Deep Inverse Sensor Models as Priors for Evidential Occupancy Mapping

Tiefe inverse Sensormodelle als A-priori-Information in evidenzbasierten Belegungskarten

Von der Fakultät für Maschinenwesen der Rheinisch-Westfälischen Technischen Hochschule Aachen zur Erlangung des akademischen Grades eines Doktors der Ingenieurwissenschaften genehmigte Dissertation

vorgelegt von

Daniel Max Bauer

Berichter: Univ.-Prof. Dr.-Ing. Lutz Eckstein
Univ.-Prof. Dr. sc. techn. Bastian Leibe

Tag der mündlichen Prüfung: 14.02.2024

Abstract

In recent years, the automotive industry was heavily shaped by the desire to increasingly automate driving tasks. This trend is motivated by statistical findings which indicate that the majority of traffic fatalities are caused by human error [ORG18]. In order to automate tasks, the vehicle obtains motion and occupancy information provided by a diverse set of sensors like cameras, lidars, radars or ultrasonic sensors via a transformation called Inverse Sensor Model (ISM). Much research has been done on handcrafted ISMs to capture more and more sensor characteristics. However, even though the behavior of handcrafted models can be easily tested and, therefore, well understood, they exhibit severe limitations. Some of the biggest challenges are modeling all sensor effects in arbitrary environments for the different types of sensors, the overhead of re-calibrating the ISMs with occurring changes (e.g. repositioning or updated hardware) and modeling the spatial coherence of detections (e.g. "do two detections belong to the same object?"). An alternative to handcrafted ISMs is to learn the sensor characteristics from data. With the improvements of deep learning within the last two decades, end-to-end learned transformations produce competitive or even improve upon their handcrafted counterparts. Nevertheless, the inner workings of models learned from data in an end-to-end manner are currently not well understood. Therefore, these learned ISMs are mostly treated as black box models which are difficult to test for arbitrary environments.

The aim of this work is to thoroughly investigate the current state of learned ISMs and compare them against handcrafted ones for sensors commonly deployed in automated vehicles. Additionally, methods are investigated to combine the handcrafted with the learned ISMs predictions to obtain the best of the two worlds. Thus, the contributions are twofold. In the first part, baseline handcrafted ISMs are defined and an architecture search is performed to identify the best learnable model, given imposed restrictions. Also, methods are compared to make the learned models aware of uncertainties, different input encodings for each sensor modality are compared and, eventually, the performance of the leaned ISMs are compared against each other and the handcrafted ISMs. In the second part of this work, the temporal accumulation of ISMs is investigated. Here, problems are identified and solutions are proposed when accumulating predictions from learned ISMs. Additionally, methods are investigated to combine the predictions of the two types of models. Here, under the assumption that handcrafted ISMs are better tested then the learned ones, a combination is proposed to restrict the influence of learned ISMs to solely initialize the environment and rely on the handcrafted ISM for convergence. To this extent, new combination rules are proposed and the results are compared against baseline fusion. All experiments are conducted on the NuScenes dataset [CAE20] which provides real-world urban data from multiple cities.

Kurzfassung

Ein Trend der letzten Jahre in der Automobilindustrie besteht in der zunehmenden Automatisierung von Fahrtätigkeiten. Dies ist auf Statistiken zurückzuführen, die einen Großteil der Verkehrstode mit menschlichem Fehlverhalten begründet [ORG18]. Um diese Automatisierung zu erreichen, muss dem Fahrzeug Information über die statischen und bewegten Verkehrsteilnehmer bereitgestellt werden. Dies geschieht durch so genannte inverse Sensormodelle (ISMs), die Messungen von Sensoren wie z.B. Kameras, Radaren, Lidaren und Ultraschallsensoren transformieren. Viel Forschung wurde getätigt, um zunehmend mehr Sensorcharakteristiken in händisch definierten ISMs abzubilden. Diese händischen ISMs sind aufgrund ihrer Transparenz leicht testbar und somit gut verstanden. Jedoch liegen ihre Schwächen z.B. in der Modellierung aller möglicher Sensoreffekte in generischen Umgebungen, dem Mehraufwand der Rekalibrierung der ISMs durch Änderungen (z.B. neue Sensorik oder Repositionierung) oder der Modellierung räumlicher Kohärenz von Messungen (z.B. Zuordnung von Detektionen zu Objektinstanzen). Eine Alternative hierzu stellen gelernte ISMs dar. Diese Modelle haben binnen der letzten zwei Jahrzehnte so stark von den Fortschritten im Bereich Deep Learning profitiert, dass sie die händischen ISMs eingeholt und, in manchen Fällen, übertroffen haben. Nichtsdestotrotz ist die innere Funktionsweise dieser gelernten Modelle bisher nicht vollständig verstanden. Daher werden sie weitestgehend als Blackbox-Modelle gehandhabt, deren Validierung und Verifikation in generischen Umgebungen noch weiterer Forschung bedarf. Ziel dieser Arbeit ist eine Gegenüberstellung des momentanen Stands gelernter und händisch definierter ISMs für Sensoren im Automobilbereich. Hinzukommt die Erforschung der Kombination der Prädiktionen beider ISM Typen mit dem Ziel das Beste aus den Welten zu vereinen. Daher sind die Beiträge der Arbeit zweigeteilt. Im ersten Teil werden Basismodelle für händische ISMs definiert und die beste Architektur für die gelernten ISMs unter Berücksichtigung der gegebenen Randbedingungen ermittelt. Weiterhin werden Methoden definiert und gegenübergestellt, um die Messungenauigkeiten in die Prädiktionen der gelernten ISMs einfließen zu lassen. Außerdem werden diverse Darstellungen der Messungen für die betrachteten Sensoren gegenübergestellt. Schließlich werden ISM-Varianten für jeden Sensor gelernt und mit den händischen ISMs verglichen. Im zweiten Teil wird die zeitliche Akkumulation von ISM-Prädiktionen betrachtet. In dem Zusammenhang werden auftretende Probleme für gelernte ISMs identifiziert und Lösungsansätze evaluiert. Des weiteren wird die zeitliche Fusion gelernter und händische ISMs untersucht. Hierzu wird, unter der Annahme, dass händische ISMs besser getestet werden können, eine Fusion vorgeschlagen, die die Rolle gelernter ISMs auf die Initialisierung der Umgebung beschränkt. Dies ermöglicht es den händischen ISMs das Konvergenzverhalten zu definieren. Alle Experimente werden auf Basis des NuScenes-Datensatzes [CAE20] durchgeführt, welcher reale Messungen aus urbanen Umgebungen mehrerer Städte beinhaltet.

Contents

1 Introduction

With the DARPA "Grand Challenges" [THR06] in 2004 and 2005 and the following DARPA "Urban Challenge" [URM07] in 2007, the feasibility for automated driving up to level 4 as defined by the Society of Automotive Engineers (SAE) [NN17] has been shown in controlled environments. This opened up the discussion how to revolutionize mobility.

The main challenge that can be addressed with automated vehicles is the reduction of the approximately 1.3 million people that die each year worldwide in traffic accidents according to WHO [ORG18]. Automated vehicles are less prone to human errors like distraction, fatigue or speeding. Additionally, the automated vehicle can provide faster reaction times with the increase of technology and a full 360° coverage of the surroundings given the correct sensor setup. Furthermore, automated vehicles offer the potential to reduce inner city parking demand [ZHA20a] and may decrease the cost of ride hailing, enabling mobility e.g. for elder people and potentially reducing emissions by making personal cars obsolete [SEV21].

In order to realize automated driving, companies either chose the top-down or bottom-up approach. The top-down approach consists in over-equipping the vehicle with a multitude of sensors and compute power followed by a phase of iterative reduction and refinement to directly target SAE level 4-5 automation. Prominent examples of which are e.g. Waymo [SUN20] and ArgoAI [CHA19]. On the other hand, most "traditional" automakers like Ford, VW or BMW, often focused on iteratively improving their already existing automated functions (SAE level 1-2) to higher levels of automation while at the same time being open for collaborations with the top-down approaching companies [FOR20].

In order to automate more sophisticated driving tasks (e.g. parking or lane changes), the vehicle has to be capable to perform the basic steps of robotic systems, namely sense, plan and act [ARK98]. Based on the degree of planning, one can distinguish the reactive paradigm which uses fixed mappings between sense and act, hierarchical paradigm with online planning and hybrid paradigm with steps-wise online planning. An illustration of the hierarchical paradigm is depicted in Fig. 1-1. However, for all of those paradigms the sensing block plays a crucial role as it provides the baseline for all of the upstream tasks. This sensing block can be further divided into sensory processing followed by world modeling (see [ARK98]).

