Data-Driven Occupancy Grid Mapping using Synthetic and Real-World Data*

Raphael van Kempen[©] Institute for Automotive Engineering RWTH Aachen University Aachen, Germany raphael.vankempen@ika.rwth-aachen.de

Bastian Lampe Institute for Automotive Engineering RWTH Aachen University Aachen, Germany bastian.lampe@ika.rwth-aachen.de

Lennart Reiher Institute for Automotive Engineering RWTH Aachen University Aachen, Germany lennart.reiher@ika.rwth-aachen.de

Timo Woopen[®] Institute for Automotive Engineering RWTH Aachen University Aachen, Germany timo.woopen@ika.rwth-aachen.de

Till Beemelmanns Institute for Automotive Engineering RWTH Aachen University Aachen, Germany till.beemelmanns@ika.rwth-aachen.de

Lutz Eckstein Institute for Automotive Engineering RWTH Aachen University Aachen, Germany lutz.eckstein@ika.rwth-aachen.de

Abstract—In perception tasks of automated vehicles (AVs) data-driven have often outperformed conventional approaches. This motivated us to develop a data-driven methodology to compute occupancy grid maps (OGMs) from lidar measurements. Our approach extends previous work such that the estimated environment representation now contains an additional layer for cells occupied by dynamic objects. Earlier solutions could only distinguish between free and occupied cells. The information whether an obstacle could move plays an important role for planning the behavior of an AV. We present two approaches to generating training data. One approach extends our previous work on using synthetic training data so that OGMs with the three aforementioned cell states are generated. The other approach uses manual annotations from the nuScenes [1] dataset to create training data. We compare the performance of both models in a quantitative analysis on unseen data from the realworld dataset. Next, we analyze the ability of both approaches to cope with a domain shift, i.e. when presented with lidar measurements from a different sensor on a different vehicle. We propose using information gained from evaluation on realworld data to further close the reality gap and create better synthetic data that can be used to train occupancy grid mapping models for arbitrary sensor configurations. Code is available at https://github.com/ika-rwth-aachen/DEviLOG.

Index Terms—AD, perception, simulation, deep learning

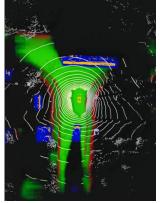
I. INTRODUCTION

An automated vehicle can determine the drivable space in the static environment by localizing itself on a high-definition map. These HD maps describe the exact road geometry and traffic rules applying to each lane.

However, these maps can be outdated so that it becomes necessary to substitute them with measurement-based repre-

*This research is accomplished within the project "UNICARagil" (FKZ 16EMO0284K). We acknowledge the financial support for the project by the Federal Ministry of Education and Research of Germany (BMBF).





(a) Trained on Synthetic Data

(b) Trained on nuScenes

Fig. 1: The left OGM was predicted by a model trained on synthetic data and the right OGM was predicted by a model trained on labels generated from annotations of the nuScenes [1] dataset. Green indicates belief mass for free, red for statically occupied and blue for dynamically occupied cells. The lidar point cloud as input data is superimposed and colored by the class annotations, i.e. vehicles are orange, drivable surface is yellow and occupied space is grey.

sentations to determine the actual drivable space. Grid-based environment representations are particularly suitable for this as they discretize a defined area around the vehicle into cells that can be aligned with the HD map. Distances measured e.g. by a lidar sensor can be used to assign an occupancy state to each cell in an OGM. Such OGMs are also used in the UNICARagil project [2], [3], where our approach is also

