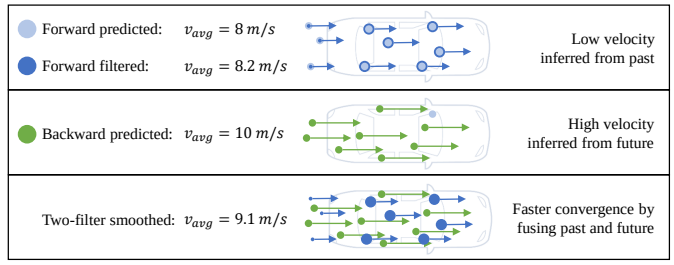


Fig. 3. Illustration of the particle velocity estimation using the two-filter smoother based on an accelerating vehicle: The dot size and the arrow length represent the weight and the absolute velocity value of the particles, respectively. Exemplary values are chosen for the average particle velocity v_{avg} to highlight the effect. While the particles tend to fall behind the vehicle (assuming constant velocity motion model [17]) in the forward prediction, the update step reduces the influence of particles outside the vehicle contour. The backward predicted estimates tend to get ahead the vehicle in the opposite direction due to a higher predicted velocity. The two-filter smoother fuses both estimates, which results in a faster convergence.

Since a combination of both particle velocity distributions by multiplication would result in zero unless two samples lie exactly at the same spot, we approximate the distributions by the mean and the dynamic mass similar to the mean and variance of Gaussian distributions. Therein, we use the corresponding dynamic mass instead of the variance of the particle velocity distribution since the variance is prone to the sampling process of new particles. In fact, the dynamic mass can be considered as a measure for the reliability of the estimated velocity and was also used in [27] for weighting cell velocity estimations for object extraction. We obtain the smoothed velocity vector for a cell c from the two-filter (TF) approach

$$\boldsymbol{v}^c_{t|0:T,TF} = \frac{m(D^c_t|z_{0:t})\boldsymbol{v}^c_{t|0:t} + m(D^c_t|z_{t+1:T})\boldsymbol{v}^c_{t|t+1:T}}{m(D^c_t|z_{0:t}) + m(D^c_t|z_{t+1:T})} \tag{13}$$

for $t = 0, ..., T$, where $\boldsymbol{v}^c_{t|0:t} \in \mathbb{R}^2$ and $\boldsymbol{v}^c_{t|t+1:T} \in \mathbb{R}^2$ denote the forward filtered and backward predicted cell velocities, respectively. Since $m(D^c_t|z_{T+1:T}) = 0$ as defined by the vacuous prior of the backward recursion, the smoothed velocity estimates of the last time step $t = T$ equals the filtered estimates such that $\boldsymbol{v}^c_{T|0:T,TF} = \boldsymbol{v}^c_{T|0:T}$. In contrast to the approach presented in section IV-A, our two-filter velocity smoother benefits from the inclusion of particle estimations from the backward pass and a lower computational complexity, which is linear with the number of cells N_{cell}, i.e., $O(N_{cell})$.

V. Experiments

In our experiments we first compare the performance of our occupancy smoothing approach with the filtered dynamic grid maps. Secondly, we evaluate the velocity orientation estimates of both proposed velocity smoothing approaches and validate the absolute velocity estimates using the motion data of a reference vehicle. We generate all results with a grid size of 680×680 cells, a cell resolution of 0.15 m $\times$ 0.15 m, and a maximum number of 100 particles per cell.

Our experimental setup includes a Hesai Pandar lidar mounted on the roof rack of the ego vehicle. The sensor provides a 3D point cloud consisting of 40 vertical layers with a resolution of 0.33° and 1° depending on the elevation angle. A horizontal resolution of 0.2° is obtained with a sensor rotation rate of 10Hz. To obtain validation data, the reference vehicle is equipped with a DGPS system (based on Real-Time Kinematic technology) from OxTS for velocity estimation. We have visualized the results for a section of the evaluated sequence including the front camera view and raw lidar data in the video attachment of this paper.