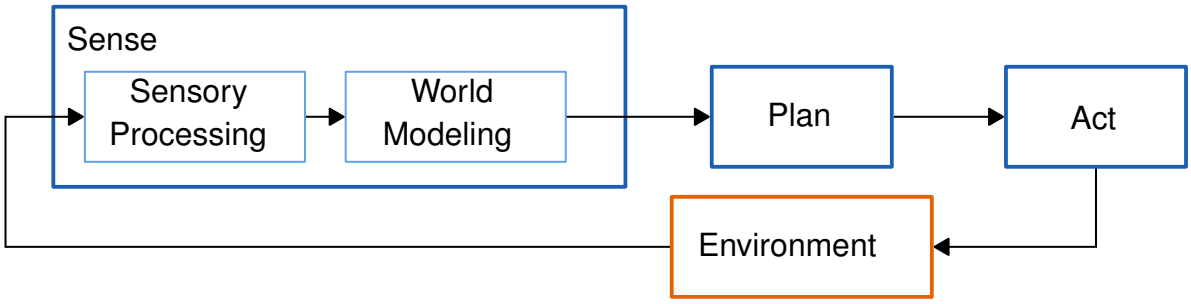


Fig. 1-1: Block diagram of a hierarchical robotic system.

The focus of this work lies on improving the world modeling capability of the sensing block for current and near future passenger vehicles. In order to perceive all obstacle locations in the full 360° surrounding of the ego vehicle, sensors like lidars, radars, cameras and ultrasonic sensors can be deployed (see e.g. NuScenes sensor setup [CAE20]). The obstacle locations, however, are only readily provided by spinning 360° lidars, which, nowadays, are too expensive to be deployed in passenger vehicles. An example of such a lidar is the Velodyne VLP16 (Puck) ranging currently around 4,000 $ according to [CNE18] and [VEL18b]. Therefore, these lidars are currently only deployed by companies focusing on automated driving (SAE level 4-5) or to provide ground truth information to verify automated driving features up to SAE level 3. For the remaining sensors, additional steps have to be performed like e.g. temporal accumulation to densify the measurements (mostly for radar and ultrasonic sensors) or domain adaption as applied to transform camera measurements into Bird's-Eye-View (BEV). It shall further be mentioned that, while being small, robust and cheap, ultrasonic sensors only provide information for narrow parking maneuvers since their range is restriction to about 6 m (e.g. [BOS22]). Therefore, they are still found in nowadays passenger vehicles but will not be included in this work.

Besides manually defining the transformation from sensor measurements to object locations, Neural Networks (NN) models are recently utilized to learn the transformation from data. This alleviates e.g. the problem of iteratively adapting the handcrafted model to every possible edge case while at the same time ensuring that the newly implemented changes do not interfere with the former performance of the model. Instead, a once trained NN model can be steadily improved by collecting edge cases, including them into the database and retraining the model. While this sounds tempting, there are also downsides to the application of NNs. Some of these downside include the inability of NNs to identify situations for which they have not been trained ("unknown unknowns"). In such situations, the predictions might not be based on semantically meaningful features and, as such, are sensitive to distributional shifts [APT19].

To mitigate these problems of NNs and verify them for production, the article "Safety

First for Automated Driving" [APT19] proposes the following. The NN can be extended to also predict its uncertainty. In this context, the aleatoric or data inherent and the epistemic or model related uncertainty have to be distinguished. The combination of these uncertainties can help downstream function modules to decide to which extent the predictions can be trusted. Additionally, the epistemic uncertainty can be used as an indicator whether to collect a data point for later retraining or not. Another way to verify NN predictions is the utilization of a so called observer. The observer checks the predictions of the NN e.g. given a set of rules like "Does the predicted trajectory coincide with any obstacle locations?". Eventually, a module can be run in parallel to the NN and predict the same target to provide redundancy. This can e.g. be realized using a slightly differently trained NN or one of the manually defined white box models.

In this work, the State-of-the-Art (SotA) of learned and handcrafted inverse sensor models is summarized in Chapter 2 and analyzed with regards to exemplary requirements found for nowadays passenger vehicle applications in Chapter 3. Based on these findings, Chapter 3 continues to detail extensions to potentially improve the robustness of the best suited SotA NN candidate following the guideline of [APT19]. More specifically, both SotA as well as newly proposed uncertainty estimation techniques are defined to modify the NN's predictions. Furthermore, a novel temporal fusion approach is defined in Chapter 3 to tackle the challenges that occur when temporally accumulating the learned environment models used in this work. At the end of Chapter 3, a novel alternative temporal fusion is defined in which the NN estimates have to be first verified by the handcrafted model. In this scenario, the handcrafted model acts as an observer for the NN predictions which should potentially further strengthen the robustness aspect according to [APT19].

To analyze the benefits and downsides, both learned and handcrafted benchmark models are tuned for the investigated task in Chapter 4. Afterwards, the benchmark models are compared against each other for each sensor modality and against the proposed uncertainty extensions. Similarly, Chapter 5 contains the experiments to compare the SotA temporal fusion with the proposed variants to highlight the need for the proposed novel fusion approach as well as show its limitations. Eventually, Chapter 6 concludes the work with a discussion of the obtained results with respect to the imposed requirements and an outlook for further work.

It is expected that these improvements have the most benefit in the passenger vehicle domain under the assumption that high accuracy, dense 360° lidar sensors cannot be deployed in this domain at least within the next decade. These findings are aimed to improve the perception block of the robotic pipeline (see Fig. 1-1) and, as such, will improve the performance of all downstream functions increasing the overall robustness of automated driving features.

2 State of the Art

This chapter briefly describes the main forms of environment representation used in this work, followed by an overview of modeled and learned sensor transformations to obtain the environment state from measurements detailed for each sensor investigated in this work. Eventually, the chapter is concluded with a summary of fusion methods to accumulate the the environment state over time.

2.1 Occupancy Mapping

Occupancy grid maps are an often employed form of environmental representation in the field of robotics [CAR15, ELF89, FLE07]. In this form, first introduced in [ELF89], the surrounding is stored as a BEV grid map where each cell contains information about the corresponding environmental patch's occupancy state. Here, a cell is defined to be occupied, if the robot is not able to traverse the area and free vice versa. In the original form, this state is expressed in the probabilistic domain using the probabilities for occupied p_o and free p_f for which the following holds

$$p_o \in [0, 1] \qquad \text{Eq. 2-1}$$

$$p_f = 1 - p_o \qquad \text{Eq. 2-2}$$

$$\boldsymbol{p} = \begin{bmatrix} p_f, & p_o \end{bmatrix} \qquad \text{Eq. 2-3}$$

Here, p_o equaling zero indicates a cell being free, p_o equaling one indicates a cell being occupied and p_o equaling 0.5 represents the unknown state. An alternative to the probabilistic formulation is proposed in [PAG96] which is widely applied for evidential mapping [MOR11, YU15, MOU17]. It uses the evidential representation [DEM68, SHA76] to define the occupancy state as the so called power set

$$2^U = \{\emptyset, F, O, U\}, \qquad U = \{F, O\} \qquad \text{Eq. 2-4}$$

consisting of the empty $\emptyset$, free F, occupied O and unknown set U. Additionally, mass functions $m(A)$ with $A \in 2^U$ are defined which map each element of 2^U to the amount of evidence associated with it. In their normalized form, which will be assumed from here on, the mass functions are defined to suffice the following conditions

$$m(\emptyset) = 0 \qquad \text{Eq. 2-5}$$

$$\sum_{A \in 2^U} m(A) = 1 \qquad \text{Eq. 2-6}$$

In the normalized form, $m(\emptyset)$ is always zero and, thus, can be removed from consideration. The vector of normalized mass functions can then be written as follows

$$\boldsymbol{m} = \begin{bmatrix} m(F), & m(O), & m(U) \end{bmatrix}^\top = \begin{bmatrix} m_f, & m_o, & m_u \end{bmatrix}^\top \quad \text{Eq. 2-7}$$

The evidential formulation adds an additional degree of freedom by modeling the unknown class separately. This makes it possible to distinguish the case of conflicting information, indicating a cell to be free and occupied at the same time, from the case of absent information which, in the probabilistic view, both results in p_o equaling 0.5. In [MOR11, YU15, KUR12], it is proposed to utilize the conflicting information to represent dynamic objects which comes naturally since they are neither free nor occupied but rather in a transition state.