developed. The trajectories planned by an AV are constrained by static obstacles (e.g. road boundaries, barricades) as well as potentially dynamic obstacles (e.g. cars, pedestrians). While static obstacles always constrain the drivable space, the impact of dynamic obstacles depends on their future motion. Hence it is necessary to predict their future behavior. This is often achieved using object-based approaches estimating the motion states of objects in the vehicle's environment. Several approaches for detection, tracking and prediction of objects using cameras, lidar and radar sensors have been published in the past years. However, object-based algorithms are often trained on a fixed set of classes and objects that do not appear in the training dataset may not be detected or falsely classified. On the other hand, OGMs can represent obstacles of arbitrary shape and class. For object-based detectors a threshold usually needs to be found that minimizes both the false-positive-rate and the false-negative-rate. If free space can be estimated with high certainty using an OGM, which is combined with the object-based representation, it is possible to allow a higher false-positive-rate to further reduce the false-negative-rate and the robustness of perception can be increased. Semantic grid maps (SGMs) extend the task of occupancy grid mapping to assigning semantic classes (e.g. road, sidewalk, pedestrian, car) to each cell [4], [5]. While the aforementioned works focus on creating semantic grid maps for a wide variety of classes, we extended our previous approach [6] to the three classes free, statically occupied and dynamically occupied. Our goal is to estimate OGMs with very high performance on these classes and to increase the robustness of perception by fusing them with object-based representations. As detecting and estimating the actual motion of objects is a more complex task, we want to separate this into another module to increase the robustness of the module developed in this work. To enable sensor fusion and cooperative perception [7], it is necessary to quantify uncertainties in the outputs of a perception algorithm, which is implemented in our approach by predicting evidential OGMs.

A conventional way to generate OGMs is using a geometric inverse sensor model (ISM) with range measurements [8]. A grid map obtained from one measurement is usually quite sparse and does not contain semantic information. A denser OGM including dynamic information can be obtained by tracking it over time [9] or by using deep ISMs, which use supervised learning to estimate OGMs or SGMs from measurements. Some approaches use cross-modal training [10] but suffer from the restrictions of the underlying geometric ISM as the generated labels are just an estimation of the ground-truth. Other works use manually annotated datasets [4], [11]–[13] to generate ground truth OGMs or SGMs, e.g. using semantically labeled point clouds or bounding boxes. These manually annotated datasets have to be created with high effort and can only represent the sensor setup and environment conditions that were recorded. Our approach presented in [6] uses advanced sensor models to generate synthetic training data to train a deep neural network to predict OGMs from lidar measurements, which outperforms a geometric approach. This approach allows generating training data for arbitrary sensor configurations with low effort. While the presented qualitative analysis on real-world data was already quite promising, a quantitative analysis on real-world data was still due and is presented in this work among other things.

In this work, we extend our previous approach to predict OGMs with cell states *free*, *statically occupied* and *dynamically occupied*. Additionally, we provide a quantitative performance analysis on the real-world nuScenes dataset [1]. The performance of the model trained on synthetic data is compared to a model trained on labels generated from the manual annotations of the dataset. Finally, we compare the transferability of both approaches to a new sensor configuration by applying them to one of our research vehicles. As a result, we propose a methodology of using real-world datasets to identify blind spots in the simulation and improve the domain adaptation of models trained on synthetic data to real-world measurements. By making our baseline and evaluation algorithm open-source, we enable others to compare their approaches with ours.

II. BACKGROUND

This section gives an overview of current approaches for computing OGMs and SGMs. Evidence theory, the basis for the output of the proposed neural network, is briefly introduced and we will give an short overview on the state of the art for using synthetic training data for automotive perception tasks.

A. Occupancy Grid Maps

An OGM [14] is a grid-based representation of the vehicle's environment. Each cell in this grid contains occupancy information. First approaches used a binary Bayes filter to generate an OGM from distance measurements. Recent approaches are often based on evidence theory, which is described in Section II-C. This allows to distinguish unobservable cells from cells with conflicting evidence. Hand-crafted geometric approaches have often been used in the context of OGMs. They assume a ground model to separate drivable from occupied area but have problems in non-flat and dynamic environments. Recent approaches use deep learning models to predict OGMs from distance measurements. Wirges et al. [15] use sequential sparse measurement OGMs to create a dense label OGM to be predicted from one measurement OGM. In [16], a model is trained to predict future OGMs using lidar measurements as input data. The approach is still based on a naive model that solely treats all reflection points in a specific height as obstacles. Bauer et al. [10] transform radar data into an OGM with two channels containing static and dynamic detections. [17] and [11] both use sequential lidar point clouds to create label OGMs including velocity information from tracked or annotated objects. The authors of [13] create label OGMs with occupancies estimated by a particle-based approach and velocities from bounding box annotations and train a recurrent neural network to predict dynamic OGMs from measurement OGMs that were created using a geometric ISM.