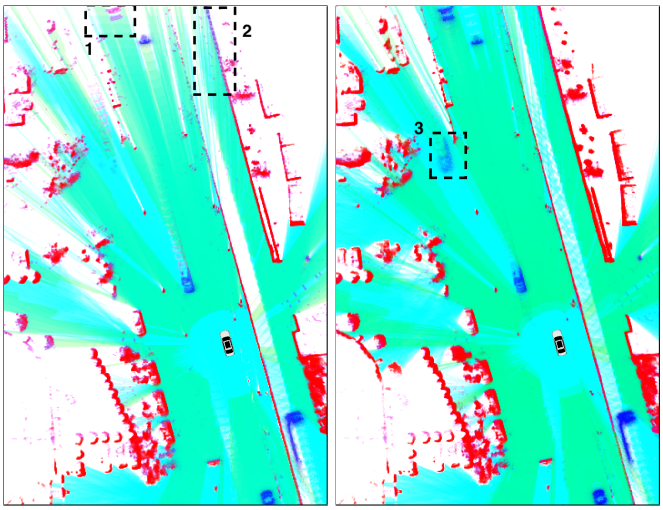
A. Comparison of Occupancy Estimates

To highlight the enhancement of the occupancy grids through our approach, Fig. 4 shows a qualitative comparison between the filtered and smoothed dynamic occupancy grid maps. As can be seen in the upper left corner of Fig. 4a, the filtered occupancy grid suffers from high uncertainties in newly observed areas, i.e., the corresponding grid cells have not yet been classified correctly in the forward recursion. By incorporating the backward recursion, our smoothed map (Fig. 4b) exhibits an improved classification of the cells in this area, which is especially noticeable for static cells originating from buildings or parked vehicles.

In Fig. 4a, the region "1" marked by a dashed box corresponds to an oncoming vehicle entering the grid. In the forward recursion, the associated occupied cells are unclassified since several timesteps are required to infer dynamic occupancy. The smoothed grid, however, clearly identifies the cells as dynamic as soon as the object enters the grid. Region "2" in Fig. 4a shows a newly appearing part of a guardrail which is partly misclassified as dynamic occupancy. This is a common issue for dynamic grids when only lidar data is used, since elongated objects parallel to the ego velocity orientation tend to get assigned the same speed as the ego vehicle. The smoothed map identifies this part of the guardrail correctly as static occupancy, even without remedying with radar velocity measurements [16]. As indicated by region "3" in Fig. 4b, an object occluded during multiple time steps can be recovered using the backward recursion if the object is observed afterwards. In this way, the object existence time can be extended and the overall situational awareness is raised.

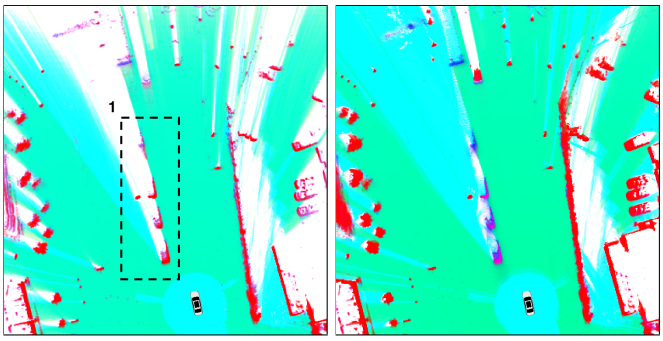
Fig. 5 presents two more scenarios involving intersections. Therein, the region marked with "1" in Fig. 5a includes vehicles which were initially standing when the ego vehicle approaches the intersection. While forward filtering has not yet seen the vehicles moving, the smoothed cell estimates mostly infer a dynamic occupancy from future measurements, which improves the estimation of dynamic states when the vehicles start moving. Region "2" in Fig. 5b shows a challenging situation with three vehicles successively turning right. The noise in the wake of the objects results from the initialization of new dynamic occupancy, which is reduced by offline smoothing due to an improved classification of the cell occupancy.