To build such occupancy maps, sensor measurements are being fed into so called ISMs to obtain an estimate of the occupancy state around the vehicle. This estimate is then transformed into map coordinates and fused into the map using evidential combination rules. To obtain unbiased maps in which each information is weighted equally, the ISM estimates have to be informational independent [PAG96]. The following sections will elaborate on the variants of ISMs and evidential combination rules at hand.

2.2 Geometric Inverse Sensor Models

In contrast to sensor models, which describe the sensor characteristics given the environment, ISMs describe the environment given the sensor measurements. These models can be divided into two categories. On the one hand, the physical measurement principles of the sensor can be used to define a geometrical relationship between the measurement and the environment. These models will be referred to as geo ISMs. On the other hand, data-driven methods can be applied to learn the ISM from large bodies of data. Even though it is possible to use other data-driven methods to learn ISMs, the literature mainly focuses on the application of deep artificial neural networks. These models, which will be referred to as deep ISMs, are specifically suited for the task at hand, since they excel over other data-driven methods when it comes to utilizing large amounts of data [ZHO14] and are universal function approximators [HOR91].

2.2.1 Ray-Casting in geo ISMs

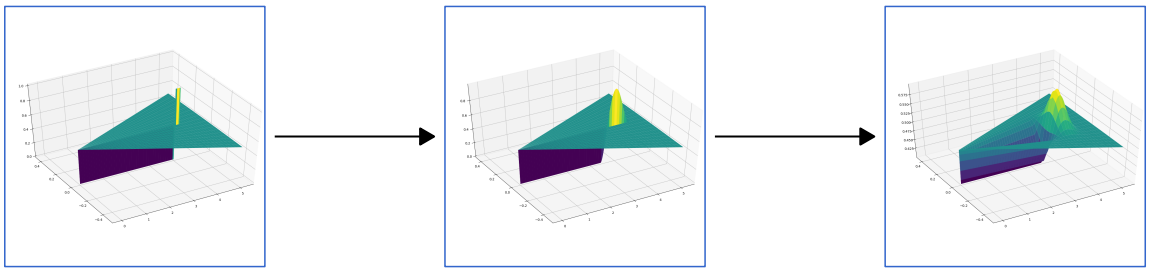
Most of the sensors deployed in automated driving for environment perception use a radial sensing principle like e.g. cameras and lidars. These sensors are only capable of perceiving objects in direct line of sight. To describe these sensor models, the following notation shall be introduced. For the geo ISMs, a Gaussian measurement noise is assumed whose mean and variance are defined in polar coordinates $[\boldsymbol{e}_r, \boldsymbol{e}_\varphi]$ as (μ_r, μ_φ)

and $(\sigma_r, \sigma_\varphi)$, respectively. For a given detection $(r_{\text{det}}, \varphi_{\text{det}})$, the mean is defined as the measured value while the variance is in most works obtained through offline calibration.

To model a radial sensor, Elfes [ELF89] proposed in his seminal work to utilize the Bayesian framework as follows. First, rays are cast from the sensor towards detections, modeling the regions crossed by the ray as an Inverse Detection Model (IDM). As a first step, the detection this IDM is assumed to be an ideal IDM ($\text{IDM}_{\text{ideal}}$), which defines everything along the distance r between the object boundary μ_r and the sensor as free, the position of the object boundary as occupied and everything else as unknown. Given the object angle μ_φ, this can be written as follows

$$\text{IDM}_{\text{ideal}}(r, \mu_r) = P(p_o(r) = 1 | \mu_r) = \begin{cases} 0 & , \varphi = \mu_\varphi \text{ and } r < \mu_r \\ 1 & , \varphi = \mu_\varphi \text{ and } r = \mu_r \\ 0.5 & , \varphi = \mu_\varphi \text{ and } \mu_r < r \\ 0.5 & , \text{else} \end{cases} \quad \text{Eq. 2-8}$$

Next, since the radial measurement is in fact not ideal but rather contains uncertainty, the $\text{IDM}_{\text{ideal}}$ is convolved with a radial and angular Gaussian noise model. Elfes argues that this model can only be evaluated in closed form for special sensor models and, thus, instead provides visualizations of the numerical results. These steps to obtain a 2D probabilistic IDM are visualized in Fig. 2-1.



a) b) c)

Fig. 2-1: Visualization of a) the $\text{IDM}_{\text{ideal}}$ with the sensor at position $(0, 0)$ and the detection at $(4, 0)$, b) the influence of radial Gaussian noise and c) radial and angular Gaussian noise on the $\text{IDM}_{\text{ideal}}$.

In [LOO16], Loop et al. provide a formulation of the IDM's radial component along μ_φ using the error function $\text{Erf}(r) = \frac{2}{\pi}\int_0^r e^{-r'^2}dr'$ which will be referred to as ideal IDM convolved with Gaussian noise model ($\text{IDM}_{\text{Gauss}}$). This IDM can be written as follows

$$\text{IDM}_{\text{Gauss}}(r, \mu_r, \sigma_r, \varphi = \mu_\varphi) = P(p_o(r, \varphi = \mu_\varphi) = 1 | \mu_r) = \frac{1}{2}\text{Erf}\left(\frac{\mu_r - r}{\sqrt{2}\sigma_r}\right) - \frac{1}{4}\text{Erf}\left(\frac{\mu_r - r - 3\sigma_r}{\sqrt{2}\sigma_r}\right) + \frac{1}{4} \quad \text{Eq. 2-9}$$

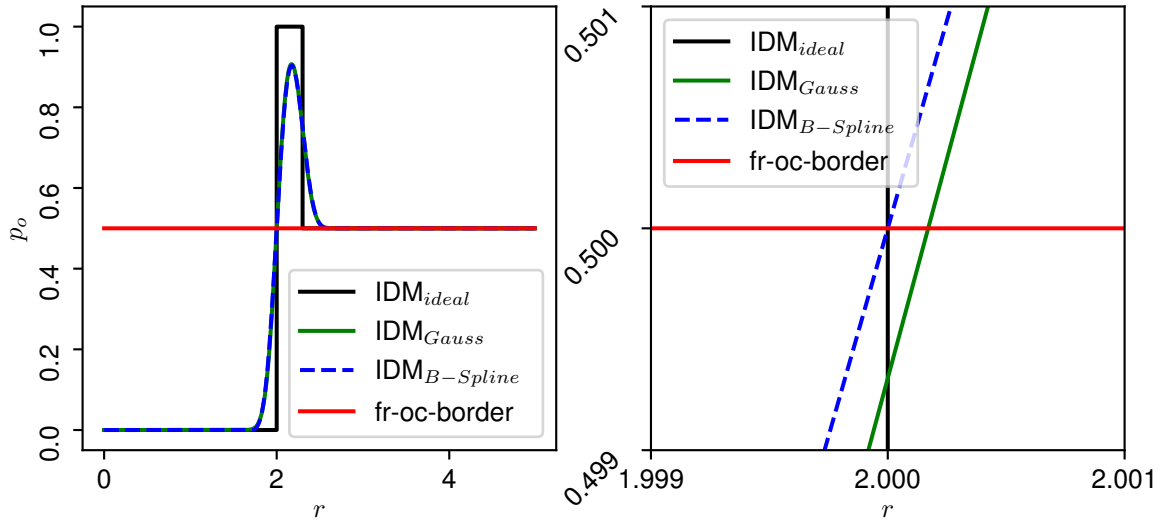


Fig. 2-2: Example of the $\text{IDM}_{\text{ideal}}$ with $\mu_r = 2$ and an object thickness $\tau = 0.3$, together with the $\text{IDM}_{\text{Gauss}}$ with $\sigma_r = 0.1$ and $\text{IDM}_{\text{B-Spline}}$ with the same variance. The left side shows a zoom of the right side around the border between free and occupied space.

They continue to show that the application of a Gaussian noise model always leads to a slightly shifted decision boundary between free and occupied space with regards to the $\text{IDM}_{\text{ideal}}$'s. This effect is illustrated in the zoomed region left in Fig. 2-2. Therefore, they propose a quadratic b-spline as an alternative noise model designed in a way to always deliver the same free-occupied-border as the $\text{IDM}_{\text{ideal}}$ while approximately maintaining the shape of the $\text{IDM}_{\text{Gauss}}$ (see Fig. 2-2). The quadratic b-spline and the corresponding radial IDM, after convolving the proposed b-spline with the $\text{IDM}_{\text{ideal}}$, are

referred to as ideal IDM convolved with B-Spline noise model ($\text{IDM}_{\text{B-Spline}}$). This IDM can be written as follows

$$\text{BSpline}(r) = \begin{cases} 0 & ,r < -3 \\ (3+r)^3/48 & ,-3 \leq r \leq -1 \\ \frac{1}{2} + r(3+r)(3-r)/24 & ,-1 < r < 1 \\ 1-(3-r)^3/48 & ,1 \leq r \leq 3 \\ 1 & ,3 < r \end{cases} \quad \text{Eq. 2-10}$$

$$\text{IDM}_{\text{B-Spline}}(r, \mu_r, \sigma_r, \varphi = \mu_\varphi) = \text{BSpline}((r-\mu_r)/\sigma_r) - \frac{1}{2}\text{BSpline}((r-\mu_r)/\sigma_r - 3) \quad \text{Eq. 2-11}$$

The $\text{IDM}_{\text{B-Spline}}$ has been widely adopted in occupancy mapping [MOU17, REI13, YU15].