B. Semantic Grid Maps

SGMs assign semantic information on the obstacle class to each cell instead of occupancy information. They usually do not quantify the uncertainty of the classification. Wu et al. [18] presented a model that predicts SGMs with information on occupancy, object class and motion information in one shot. The input data consists of sequential lidar point clouds represented as binary matrices stacked along a third dimension. Annotations from the nuScenes dataset [1] are used to train the model. The authors of [4], [19] and [12] combine multiple semantically segmented lidar point clouds from manually annotated datasets to create label SGMs that are to be predicted by the trained model from sparse lidar measurements.

C. Evidence Theory and Subjective Logic

In contrast to Bayesian probability theory, in the framework of **Evidence Theory** [20] it is possible to consider epistemic uncertainty, which has already been used with OGMs before [9], [15]. Each cell in the OGM contains belief masses mfor the possible cell states. In this work, these are the mutually exclusive cell states free (F), statically occupied (O_s) and dynamically occupied (O_d) forming the frame of discernment Θ.

$$\Theta = \{F, O_s, O_d\} \tag{1}$$

During evaluation, we will consider the reduced power set

$$\mathcal{E} = \{ \{F\}, \{O_s\}, \{O_d\}, \{O_s, O_d\}, \Theta \}$$
 (2)

which extends the FOD with the two additional states statically or dynamically occupied and unknown. The belief mass distribution m(A) can also be interpreted as a subjective opinion in the framework of Subjective Logic [21]. Motivated by the work of Sensoy et al. [22] showing that these parameters can be predicted by a neural network to quantify uncertainty in classification tasks, we created an deep neural network to predict evidential OGMs and use their proposed loss function in our previous [6] and in this work. Dempster's Rule of Combination [23] can be used to combine evidence from different sources, e.g. evidence for the occupancy state of a grid cell. It allows to compute the joint mass $m_{12}(X)$ for a hypothesis X, which is an element in the power sets of the combinable FODs 2^{Θ_1} and 2^{Θ_2} of the different sources.

D. Synthetic Training Data

The performance of deep learning models usually drops when presented with test data from a different domain. For perception tasks, this comprises different countries or weather conditions as well as different sensor models or mountings. A domain shift is also caused by the so-called reality gap when a model that has only seen synthetic data during training is presented with test data from the real world. Reiher et al. [24] presented a methodology for the transformation of camera images into an SGM using synthetic training data. They use segmented camera images predicted by a model trained on real-world data as intermediate representation and thus successfully bridge the reality gap. Zeng et al. [25]

propose a network architecture that improves generalization of lidar object detection models to other domains. They observed that BEV features are more transferable, which is supported by our previous results [6], where we have shown that it is possible to train a model for evidential occupancy grid mapping from lidar measurements using synthetic data only and obtain qualitatively promising results on real-world test data, as well. In this work, we will present an extension of this algorithm that separates static from dynamic obstacles and evaluate the performance quantitatively on the nuScenes dataset [1].

On the other hand, sensor simulation is also improving. While Virtual Test Drive (VTD) [26] uses a physics-based sensor model, the authors of [27] propose a deep neural network that predicts realistic from synthetic lidar points clouds.

III. DATA-DRIVEN OCCUPANCY GRID MAPPING

We treat occupancy grid mapping as a supervised learning problem and present our neural network architecture as well as two methods for generating training data. One uses synthetic data generated using simulation and the other one uses labels generated from a manually annotated dataset.