In [17], subsequent perception tasks such as object tracking are performed on classified scan grids. Since lidar measurements do not provide dynamic information and the dynamic grid data are correlated due to filtering over time, classified scan grids were introduced, which consist of the unclassified

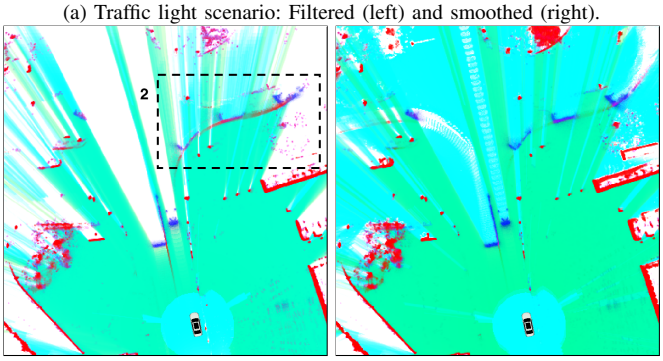


(a) Filtered dynamic grid. (b) Smoothed dynamic grid.

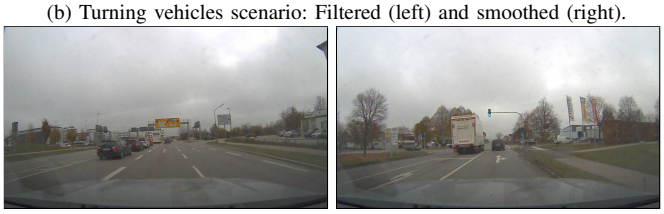
Fig. 4. Comparison of dynamic occupancy grid map: Static occupancy is visualized in red, free space in green, dynamic occupancy in blue, unclassified occupancy in pink, passable area in cyan, and an unknown state in white [17].



(a) Traffic light scenario: Filtered (left) and smoothed (right).

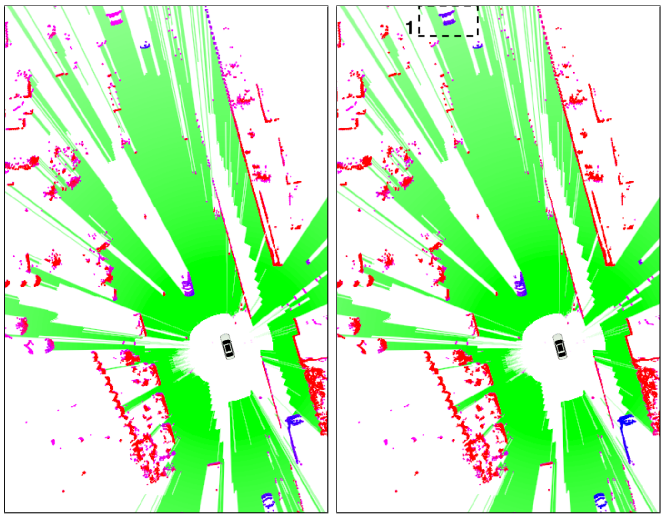


(b) Turning vehicles scenario: Filtered (left) and smoothed (right).



(c) Camera views: Traffic light (left) and turning vehicles (right) scenario.

Fig. 5. Comparison of dynamic occupancy grid maps for two intersection-related scenarios: The first scenario in (a) shows the ego vehicle approaching an intersection with vehicles initially standing at the traffic light. The scenario in (b) shows three vehicles turning right at the intersection.



(a) Filtered classified scan grid. (b) Smoothed classified scan grid.