In [PAG96], Pagac et al. extended the probabilistic IDM to the evidential framework given an arbitrary probabilistic IDM (IDM_{prob}). This evidential IDM (IDM_{ev}) can be written as follows

$$\text{IDM}_{\text{ev}} = \begin{cases} \begin{bmatrix} 2\Delta p, & 0, & 1-2\Delta p \end{bmatrix}^\top, & \text{IDM}_{\text{prob}} < 0.5 \\ \begin{bmatrix} 0, & 2\Delta p, & 1-2\Delta p \end{bmatrix}^\top, & \text{IDM}_{\text{prob}} \geq 0.5 \end{cases} \quad \text{Eq. 2-12}$$

$$\Delta p = \|0.5 - \text{IDM}_{\text{prob}}\| \quad \text{Eq. 2-13}$$

Eventually, to arrive at the ISM, the IDM is applied on each detection of the sensor measurement and their influences are accumulated using one of the fusion approaches described in 2.4.1.

2.2.2 Geo ISMs for Lidars

In this thesis the focus lies on 360° spinning lidars with 16 and more rays normally used for validating automated driving functions. The measurements of these lidars almost always contain both detections belonging to obstacles and drivable ground points. Thus, when applying the classical ray-casting approach, the IDM would mark ground points as occupied space. Also, since lidar detections are highly precise, the IDM rays are normally cast in a way that they are cutoff when intersecting with any detection. This is done in order not to put free space behind object boundaries. Yet, in case of ground plane points, this leads to ignoring all detections of successive lidar rays. Consequently, a common first step before projecting the detections into BEV and applying the IDM consists in ground plane detection, to be able to adapt the applied IDMs.

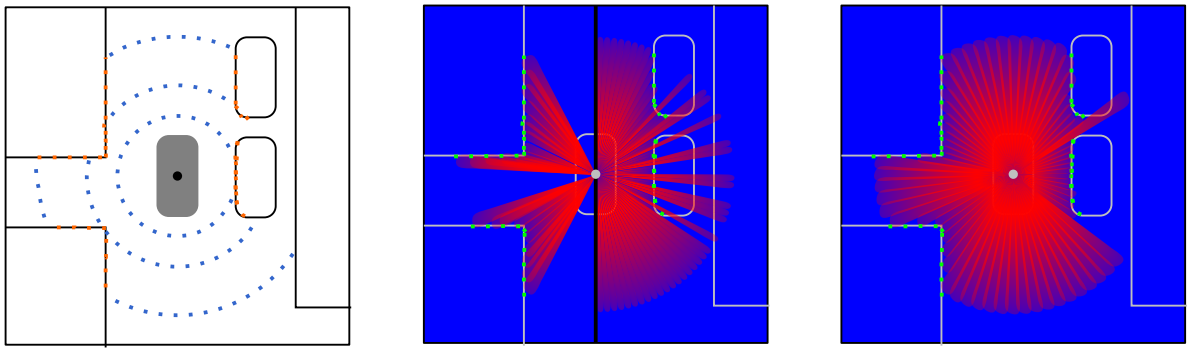
The most simple approach to detect ground points, under the assumption of a flat

surface, is the application of a height threshold as e.g. proposed in [THR06]. This, however, often leads to artifacts since the ground in most environments has a non-zero curvature. Solutions for this problem, as described in [NAR18], range from fitting either a single plane or piece-wise planar model e.g. using RANSAC or the Hough transform [FIS81, HOU62, OLI16, TIA20] and classifying all points within a given distance as ground, performing the classification based on thresholds of local features like average height, average variance, deviation in region normal vectors etc. [LI14, ASV15], through to the application of Conditional Random Fields [RUM17], Markov Random Fields [GUO11], deep learning [ZHA14] or manual labeling [VEL18a]. Finally, since the lidar shall be used in this work to provide ground-truth for object boundaries, the detection of the ground plane has to be adapted to the perceptive capabilities of the used radars. Here, to the best of the author's knowledge, only height threshold-based ground plane removal has been proposed [WES19, SLE19].

After distinguishing the ground plane detections, several possibilities to adapt the ISM arise. The first possibility is to adapt the IDMs for both the ground and object detections. Here, the ground point IDM must only apply the free space part of the model. For the object IDM, the ground detections shall be ignored so that they allow the casted rays to pass through and not to cause any collisions. The problem with this approach is that a ray needs to be cast for every detection which, for spinning lidars with 32 beams and more, can only be handled by discretizing the detections into a BEV image (e.g. Velodyne's HDL-32E provides up to 1.39 million points/second [21]). Therefore, this method is rarely adopted in practice.

Another alternative is to remove the ground points and only apply the IDM for object detections. However, by removing the ground plane detections, the information about free space gets lost for areas not affected by the object detection's rays, as illustrated in the left sector in Fig. 2-3 b). To solve this problem under the assumption of dense detections, IDM rays are cast for each angle up to a maximum distance within the lidars Field of View (FoV) instead of only for angles with detections. In doing so, the IDM can still be applied for all detections hit by rays and all the rays ending at maximum distance are altered to only model the free space. This variant is favored by the majority of works [THR06, NAR18, FIS81, HOU62, OLI16, TIA20].

In case the density assumption is violated, this approach, however, might lead to casting free space rays in between object detections assigning actually occupied space as partially free (see Fig. 2-3 b) right sector). To counteract this effect, it is possible to adapt the BEV discretization according to the point clouds density to close the gaps between detections or to increase the opening angle of the IDM rays to adapt for increased point cloud sparsity with increasing distance. The effects of counteracting the density violation are illustrated in Fig. 2-3 c).



a) ground-plane detection & removal

b) ray casting problems

c) better tuned ray casting

Fig. 2-3: Illustration of the Inverse Lidar Model (ILM) stages based on a scene with the ego vehicle and its mounted lidar sensor in the center, two building on its left and two parked vehicles in front of a building on its right side. The phases consist of detecting and removing the ground-plane as shown in a) with ground detections in blue and object detections in orange, followed by the application of the IDM. Here, three different variants are illustrated. The left sector in b) shows the result of only applying the IDM for angles with object detections with resulting sparse free space. On the right side in b), the application of an IDM with a small opening for all 360° angles using small angular increments is shown, illustrating the effect of rays between detections cutting through objects. Finally, in c), the IDM applied for all angles but this time with a larger opening angle and bigger angular increments is shown, which largely eliminates the problematic effects illustrated in b).

2.2.3 Geo ISMs for Cameras

In the case of cameras, measurements are usually provided as a 2D projection not equal to BEV. Thus, in order to compute the BEV projection, the first step consists in obtaining the depth information. Mallot et al. [MAL91] propose to utilize a homographic transformation, they refer to as inverse perspective mapping, which projects the whole environment including all objects on a flat ground surface in BEV. This geometric transformation can either be obtained via triangulation, in case of given intrinsic and extrinsic camera parameters, or by providing at least four point correspondences between the camera and the BEV image. These correspondences can e.g. be provided by projecting radar or lidar both into BEV and into the camera's pixel coordinates. This transformation, however, causes several problems. First, it depends on the camera calibration and, thus, needs to be recalibrated to account for changes. Additionally, all non-driveable objects violate the transformation's assumption of flatness and, therefore, appear as stretched out shapes on the ground which have to be filtered in a way to retain their actual boundaries. Finally, since this pipeline does not recover a meaningful 3D point cloud of the environment, the methods to distinguish between ground and obstacle points as described for lidar are not applicable. To obtain this distinction,

an additional step is required which could include e.g. semantic segmentation or object detection.

Alternatively, the depth can be estimated using the temporal correlation of subsequent images which was originally proposed to solve problems like structure from motion [LON81] or visual SLAM [DAV07]. These methods can be divided into so-called feature-based and direct approaches, both of which have been heavily investigated in the past [MA20]. However, all depth estimation methods relying on temporal correlation assume a static environment and, thus, are, in their original formulation, incapable of estimating the depth for dynamic objects. A practical solution is currently still under investigation, as detailed in [SAP18].