A. Network Architecture

Our network architecture is based on PointPillars [28], which performs well for object detection in lidar point clouds. We use the same feature encoding layer and an adapted version of the CNN backbone. The prediction heads are replaced by our Evidential Prediction Head [6], which is extended to three layers in this work to comprise evidence for the possible cell states $A \in \Theta$. The model is trained with the Expected Mean Squared Error loss function proposed by Sensoy et al. [22]. It is possible to convert evidence $e_A \geq 1$ to a subjective opinion (b, u). Here, $K = |\Theta|$ is the number of classes and $S = \sum_{A \in \Theta} \alpha_A$ is the Dirichlet strength.

$$b_A = \frac{e_A}{S} \tag{4}$$

$$\alpha_A = e_A + 1, \qquad A \in \Theta$$

$$b_A = \frac{e_A}{S}$$

$$u = \frac{K}{S}$$
(3)
$$(4)$$

This belief masses $b_A \in [0,1]$ are used in the evidential OGM predicted by our network.

B. Synthetic Training Data from Simulation

As already in [6], we use VTD [26] to create simulation scenarios in an urban 3D environment including static obstacles (e.g. traffic lights, benches, poles) as well as dynamic objects (e.g. pedestrians, bicycles, vehicles) from a large catalogue of vehicle types. The simulated ego vehicle is of a similar size and type as the one used to record the test dataset and uses a vehicle dynamics model to achieve e.g. realistic roll and pitch angles when accelerating or turning. It is equipped with a virtual lidar sensor with a pose similar to the real vehicle. Laser beams are simulated using ray casting in 900 horizontal directions on 32 vertical layers similar to the real sensor. A

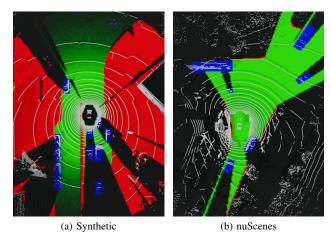


Fig. 2: Training samples are generated using simulation and from the nuScenes dataset [1]. One sample consists of a lidar point cloud (grey) and an OGM, where green indicates belief mass for *free*, red for *statically occupied* and blue for *dynamically occupied* cells.

second virtual lidar sensor with the same pose and field of view but using 3000 instead of 32 vertical layers creates denser lidar point clouds. These synthetic point clouds contain information about the material, which caused the reflection, which is interpreted as evidence for the occupancy state of the grid cell containing the reflection point and its surrounding cells in the label OGMs. In contrast to our previous approach [6], only reflections on legally drivable material (i.e. asphalt, road marks) contributes a belief mass of b(F) = 0.1 and all reflections on non-drivable material (e.g. sidewalks, buildings) contribute $m(O_s) = 0.1$. By combining all belief masses assigned to one cell using Dempster's Rule of Combination, the uncertainty in the created grid map increases with higher distance from the sensor due to a lower density of the lidar rays affecting one cell. In addition to this, a belief mass $m(O_d) = \frac{1}{|C|} \sum_c m_c(O_s)$ is assigned to all cells covered by dynamic objects, where $C = \{c_i\}$ is the set of cells occupied by the object. However, information that cannot be inferred from the input data should not be contained in the labels. Thus, only objects that were hit by a minimum of 20 simulated laser beams are processed. One such training sample is shown in Figure 2a.

C. Training Data from Annotations

Our second method uses the manually annotated nuScenes dataset [1] to create training samples consisting of lidar point clouds and OGMs. While raw lidar measurements can be directly obtained from the dataset, OGM labels have to be generated from the annotations available. We create OGMs of the same size and resolution as in the previous method and assign a belief mass $m(O_d)=1.0$ to all cells covered by objects that are represented by a minimum of 20 points in the lidar point cloud. Additionally, nuScene's map expansion is used to assign a belief mass m(F)=1.0 to all cells mapped

as drivable surface and not occupied by objects. All other cells are assigned a belief mass $m(O_s)=1.0$. As also in the previous method, we want to respect observability and ensure that labels contain only information that can be inferred from the input data. Using two-dimensional ray casting from the sensor to the grid borders, all cells behind static or dynamic obstacles are assigned a maximum uncertainty $m(\Theta)=1$, with the exception of all cells covered by dynamic objects as it is assumed that their size can be inferred from the input data if they are represented by at least 20 reflection points in the measurement. An example of such a label is shown in Figure 2b.