Fig. 6. Comparison of classified scan grid: Static occupancy is visualized in red, free space in green, dynamic occupancy in blue, unclassified occupancy in pink, and an unknown state in white [17].

occupancy from lidar measurements classified using dynamic grids. These are used as an uncorrelated input for the extraction and tracking of objects to prevent multi-filtering of occupied cells. Fig. 6 shows the filtered and smoothed classified scan grids corresponding to the situation in Fig. 4. As can be seen in the upper left corner, cells representing the newly appearing static environment obtain a more accurate classification. Furthermore, region "1" in Fig. 6b includes the oncoming vehicle entering the grid area, whose cells are clearly classified as dynamic in the smoothed case. Hence, an earlier detection and extraction of moving objects at time of appearance can be achieved based on the smoothed classified scan grids.

To provide a statistical evaluation of our approach, we quantitatively compare the map estimates with respect to occupancy classification. Analogously to [17], we classify all cells, for which the current measurement exceeds a minimum occupancy evidence, binarily into static and dynamic occupancy using

$$\Lambda = \begin{cases} \{S\} & \text{if } m(S_t^c) \geq \psi_\Lambda, \\ \{D\} & \text{otherwise}, \end{cases} \tag{14}$$

with the classification threshold ψ_Λ. We evaluate these outcomes of the classification using varying ψ_Λ with manually labeled ground-truth data. Fig. 7 compares the classification performance using receiver operating characteristic (ROC) curves and the corresponding area under the curve measure for a sequence of 700 frames. For our scenario, both the ROC curves and the area under the curves show that our proposed offline smoothing approach outperforms the online filtering approach, which confirms the previous qualitative results.

B. Comparison of Velocity Orientation Estimates

To evaluate our approaches for smoothing velocity estimates, we conduct a comparison using velocity orientation grids [17], which visualize the estimated velocity orientation

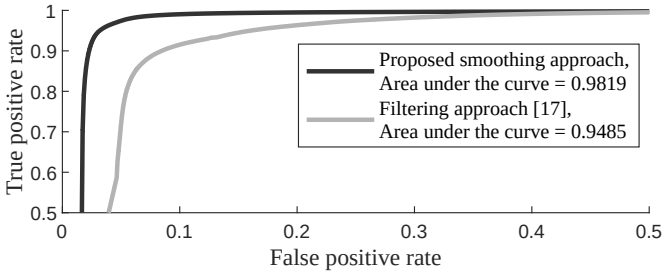


Fig. 7. Statistical comparison of the proposed smoothing approach against the filtering approach from [17] using ROC curves, where 'positive' denotes the binary classification of dynamic cell occupancy.

of each cell. To this end, we adopt the HSV (hue, saturation, and value) color coding from [17] defined by

$$\text{HSV}^c = \left(\arctan\left(\frac{v_{y,t}^c}{v_{x,t}^c}\right), m(D_t^c), 1 - m(S_t^c) \right). \tag{15}$$

Fig. 8 depicts the filtered and both smoothed velocity orientation grids for the same situation as shown in Fig. 4. As marked by dashed boxes in Fig. 8a, regions "1" and "2" contain cells of static objects estimated as dynamic. The colors of the cells indicate a movement parallel to the longitudinal axis of the nearby structure, which can be traced back to the aforementioned difficulties in classifying cell occupancies of elongated objects. Both smoothing approaches can resolve these issues using future information and correctly infer a static occupancy.

The region "3" in Fig. 8b and Fig. 8c indicates an occluded vehicle as discussed previously. This example shows that our approach is not only able to recover the occupancy in occluded areas but additionally provides an accurate estimate of the velocity orientation of the corresponding grid cells.

In region "4", the velocity orientation estimates of the cells corresponding to the object differ for both smoothing approaches. Compared to the filtered estimates, particle reweighting obviously provides a more intense coloring of the object cells due to an increased amount of dynamic mass after smoothing. However, it suffers from ambiguous velocity orientation estimations, which can be explained by a low number of good particles since filtering does not always generate particles with reasonable velocity values in the beginning. Thus, one also increases the importance of unsuitable particles in case no appropriate particles exist. In contrast, two-filter smoothing performs better by including the backward recursion which makes it less prone to initialization errors of the forward recursion.