Another way of obtaining depth information of monocular images is by solving the underconstrained problem of directly inferring the depth using deep learning. Eigen et al. [EIG14] were some of the first to tackle the problem by training a multi-scale network in a supervised way on interpolated lidar depth labels. In [LIU15, LI15], this approach is extended through an additional post-processing step with a CRF. Also, [LAI16] show improved performance when the inverse Huber loss (BerHu) is applied for training. Moreover, Cao et al. [CAO17] reformulated the depth regression as a classification problem by discretizing depth labels. This is further improved in [FU18] by using ordinal regression which is still one of the state-of-the-art supervised methods to date. However, most of the time, expensive lidar sensors are deployed to obtain sparse depth labels. As a means to obtain cheaper dense depth supervision, Garg et al. [GAR16] proposed a self-supervised learning approach using a stereo camera. They achieve this feat by reconstructing the image of one camera using the other camera's image through a transformation defined by the depth estimate and the given extrinsics between the stereo pair. The comparison of the so reconstructed image with the original image is then used as a supervision signal. This approach is extended in [GOD17] by estimating the disparities for both stereo cameras and additionally enforcing the disparities to be consistent. Zhou et al. [ZHO17] improve this approach by integrating findings from [UMM17] to arrive at at fully self-supervised monocular method. In their approach, the fixed correspondence between the stereo pair is being replaced by temporal context with a pose estimation network to obtain the relative motion. However, due to utilizing the temporal context as supervision signal, methods of this kind inherit the problems of structure from motion, as mentioned above (e.g. dynamic objects). To alleviate this problem, Yang et al. [YAN20] propose, similarly as in [FEN19], to account for heteroscedastic aleatoric uncertainties in the loss to dampen the influence of outliers. Alternatively, Godard et al. [GOD19] propose to alter the loss function by, instead of using the average loss between all temporal contexts, using the minimum reprojection loss to ignore occlusion outliers and an auto-masking loss to further remove outliers like moving objects, pixels at far distances or from low gradient

environments. To obtain the best of two worlds, Kuznietsov et al. [KUZ17] proposed a semi-supervised method which deploys sparse lidar depth labels in addition to a direct image alignment loss. Recently, Guizilini et al. [GUI20] proposed a semi-supervised improvement over [GOD19] which additionally adapts the network architecture in a more information preserving way.

2.2.4 Geo ISMs for Radars

For this work, only the radar signals prefiltered by the manufacturer are considered. These measurements are provided in the form of point clouds in 2D Cartesian coordinates with attributes ranging from relative velocity, false alarm probabilities, dynamic states (e.g. oncoming, crossing, etc.) up to track ids, depending on the features offered by the manufacturer [CAE20]. Since the detections are provided in Cartesian coordinates, they can easily be projected into BEV images and the IDM, as described in Section 2.2.1, can be applied. Such a radar-based ISM will be referred to as geo Inverse Radar Model (IRM).

However, the accuracy of radars is not constant, as assumed in the standard IDM, but rather depends on the radial distance, reflecting material, detection angle and atmospheric interference. Therefore, the IDM's probabilities have to be adapted to account for these influences. To do so, Clark et al. [CLA12] propose the following scaling factor

$$G_{\text{Clarke}} = G_p G_\varphi G_a \qquad \text{Eq. 2-14}$$

where G_p scales the IDM linearly dependent on the received signal's power amplitude relative to its range. Moreover, G_φ is used to decrease the detection probabilities quadratically depending on the angle between detection and sensor center ray relative to the maximum angle in the sensor's FoV. Eventually, G_a is used to incorporate the influence of range dependent grid areas. This, however, becomes only relevant for polar grids and can be neglected in Cartesian coordinates. This model is further extended in [PRO18] by additionally decreasing the probabilities linearly depending on the ego vehicle's velocity.

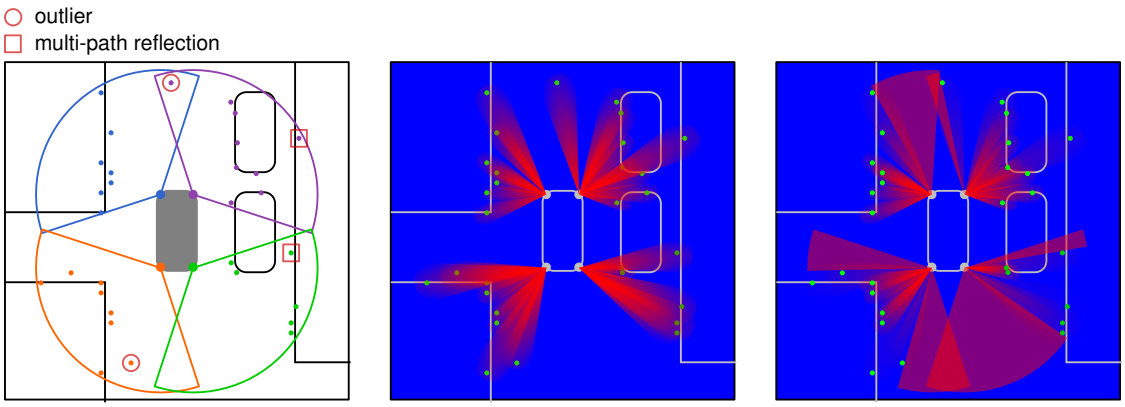
In addition to the different noise influences, most point clouds of contemporary automotive radar used in production are highly sparse (e.g. 64 detections per radar for the sensors used in this work). The sparsity in combination with the fact that the signals can contain outliers and detections behind objections due to multi-path reflections can lead to casting IDM rays which partially cut through actually occupied areas, as shown in Fig. 2-4 b). To counteract this effect, Weber et al. [WER15] propose to, additionally, reduce the influence of detections linearly dependent on their range. Moreover, they ignore free space of rays which falls together with occupied areas of other rays. Even

though this procedure not fully removes the problem of rays cutting through objects, it highly dampens it and prohibits foreground detections from being overwritten.

Another problem connected to the detection sparsity is the lack of assigned free space, which is also illustrated in Fig. 2-4 b). To enrich the free space, Prophet et al. [PRO18] propose the following procedure

1. identify the detection D_1 which is closest to the boundary of the FoV
2. define the cone spanned between D_1 and the FoV's border as free space
3. in case another radar provides detections falling inside the first's FoV, identify the detection D_2 closest to D_1 and restrict the free space cone's angle by D_2

which also leads to free space in regions without detections, as shown in Fig. 2-4 c).



a) detections of four corner radars

b) naive IRM

c) IRM with scaling factors and free space enrichment

Fig. 2-4: Illustration of a geo IRM for an ego vehicle (center) being equipped with four corner radars in a scene with two buildings on its left and two parked vehicles in front of a wall to its right. The detections, sensor centers and corresponding FoVs are shown in a) in different colors for each sensor. Additionally, two outliers (red circles) and two detections due to multi-path reflections (red squares) are shown. The direct application of an IDM on each detection is shown in b), illustrating the effects of sparse free space, the unfiltered application of the IDM for outliers and free space being assigned to actually occupied areas due to the rays for multi-path reflection detections. Eventually, c) the enrichment of free space and filtering of outlier and multi-path reflection effects as described in Section 2.2.4.

For completion it shall be mentioned that in case of denser radar point clouds, e.g. as provided by imaging radars, similar methods as for the lidar point cloud after ground-plane removal can be applied [SLU19].

2.3 Deep inverse Sensor Models

In recent years, literature primarily focuses on the creation of deep ISMs, which utilize deep Convolutional Neural Network (CNN)s to learn the characteristics of ISMs.