IV. EXPERIMENTAL SETUP AND RESEARCH OUESTIONS

Using the methods described above, 10.000 synthetic training samples as well as 30.000 samples using nuScenes' training split were generated. We used the same neural network architecture as described in [6] to predict OGMs of $81.92\,m$ length, $56.32 \, m$ width and a cell size of $32 \, cm \times 32 \, cm$. A minimum loss was achieved after approx. 50 epochs of training for both models. Their performance is evaluated on 6000 samples from nuScene's validation split using the metrics described next. The ground-truth OGMs for evaluation are created as described in Section III-C. As is apparent in Figure 2, the synthetic labels contain significantly more static cells than the labels created from nuScenes. To get comparable results, only cells with a known state (i.e. $m(\Theta) < 0.5$) in the ground-truth OGM are evaluated. As there is no uncertainty quantification in these labels, we use a threshold of $m_{\vartheta} = 0.5$ to classify cells in the predicted OGM into classes free, statically occupied and dynamically occupied. This allows us to compute the **Precision** $P_A = \frac{TP}{TP+FP}$ and **Recall** $R_A = \frac{TP}{TP + FN}$ from the numbers of true positive (TP), false positive (FP) and false negative (FN) cell classifications for the relevant states $A \in \mathcal{E}$. By analyzing these metrics we want to answer the following research questions:

- Is it possible to train a deep neural network to predict an evidential OGM, which distinguishes between statically and dynamically occupied areas, when presented with a single lidar measurement?
- If such a model was trained with synthetic data only, is it possible to perform a domain shift and achieve a sufficient performance on real-world data, as well?
- If such a model was trained with annotated data, how well does it perform when presented with unseen data from the same dataset?
- Which of both approaches generalizes better to new sensor configurations, i.e. which model performs better when presented with input data from another sensor on another vehicle?

V. RESULTS AND DISCUSSION

First, we analyze the performance of both models quantitatively on validation data from the nuScenes [1] dataset. Then we will perform a domain shift and use both approaches

	P_F	R_F	P_{O_s}	R_{O_s}	P_{O_d}	R_{O_d}	$P_{O_{sd}}$	$R_{O_{sd}}$
A	0.98	0.34	0.06	0.44	0.55	0.64	0.23	0.73
В	0.99	0.71	0.33	0.45	0.85	0.82	0.64	0.72

TABLE I: Quantitative evaluation of both approaches on unseen real-world data. *Model A* uses synthetic training data and *model B* uses labels created from manual annotations.

with lidar measurements recorded with our research vehicle to evaluate the generalization capability of both approaches qualitatively. *Model A* is trained using synthetic data and *model B* is trained using labels generated from annotations.

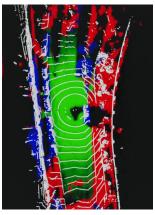
A. Evaluation on nuScenes

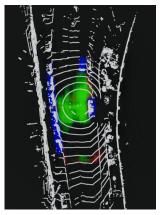
Figure 1 shows predictions of both trained models on lidar measurements from unseen data of nuScene's validation split. Both models successfully predict evidential OGMs and take observability into account. Both approaches have also learned to estimate statically occupied and dynamically occupied areas where also the dimensions of dynamic objects are being estimated. The most noticeable difference is the number of statically occupied cells, which results from the different methods for creating the labels. As the labels from synthetic data are created using a simulated lidar sensor, the field of view can be estimated better than with the ray casting approach used to create labels from annotations. This is also reflected in the quantitative evaluation and explains the low precision for statically occupied cells in Table I. The table summarizes the precision and recall for the cell states free (F), statically occupied (O_s) , dynamically occupied (O_d) and for the combined state statically or dynamically occupied (O_{sd}) that were achieved with both approaches on the validation data. A high precision of 0.98 for model A and 0.99 for model B is achieved for free space estimation, while the recall is considerably lower. A qualitative analysis shows that free space is missed especially in higher distance from the sensor where lidar measurements are only sparse. This makes it hard to estimate the cell states from a single lidar measurement and could be solved by tracking the OGM over time. The higher recall of model B can be explained by the higher diversity of road conditions in real-world data. E.g. elevation is not represented yet in the simulated scenarios so that free space cannot be estimated correctly at inclinations. While both models have a similar recall when combining statically and dynamically occupied cells, model B can better distinguish dynamic from static obstacles. This can be explained with the higher diversity of dynamic and static objects in the real world. However, the evaluation on an annotated dataset allows to detect missed object types so that they can also be modeled in simulation.