Since velocity estimates are important for clustering cells and extracting objects as well as for reasoning dynamic occupancy, our approach enhances the performance of the grid generation process and provides more reliable data for subsequent perception tasks.

C. Evaluation of Velocity Estimates with DGPS Reference

In this section, we further validate the absolute velocity estimates using DGPS data from our reference test vehicle. To this end, we first evaluate the velocity distribution of particles

(a) Filtered. (b) Smoothed by reweighting particles. (c) Smoothed by two-filter.

Fig. 8. Comparison of velocity orientation grid: HSV color coding given by (15) results in black for static cells and in the color corresponding to the cell velocity orientation according to the color circle in the bottom right corner [17].

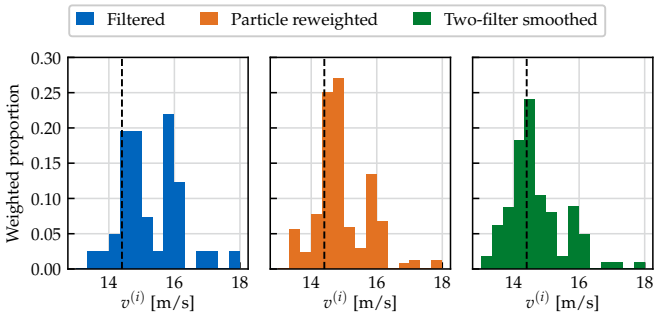


Fig. 9. Comparison of the absolute particle velocity distribution in a single grid cell, where the true velocity of the corresponding reference object is 14.4 m/s (marked by dashed line).

by exemplarily choosing a representative cell which we have manually associated to our reference object. Fig. 9 shows the weighted histograms of the absolute particle velocities in this grid cell. For the results of filtering and particle reweighting, the particle states are equivalent, but are assigned with different weights. The histogram for two-filter smoothing contains the particles from both the forward filtering and from backward prediction.

The filtered particle distribution exhibits two apparent modes around $14.5m/s$ and $16m/s$ with a spread between $13m/s$ and $18m/s$. It can be observed that both smoothing approaches increase the weights around the mode corresponding to the true velocity value and reduce the influences of the second mode as well as the weights of the outliers on the right side. Therein, the additional particles from backward prediction provide a better centered distribution of the two-filter smoothed estimates around the true velocity value compared to the particle reweighted estimates. This example shows that smoothing is able to reduce the influence of erroneous particles and to improve the intra-cell particle velocity distribution.

To demonstrate the benefits of our approach for object tracking [27], we evaluate the accuracies of the velocity estimations for all cells corresponding to the reference object. To obtain these linked cells, we use the cell-to-object association from [19]. Therein, the particles have labels, which can be explicitly assigned to objects. Using these labels, the cells are associated to the object if a minimum ratio of particles in the cell is assigned to it, while the dynamic mass of the cell exceeds a threshold at the same time. For a sequence of approximately 70 seconds, Fig. 10 compares the absolute velocity computed by the cells with the DGPS velocity of the reference object. For the evaluation of both smoothing approaches, we adopted the same cells as we have extracted for filtering.

In the presented sequence, the reference vehicle first accelerates, then cruises with almost constant velocity, and brakes in the end until it stops. Therein, particle reweighting provides slightly better velocity estimates than filtering with a lower variance. However, both filtering and particle reweighting are subject to a delay in the acceleration and braking phase. While the delay for filtering is plausible, the delay for particle reweighting might be a result of a fixed set of cells adopted from filtering together with a fixed set of particle states, which restricts the variability to counteract the delay. In contrast, two-filter smoothing can almost fully compensate the delay with backward predicted velocities, which behave contrarily to the forward filtered estimates. The increase of the error and variance in the end for the two-filter can be explained by a standing reference vehicle in the first steps of the backward recursion.