2.3.1 Deep ISM Architecture

Architecture-wise, the majority of works utilize UNets [RON15] consisting of an encoder and a subsequent decoder network with skip connections [PRO19, SLE19, WIR18, WES19, SCH18, LU19, MAN20]. The encoder network successively subsamples, commonly by a factor of two, the feature dimension either using a form of pooling (e.g. max or average pooling) to obtain shift invariance or strided convolutions as an alternative with a learned kernel. This results in bringing the spatial dimensions closer allowing the computation of increasingly global features while keeping the convolution kernel size small. In the standard setup, the padding is set to $(\kappa - 1)/2$ in order to pad the kernel overlap at the borders. Regarding the kernel size, only 3×3 convolutions are applied since it has been shown in [SIM14] that two 3×3 convolutions result in the same receptive field as one 5×5 convolution but need less parameters. As a downsampling mechanism, most papers deploy strided convolutions with stride size two as it is a learnable operation as opposed to the pooling alternatives. To define the depth of the network, the resulting receptive field size is essential and has to be chosen specifically for the task at hand. For the described standard operations with kernel size $\kappa = 3$ and stride $s \in [1, 2]$ and padding of one, the receptive field ($\mathrm{RecepField}$) can be computed as follows

$$\mathrm{RecepField}_{i+1} = \mathrm{RecepField}_i + (\kappa_i - 1) \prod_{j=0}^{i} s_j \qquad \text{Eq. 2-15}$$

The decoder is the inverse of the encoder using a bilinear upsampling [ODE16] to replace the subsampling layers. The skip connections either add or concatenate information of the encoder to corresponding decoding layers. This is done in order to regain the information lost in or during encoding. These skip connections can include additional convolutions which are mostly used to compress the amount of features before concatenation. While the feature dimension is successively halved in the encoder, the amount of features is commonly doubled after each subsampling. This introduces the initial amount of features as an architectural hyperparameter and, in some cases, a maximal number of features. The architecture search is being standardized in [RAD20] for a generalized UNet architecture with ResNet layers by successively evaluating the influence and ranges of each parameter. An example of such a standard UNet architecture is depicted in Fig. 3-7. In most cases, the convolution (conv) layers consist of 3×3 or 1×1 convolutions, in some cases with Dropout [SRI14] or stride s, followed by

batch normalization (BatchNorm) [IOF15] and a non-linearity like leaky Rectified Linear Unit (ReLU) [MAA13], which is depicted on the left-hand side in Fig. 2-5.

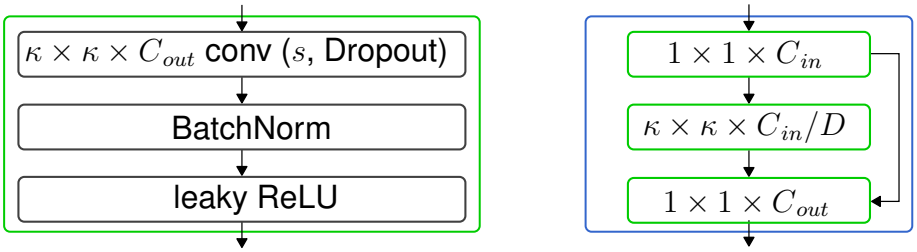


Fig. 2-5: Structure of a convolution (left) and ResNet layer (right) as commonly used in the literature with the kernel size κ, stride s, amount of channels C and channel reduction factor D. The green blocks in the ResNet layer are convolutions as depicted on the left-hand side.

Some of the variations to UNets include exchanging the encoder with backbone networks like VGG19 [SIM14, WUL18] or EfficientNet [TAN19, PHI20]. However, most commonly, the standard convolution layer is replaced by a ResNet layer [HE16, WIR18, REI20, ROD20, PHI20]. Here, the 3×3 convolution is replaced by a miniature UNet first compressing the feature channels with a 1×1 convolution, followed by the actual 3×3 convolution and, finally, decompressing it back to the original channel dimension, again, using a 1×1 convolution. Additionally, before applying the final output non-linearity, the input features are added to the outputs. The number of Multiply-Accumulate Operations (MAC), abbreviated with N_{MAC}, over the ResNet layer's three convolutions can be written as follows

$$N_{MAC} = 1 \cdot 1 \cdot WHC_{in}\frac{C_{in}}{D} + 3 \cdot 3 \cdot WH\frac{C_{in}^2}{D^2} + 1 \cdot 1 \cdot WH\frac{C_{in}}{D}C_{out} \qquad \text{Eq. 2-16}$$

for features of width W, height H, a channel reduction by D and C_{in}, C_{out} input and output channels respectively. For $C_{in} = C_{out}$, the ResNet layer needs less MACs compared to the $9 \cdot WHC_{in}^2$ MACs of a 3×3 convolution for $D > 1.12$. Different sources report only minor accuracy decrease for $D = 4$ or higher, leading to 11.8% of the MACs needed for the standard convolution [HE16, BAZ18].

While all authors agree upon the fact that re-weighting the loss to account for class imbalance is necessary, they diverge in the choice of the underlying loss function. In the majority of cases, the training is setup as a semantic segmentation problem, training the network on the cross entropy loss [PRO19, LOM17, HEN20, WUL18]. On a different note, Weston et al. [WES19] and Lu et al. [LU19] train a Variational Autoencoder (VAE) [KIN13] for which the log-likelihood is optimized in visible and the Kullback-Leibler divergence [KUL51] in occluded areas. This results in a Gaussian distributed log-odds space which equals the prior's distribution in unobserved areas.

Weston et al. further define the log-odds as priors to the final sigmoid activation and arrive at the final prediction by marginalizing over the log-odds. Another alternative is proposed by Sless et al. [SLE19] who apply the Lovasz loss [BER18] as a differentiable surrogate to the intersection-over-union. Here, the Lovasz loss comes with the benefit of accounting for class imbalances by design. Eventually, Wirges et al. [WIR18] define the training as a regression problem and, thus, apply the $\mathcal{L}_1$ and $\mathcal{L}_2$ losses for optimization. This enables them to utilize the continuous nature of the training targets to continuously dampen the influence of the loss in areas with high target uncertainties.

When it comes to choosing an optimizer, the vast majority of works use the ADAM optimizer [KIN14, VER19, WIR18, WES19, SCH18].

2.3.2 Deep ISMs for Cameras

In the vision domain, the environment representation is often elevated from the binary classification into free and occupied space towards estimating a more detailed semantic layout including classes like vehicles, streets, sidewalks, vegetation etc. Since there is a vast amount of different approaches to tackle the problems in this field of research, Fig. 2-6 provides a short overview of the solution space. The main problem in training a deep Inverse Camera Model (ICM) to estimate such a semantic layout is the transformation between camera and BEV projection. To solve this, Schulter et al. [SCH18] propose to preprocess the images by estimating the monocular depth and semantic information. These estimates are further used to create a semantically annotated BEV image which is then fed into a neural network for refinement. In [PHI20], an end-to-end architecture is proposed to combine the Monocular Depth Estimation (MonoDepth) estimation and the BEV refinement into a single network. This is achieved by first predicting a feature vector and a categorical depth distribution for each pixel, resulting in a 3D point cloud in the shape of a pyramid. Each 3D point is then assigned with the feature vector, scaled by the probability of the categorical depth distribution. Here, the learned feature vector has the potential to encompass more relevant information than the semantic labels provided in [SCH18]. Afterwards, the point cloud is projected into BEV and further refined by another, simultaneously trained network. An alternative path to handle the transformation to BEV is to directly feed the image into a Fully Convolutional Network (FCN) and learn the transformation implicitly, as proposed in [MAN20, LU19]. To obtain a more explicit integration of the transformation into the network, Reiher et al. [REI20] and Roddick and Cipolla [ROD20] proposed use a spatial transformer layer [JAD15] to transform the latent space according to the homography defined by the camera intrinsics and exterinsics as explained in Section 2.2.3. Similarly, Pan et al. [PAN20] define a custom layer called "View Transformer Module" which uses a Multilayer Perceptron (MLP) to learn the positional mapping between the compressed camera and the BEV features.

Another problem of training a deep ICM is that most image datasets like Kitti [GEI13] or CityScapes [COR16] do not come with semantic map data which adds the challenge to provide labels. To overcome this problem, Schulter et al. [SCH18] train their refinement network in a way that the outputs resemble simulated patches of street layouts. This feat can be achieved by utilizing an adversarial loss as known from the literature of Generative Adversarial Networks (GAN) [GOO20]. Since the goal is not only to refine the semantic MonoDepth point cloud projections to resemble general real street layouts but also to show the street layout at hand, an additional reconstruction loss is introduced. This prohibits the refinements to stray too far from the original street layout. Additionally, in case of GPS measurements, patches from OpenStreetMaps [HAK08] can be aligned with the ego vehicles pose to offer another training signal. Another way to obtain BEV labels is to use depth information from sensors like lidars, stereo cameras or through MonoDepth and transform semantic segmentation estimates into BEV [MAN20, LU19]. Since this results in sparse labels, one can apply an additional adversarial loss on OpenStreetMaps to enhance resemblance with real street layouts [MAN20]. A completely different approach is to utilize simulation environments to obtain high quality semantic BEV labels. In [REI20], networks are trained purely based on simulated BEV labels and semantic input images. They demonstrate that Semantic Segmentation (SemSeg) inputs can work to a certain extent as a proxy to bridge simulation and real world measurements. Similarly, Pan et al. [PAN20] train their network both with simulated and real world inputs. Here, the simulated inputs can be directly fed into the network and a reconstruction loss can be utilized. The real world inputs, however, are first transformed to semantic images which are then, via a style transfer network [JIN19], transformed to resemble simulated images. Since there are no labels for real world inputs, an adversarial loss between the generated outputs and the simulated labels is applied.