B. Evaluation of Generalization After Domain Shift

We also want to analyze the ability of both presented methods to generalize to a new domain, in particular when used with a new sensor configuration. We tested both approaches our research vehicle having a *Velodyne VLP32C* mounted on

its roof. For the approach relying on synthetic training data, it is possible to generate new training data using a simulated vehicle and sensor setup similar to our research vehicle with low effort. This is not the case for a manual annotated dataset so that the performance of a model trained on this data is compared to the model trained on annotations from nuScenes. Figure 3 shows predictions of both models. It is apparent that the distribution of reflection points differs from the sensor used in the nuScenes dataset (cf. Figure 1b). The model trained on data from this dataset shows a substantially lower performance when confronted with a domain shift, while the model trained on new synthetic data shows similar results in both domains (cf. Figure 1a).





(a) Trained on Synthetic Data

(b) Trained on nuScenes

Fig. 3: Both approaches are tested with data from a *Velodyne VLP32C* lidar sensor on one of our test vehicles. The model predicting the left OGM was trained using new synthetic data while the right OGM is predicted by a model trained with annotations from the nuScenes dataset.

VI. CONCLUSION

We presented two data-driven methods for occupancy grid mapping from lidar measurements. One approach uses synthetic training data while the other approach uses labels generated from manual annotations of the nuScenes [1] dataset. An evaluation on real-world data shows that both approaches are able to predict evidential OGMs that distinguish between static and potentially dynamic obstacles and take observability into account. A quantitative analysis shows that the model trained on synthetic data achieves lower performance on separating static from dynamic obstacles while free space is estimated with high precision with both approaches. However, the model trained on the annotated dataset performs significantly worse when presented with lidar measurements from another sensor model on another vehicle while the approach using synthetic data can be easily transferred. By an evaluation on annotated real-world data it becomes possible to identify the blind spots of simulation, e.g. falsely classified static and dynamic obstacles or road layouts. This information can be used to improve the simulation and further close the reality gap so that synthetic training data can be generated for arbitrary sensor configurations.