Overall, velocity smoothing with two-filter has a lower computational complexity compared to particle reweighting and outperforms both filtering and particle reweighting in terms of estimation accuracy. This makes it a reasonable approach for generating ground truth.

VI. Conclusions

This paper proposes an offline approach to generate accurate dynamic grids for automotive perception. By using past and future measurements, we improve the classification of static and dynamic cell occupancy and provide a more precise cell velocity estimation. The resulting grid representation is more consistent ensuring a more reliable application of subsequent perception algorithms. By using real-world sensor data, we

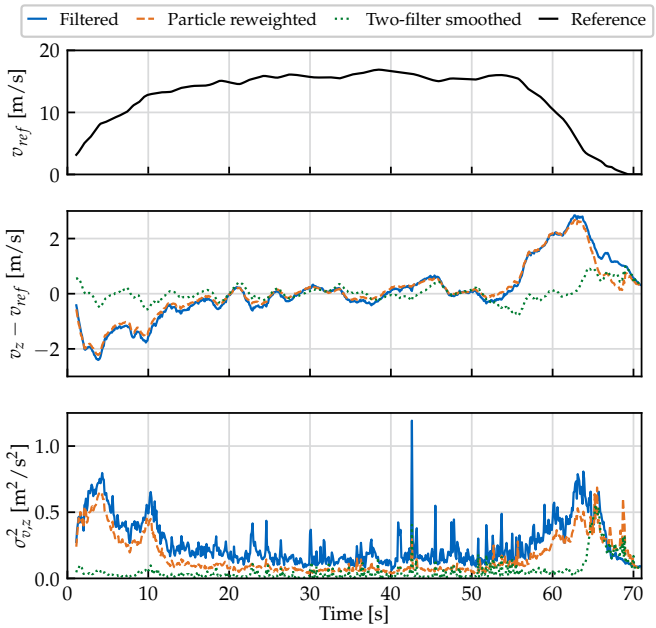


Fig. 10. Performance validation of the absolute velocity computed by the cells associated to the reference object: v_{ref} is the absolute DGPS velocity of the reference object, $v_z - v_{ref}$ is the error of the weighted means, and $\sigma^2_{v,z}$ is the weighted variance (cf. [27]) of the velocity estimations in the associated cells.

demonstrated the benefits of our concept compared to the online method and thus can justify its potential to be used for ground truth generation. To exploit these advantages for validating online tracking algorithms, we would like to generate ground-truth object data by extracting objects from the offline generated dynamic grids in the future.

References

[1] J. E. Stellet, M. R. Zofka, J. Schumacher, T. Schamm, F. Niewels, and J. M. Zöllner, "Testing of advanced driver assistance towards automated driving: A survey and taxonomy on existing approaches and open questions," in *Proc. of the 18th IEEE International Conference on Intelligent Transportation Systems*, 2015, pp. 1455–1462.

[2] M. Brahmi, "Reference systems for environmental perception," in *Automotive Systems Engineering*. Springer, 2013, pp. 205–221.

[3] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "SemanticKITTI: A dataset for semantic scene understanding of lidar sequences," in *Proc. of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9296–9306.

[4] J. E. Stellet, L. Walkling, and J. M. Zöllner, "Post processing of laser scanner measurements for testing advanced driver assistance systems," in *Proc. of the 19th International Conference on Information Fusion*, 2016, pp. 1999–2006.

[5] E. Ye and M. Althoff, "Model-based offline vehicle tracking in automotive applications using a precise 3D model," in *Proc. of the 22nd IEEE International Conference on Intelligent Transportation Systems*, 2019, pp. 1128–1135.