Finally, the question remains of how to handle occluded areas. In [SCH18], this problem is addressed in the pre-processing step by masking out objects not belonging to the street geometry like pedestrians and vehicles and only optimizing the loss in the unmasked areas. This leads the networks to interpolate the masked areas as if no occluding objects were present. Alternatively, Mani et al. [MAN20] propose to reduce occlusion of static semantics and increase density of the labels by projecting and integrating a certain amount of future labels to the current vehicle position using odometry measurements. A different approach is to acknowledge the fact of missing information in occluded areas and estimating them as unknown, thereby reducing the ill-posedness of the problem. This can for example be achieved by assigning those areas to a separate unknown class [REI20] or by masking the areas and assigning a specific loss which equally distributes probability into all classes [ROD20].

supervision signal	BEV projection			
	MonoDepth + SemSeg	FCN	spatial trafo. layer	learned trafo. layer
adversarial	[SCH18]	[MAN20]		
maps	[SCH18, PHI20]	[MAN20]	[ROD20]	
sparse supervision		[MAN20, LU19]		
simulation		[REI20]	[REI20]	[PAN20]

Fig. 2-6: Overview of deep ICM literature with row-wise supervision approaches and column-wise solutions for the transformation between camera and BEV projection

2.3.3 Deep ISMs for Range Sensors

The origins of approximating ISMs for range sensors using neural networks go back to the 90s [VAN95, THR93]. In the case of automotive radar and lidar sensors, the measurement signals can be directly projected into BEV images and fed into neural networks which makes the training data quite similar. One differentiating factor, however, is the measurement signal representation. In case of lidar, the simplest representation can be obtained by projecting the detections into BEV and feeding it into the neural network [LIA18]. To provide additional input information, Wirges et al. [WIR18] use the intensities, detection positions and transmissions measured by the lidar and mark them in separate channels each for ground and non-ground points, arriving at six input channels. Hendy et al. [HEN20] additionally compute the density of all lidar detections and the maximal height value for each grid cells as input features, resulting in eight channels. Eventually, Wulff et al. [WUL18] use the detection positions, cell density, height thresholded detections, six height statistics (min, max, mean, min-max difference, mean-standard-deviation, mean-variance) and the same six statistics for the reflectivity which sums up to a 15 channel input. Similar approaches can be observed when it comes to radar sensor. Here, it has been proposed to either use the detection positions directly [SLE19] or to accumulate detections over time in order to account for the sparsity [PRO19, LOM17]. Additionally, features like radar cross section, signal to noise ratio, ambiguous Doppler interval and relative x and y velocities can be encoded into separate channels [HEN20]. Weston et al. [WES19] use a radar setup which provides the raw, dense range returns without Doppler information. These measurements are projected into a polar BEV image and fed into the network. The only other approach that uses polar instead of Cartesian coordinates is proposed by Verdoja et al. [VER19]. However, instead of encoding the inputs and targets as images, they rather perform inference on a vector of range measurements where each dimension

corresponds to an angular bin. Given this input vector, a Laplacian depth distribution is predicted for each dimension which can be convolved with the ideal ray-casting model from Section 2.2.1 and afterwards utilized for occupancy mapping.

2.3.4 Fusion of Sensor Modalities in deep ISMs

Fusing range sensor modalities is trivial since they already start in the same representation space and can, thus, be fused at any point in the network either by concatenation, summation or a specific network layer like an attention layer [VAS17]. On the other hand, to fuse camera measurements into a BEV representation, many different approaches have been proposed. Wulff et al. [WUL18] propose an early fusion by first transforming the camera image into BEV via an affine transformation given by its camera parameters and concatenating it with the range measurements. Alternatively, Liang et al. [LIA18] propose a so called "continuous fusion layer" which fuses intermediate features of a camera encoder network with features of the lidar encoder network. This is realized by extracting the k nearest lidar points for each BEV pixel, projecting them to the camera image, taking the camera image's pixel values and feeding them into an MLP where they are weighted by their distances towards the BEV pixel. Finally, Hendy et al. [HEN20] propose a late fusion on the softmax values by applying either average or priority pooling.

2.4 Evidential Combination Rules

2.4.1 Combination of independent Evidence

There are mainly two combination rules used to combine independent evidences in the evidential occupancy mapping framework. One of those rules is Dempster's rule of combination [DEM68] which is applied in numerous works on occupancy maps [PAG96, YU15, MOR11, MOU17]. It can be used to fuse two independent sources of evidence $\boldsymbol{m}_1$ and $\boldsymbol{m}_2$ as shown for the occupancy specific case by the following equations

$$K = m_{f1}m_{o2} + m_{o1}m_{f2} \qquad \text{Eq. 2-17}$$

$$\boldsymbol{m}_1 \oplus_D \boldsymbol{m}_2 = \begin{bmatrix} (m_{f1}m_{f2} + m_{f1}m_{u2} + m_{u1}m_{f2})/(1-K) \\ (m_{o1}m_{o2} + m_{o1}m_{u2} + m_{u1}m_{o2})/(1-K) \\ m_{u1}m_{u2}/(1-K) \end{bmatrix} \qquad \text{Eq. 2-18}$$

This combination rule is defined in a way that removes the influence of the conflict K, requiring the need for the normalization of the fused mass by $1-K$. Additionally, the unknown mass after the combination m_{u12} is always less or equal to the biggest in-going unknown mass, which can be written as follows

$$\max(m_{u1}, m_{u2}) \geq m_{u12} \qquad \text{Eq. 2-19}$$

To show this property, it is sufficient to prove it for one of the in-going unknown masses, since Dempster's combination rule is associative. Thus, the property shall be shown under the assumption that m_{u1} is the bigger in-going unknown mass. It then follows that

$$m_{u1} \geq m_{u12} \underbrace{=}_{\text{with Eq. 2-18}} m_{u1}\frac{m_{u2}}{1-K} \qquad | \div m_{u1} \qquad \text{Eq. 2-20}$$

In case of $m_{u1} = 0$ both sides equal zero. For $m_{u1} \in (0, 1]$, the following holds

$$1 \geq \frac{m_{u2}}{1-K} \underbrace{=}_{\text{with Eq. 2-5}} \frac{1-m_{f2}-m_{o2}}{1-K} \qquad | \cdot (1-K) \qquad \text{Eq. 2-21}$$

$$1-K \underbrace{=}_{\text{with Eq. 2-17}} 1-m_{f1}m_{o2}-m_{o1}m_{f2} \geq 1-m_{f2}-m_{o2} \qquad | -1+m_{f1}m_{o2}+m_{o1}m_{f2} \qquad \text{Eq. 2-22}$$

$$0 \geq m_{f2}\underbrace{(m_{o1}-1)}_{\leq 0} + m_{o2}\underbrace{(m_{f1}-1)}_{\leq 0} \qquad \text{Eq. 2-23}$$

The alternative to Dempster's rule, which is equally often applied in evidential occupancy mapping [WIR18, KUR12, REI13], is Yager's rule of combination [YAG87] as shown for the occupancy specific case by the following equations

$$\boldsymbol{m}_1 \oplus_Y \boldsymbol{m}_2 = \begin{bmatrix} m_{f1}m_{f2} + m_{f1}m_{u2} + m_{u1}m_{f2} \\ m_{o1}m_{o2} + m_{o1}m_{u2} + m_{u1}m_{o2} \\ m_{u1}m_{u2} + K \end{bmatrix} \qquad \text{Eq. 2-24}$$

In contrast to Dempster's rule, Yager's rule redistributes the conflicting portion of the fused masses into the unknown class. In the case of big conflicts K, this makes it possible to recuperate unknown mass.

For the sake of completeness, it shall be mentioned that, for the general, multi-hypothesis case, there is a big body of literature describing the shortcomings of Dempster's and Yager's rule together with propositions of how to handle them [ZAD79, HAN08, YAN13, ZHA20b].