REFERENCES

- H. Caesar, V. Bankiti et al., "nuscenes: A multimodal dataset for autonomous driving," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11618–11628.
- [2] M. Buchholz, F. Gies et al., "Automation of the unicaragil vehicles," in 29th Aachen Colloquium, 2020, pp. 1531–1560.
- [3] T. Woopen, B. Lampe et al., "Unicaragil disruptive modular architectures for agile, automated vehicle concepts," in 27th Aachen Colloquium, 2018, pp. 663–694.
- [4] F. Bieder, S. Wirges et al., "Exploiting multi-layer grid maps for surround-view semantic segmentation of sparse lidar data," in 2020 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2020.
- [5] S. Richter, J. Beck et al., "Semantic evidential grid mapping based on stereo vision," in 2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI). IEEE, 2020.
- [6] R. van Kempen, B. Lampe et al., "A simulation-based end-to-end learning framework for evidential occupancy grid mapping," in 2021 IEEE Intelligent Vehicles Symposium (IV), 2021, pp. 934–939.
- [7] B. Lampe, R. van Kempen et al., "Reducing uncertainty by fusing dynamic occupancy grid maps in a cloud-based collective environment model," in 2020 IEEE Intelligent Vehicles Symposium (IV), 2020, pp. 837–843.
- [8] S. Thrun, W. Burgard, and D. Fox, Probabilistic robotics, ser. Intelligent robotics and autonomous agents. Cambridge, Mass.: MIT Press, 2005.
- [9] D. Nuss, S. Reuter et al., "A random finite set approach for dynamic occupancy grid maps with real-time application," The International Journal of Robotics Research, vol. 37, no. 8, pp. 841–866, 2018.
- [10] D. Bauer, L. Kuhnert, and L. Eckstein, "Deep, spatially coherent inverse sensor models with uncertainty incorporation using the evidential framework," in 2019 IEEE Intelligent Vehicles Symposium (IV), 2019, pp. 2490–2495.
- [11] K.-H. Lee, M. Kliemann *et al.*, "Pillarflow: End-to-end birds-eye-view flow estimation for autonomous driving," 03.08.2020.
- [12] J. Fei, K. Peng et al., "Pillarsegnet: Pillar-based semantic grid map estimation using sparse lidar data," in 2021 IEEE Intelligent Vehicles Symposium (IV), 2021, pp. 838–844.
- [13] M. Schreiber, V. Belagiannis et al., "Dynamic occupancy grid mapping with recurrent neural networks," in 2021 IEEE International Conference on Robotics and Automation (ICRA), 2021, pp. 6717–6724.
- [14] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.
- [15] S. Wirges, C. Stiller, and F. Hartenbach, "Evidential occupancy grid map augmentation using deep learning," in 2018 IEEE Intelligent Vehicles Symposium (IV 2018). IEEE, 2018, pp. 668–673.
- [16] J. Dequaire, P. Ondrúška et al., "Deep tracking in the wild: End-to-end tracking using recurrent neural networks," The International Journal of Robotics Research, vol. 37, no. 4-5, pp. 492–512, 2018.
- [17] A. Filatov, A. Rykov, and V. Murashkin, "Any motion detector: Learning class-agnostic scene dynamics from a sequence of lidar point clouds," in 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020, pp. 9498–9504.
- [18] P. Wu, S. Chen, and D. N. Metaxas, "Motionnet: Joint perception and motion prediction for autonomous driving based on bird's eye view maps," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11382–11392.
- [19] I. Shepel, V. Adeshkin et al., "Occupancy grid generation with dynamic obstacle segmentation in stereo images," *IEEE Transactions on Intelli*gent Transportation Systems, pp. 1–11, 2021.
- [20] G. Shafer, A mathematical theory of evidence, ser. Limited paperback editions. Princeton, NJ: Princeton Univ. Press, 1976, vol. 42.
- [21] A. Jøsang, Subjective Logic. Cham: Springer International Publishing, 2016
- [22] M. Sensoy, L. Kaplan, and M. Kandemir, "Evidential deep learning to quantify classification uncertainty," in *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)*, 2018, pp. 3179–3189.
- [23] G. Shafer, "Dempster's rule of combination," *International Journal of Approximate Reasoning*, vol. 79, pp. 26–40, 2016.

- [24] L. Reiher, B. Lampe, and L. Eckstein, "A sim2real deep learning approach for the transformation of images from multiple vehicle-mounted cameras to a semantically segmented image in bird's eye view," in 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), 2020, pp. 1–7.
- [25] Zeng Yihan, Chunwei Wang, Yunbo Wang, Hang Xu, Chaoqiang Ye, Zhen Yang, Chao Ma, "Learning transferable features for point cloud detection via 3d contrastive co-training," in Advances in Neural Information Processing Systems, 2021, vol. 34.
- [26] VIRES Simulationstechnologie GmbH, "Virtual test drive," 2022. [Online]. Available: https://vires.mscsoftware.com/
- [27] S. Manivasagam, S. Wang et al., "Lidarsim: Realistic lidar simulation by leveraging the real world," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 11 164–11 173.
- [28] A. H. Lang, S. Vora et al., "Pointpillars: Fast encoders for object detection from point clouds," in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 12689–12697.