[6] M. Spencer, R. Katz, and U. Lages, "Forward-backward object tracking for generation of reference scenarios based on laser scan data," in *Proc. of the 21st Intelligent Transport Systems World Congress*, 2014, pp. 3099–3105.

[7] D. Stumper, F. Gies, S. Hoermann, and K. Dietmayer, "Offline object extraction from dynamic occupancy grid map sequences," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2018, pp. 389–396.

[8] B. Yu and E. Ye, "Track-before-detect labeled multi-Bernoulli smoothing for multiple extended objects," in *Proc. of the 23rd International Conference on Information Fusion*, 2020, pp. 1233–1240.

[9] E. Ye, P. Spiegel, and M. Althoff, "Cooperative raw sensor data fusion for ground truth generation in autonomous driving," in *Proc. of the 23rd IEEE International Conference on Intelligent Transportation Systems*, 2020, pp. 1498–1504.

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

[11] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*, ser. Intelligent robotics and autonomous agents series. Cambridge: MIT Press, 2005.

[12] C. Coué, C. Pradalier, C. Laugier, T. Fraichard, and P. Bessière, "Bayesian occupancy filtering for multitarget tracking: An automotive application," *The International Journal of Robotics Research*, vol. 25, no. 1, pp. 19–30, 2006.

[13] M. K. Tay, K. Mekhnacha, M. Yguel, C. Coué, C. Pradalier, C. Laugier, T. Fraichard, and P. Bessière, "The Bayesian occupation filter," in *Probabilistic Reasoning and Decision Making in Sensory-Motor Systems*, ser. Springer Tracts in Advanced Robotics. Springer, 2008, vol. 46, pp. 77–98.

[14] R. Danescu, F. Oniga, and S. Nedevschi, "Modeling and tracking the driving environment with a particle-based occupancy grid," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1331–1342, 2011.

[15] G. Tanzmeister, J. Thomas, D. Wollherr, and M. Buss, "Grid-based mapping and tracking in dynamic environments using a uniform evidential environment representation," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2014, pp. 6090–6095.

[16] G. Tanzmeister and D. Wollherr, "Evidential grid-based tracking and mapping," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1454–1467, 2017.

[17] S. Steyer, G. Tanzmeister, and D. Wollherr, "Grid-based environment estimation using evidential mapping and particle tracking," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 3, pp. 384–396, 2018.

[18] ——, "Object tracking based on evidential dynamic occupancy grids in urban environments," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2017, pp. 1064–1070.

[19] S. Steyer, G. Tanzmeister, C. Lenk, V. Dallabetta, and D. Wollherr, "Data association for grid-based object tracking using particle labeling," in *Proc. of the 21st IEEE International Conference on Intelligent Transportation Systems*, 2018, pp. 3036–3043.

[20] S. Sarkka, *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

[21] A. P. Dempster, "Upper and lower probabilities induced by a multivalued mapping," *The Annals of Mathematical Statistics*, vol. 38, no. 2, pp. 325–339, 1967.

[22] G. Shafer, *A mathematical theory of evidence*. Princeton, NJ: Princeton Univ. Press, 1976.

[23] T. Reineking, "Belief functions: Theory and algorithms," Ph.D. dissertation, Universität Bremen, 2014.

[24] E. Ramasso, M. Rombaut, and D. Pellerin, "Forward-Backward-Viterbi procedures in the transferable belief model for state sequence analysis using belief functions," in *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, ser. Lecture Notes in Computer Science. Springer, 2007, vol. 4724, pp. 405–417.

[25] P. Smets, "Belief functions: The disjunctive rule of combination and the generalized Bayesian theorem," *International Journal of approximate reasoning*, vol. 9, no. 1, pp. 1–35, 1993.

[26] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[27] S. Steyer, C. Lenk, D. Kellner, G. Tanzmeister, and D. Wollherr, "Grid-based object tracking with nonlinear dynamic state and shape estimation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 7, pp. 2874–2893, 2019.