2.4.2 Combination of dependent Evidence

In case of dependencies between evidences, direct combination, as proposed is Section 2.4.1, leads to accounting twice for the dependent portion of information. The literature proposes two lines of solutions namely removing the redundancy of one of the evidence masses before combining them or adapting the combination rule to account for occurring redundancies. Here, the problem with newly proposed combination

rules is that some do not suffice all properties of evidential combination rules like associativity and normalization as pointed out in [CAT11]. This would pose the tedious task of first verifying the methods for correctness while at the same time Dempster's and Yager's rule from Section 2.4.1 already provide valid operations. Moreover, there is no preference or comparison in the literature between the two categories. Thus, the focus in this work lies on removing redundancies before combining evidences.

To discount evidential masses, Shafer [SHA76] has proposed a discount operation as follows

$$\gamma \otimes \boldsymbol{m} = [\gamma m_f, \gamma m_o, 1 - \gamma + \gamma m_u]^\top \qquad \text{Eq. 2-25}$$

This operation has been adopted in all of the following methods, while different approaches have been proposed to obtain the discount factor γ. One way to obtain the discount factor, as proposed in [JIA09] and further adopted in [GUR06], is to predefine redundancy categories like highly, weakly and non-dependent with corresponding redundancy weights $2/3$, $1/3$, 0. To provide an example, similar measurements from the same sensor over time is highly dependent while similar measurements from different sensor sources over time are weakly dependent. Alternatively, Su et al. [SU15] propose to obtain redundancy factors between sources of information like sensors, experts and models via statistical experiments. They propose to utilize the Pearson correlation coefficient [BEN09] as a statistical measure. This has been adapted by Shi et al. [SHI17] by using the Spearman rank correlation coefficient instead and, eventually, combined by Xu et al. [XU17] who multiply both Pearson and Spearman coefficients to form a hybrid approach. On a different note, Yager [YAG09] proposed to use the specificity S_p and entropy S_e to measure that information is mostly distributed into single element classes and to measure conflicting information respectively. These can be defined both for the general and occupancy mapping case as follows

$$S_p(\boldsymbol{m}) = \sum_{A \in 2^U} \frac{m(A)}{\mathrm{card}(A)} = \frac{m_f}{1} + \frac{m_o}{1} + \frac{m_u}{2} \in [0.5, 1] \qquad \text{Eq. 2-26}$$

$$S_e(\boldsymbol{m}) = \sum_{A \in 2^U} -\log_2(\mathrm{Pl}(A))m(A) = -\log_2(m_f + m_u)m_f - \log_2(m_o + m_u)m_o \in [0, 1] \qquad \text{Eq. 2-27}$$

$$\mathrm{Pl}(A) = \sum_{B|B \cap A \neq \emptyset} m(B) \qquad \text{Eq. 2-28}$$

with $\mathrm{card}(A)$ being the cardinality and Pl the plausibility of a set A. In case of the entropy, Jiang et al. [JIA09] have proposed to compute a discounting factor for each of the fused masses $\boldsymbol{m}_i$ as follows

$$\gamma_i^{(J)} = 1 - \frac{S_{ei}}{\sum_{i=1}^{2} S_{ei}} \qquad \text{Eq. 2-29}$$

Note that these measures have also been applied in occupancy mapping to assess the quality of maps [YU15]. Eventually, Ding et al. [DIN02] picked up the idea of entropy as a quality measure and extended it with mutual information to arrive at a so called "generalized correlation coefficient" R_g between two evidential masses as follows

$$R_g(\boldsymbol{m}_1, \boldsymbol{m}_2) = \frac{I(\boldsymbol{m}_1, \boldsymbol{m}_2)}{\sqrt{S_e(\boldsymbol{m}_1) S_e(\boldsymbol{m}_2)}} \qquad \text{Eq. 2-30}$$

This is used in [SU18] to define a discount factor to remove redundancy in one of the masses as follows

$$\gamma^{(S)} = 1/R_g(\boldsymbol{m}_1, \boldsymbol{m}_2) \qquad \text{Eq. 2-31}$$

2.4.3 Combination of Evidence during Training

In this subsection, it will be discussed how to train neural networks to predict evidential mass functions. Real world datasets almost always contain disturbances in the form of noise and outliers. In both cases, the network is faced with similar input information and either slightly or sometimes substantially differing targets. Here, the network implicitly combines the input information in order to arrive at a single estimate when faced with similar inputs during inference. As an example, in case of identical inputs but differing outputs, training the network with a mean-squared error results in taking the mean of the provided targets [GOO16]. In case of slightly differing inputs, the combination result not only depends on the loss but also on the distance in input space, the network capacity and the applied regularization. One common way to explicitly handle those disturbances is by training the network to learn a model for them. While this approach is difficult for the outlier case, much work has been published to learn a Probability Density Function (PDF) to capture the data noise, also referred to as aleatoric noise [KEN17].

For the sake of brevity, the following notation shall be introduced which will be used for the rest of this work. Estimated quantities e.g. based on CNN predictions shall be marked with "˜ " while target values in loss functions shall be written as ξ with an index representing the domain the target is created for (e.g. ξ_{p_k} target for a probability value).

In case of regression problems, the majority of works model the noise as a Gaussian distribution [YAN20, FEN19, KEN17]. This can be achieved by adding an additional output channel, interpreting the two channels as mean μ and variance σ and optimizing the following loss function over the N training tuples.

$$\mathcal{L}^{(G)} = \frac{1}{N}\sum_{i=1}^{N}\frac{(\tilde{\mu}_i - \xi_{\mu i})^2}{\tilde{\sigma}_i} + \log(\tilde{\sigma}_i) \qquad \text{Eq. 2-32}$$

For classification, the common approach is to apply Softmax output activation coupled with a cross-entropy loss. This trains the network to predict the weights of a Categorical distribution [GOO16]. To further estimate the aleatoric uncertainties, the network can be altered to predict a Dirichlet distribution over the weights by estimating its shape parameters $\boldsymbol{\alpha}$ which can be written for evidential occupancy mapping as $\boldsymbol{\alpha} = [\alpha_f, \alpha_o]^\top$. These shape parameters can be estimated by either using an exponential [WU19] or ReLU [SEN18] output activation before adding one to each dimension. To further obtain the final predictions, the estimated Dirichlet distribution is treated as a prior to a Categorical distribution and has to be marginalized out. Thus, to train the network, the negative marginal log-likelihood for a single data point i

$$\mathcal{L}_i = -\log\left(\int \prod_{k\in[f,o]} p_{ki}^{\xi_{p_k i}} \frac{1}{\mathrm{Beta}(\tilde{\boldsymbol{\alpha}}_i)} \prod_{k\in[f,o]} p_{ki}^{\tilde{\alpha_{ki}}^{-1}} d\boldsymbol{p}_i\right) = \sum_{k\in[f,o]} \xi_{p_k i}(\log(\tilde{\Sigma}_{\alpha i}) - \log(\tilde{\alpha_{ki}})) \qquad \text{Eq. 2-33}$$

$$\Sigma_{\alpha i} = \sum_{k\in[f,o]} \alpha_{ki} \qquad \text{Eq. 2-34}$$

can be minimized, as proposed in [WU19, SEN18]. Here, $\mathrm{Beta}(\cdot)$ equals the Beta distribution. Sensoy et al. [SEN18] additionally propose to minimize the Bayes risk for the Mean Squared Error (MSE)

$$\mathcal{L}_i = \int \underbrace{\|\xi_{\boldsymbol{p}i} - \tilde{\boldsymbol{p}}_i\|_2^2}_{\text{amount of error}} \underbrace{\frac{1}{\mathrm{Beta}(\tilde{\boldsymbol{\alpha}}_i)} \prod_{k\in[f,o]} p_{ki}^{\tilde{\alpha_{ki}}^{-1}}}_{\text{probability of error}} d\boldsymbol{p}_i = \sum_{k\in[f,o]} (\xi_{p_k i} - \tilde{p_{ki}})^2 + \frac{\tilde{p_{ki}}(1-\tilde{p}_{ik})}{\tilde{\Sigma}_{\alpha i}+1} \qquad \text{Eq. 2-35}$$

as an empirically verified more stable variant. Josang [JOS18] proposed the so called Subjective Logic (SL) framework, where he argues that evidential masses can be expressed as probabilistic Dirichlet distributions using evidence e as the connecting el-

ement. The evidence can be used as follows to transform one representation to the other

$$\boldsymbol{e} = [\alpha_f - 1, \alpha_o - 1]^\top \quad \text{Eq. 2-36}$$

$$\boldsymbol{m} = \frac{1}{\Sigma_\alpha}[\boldsymbol{e}, 2]^\top = \frac{1}{\Sigma_\alpha}[\alpha_f - 1, \alpha_o - 1, 2]^\top \quad \text{Eq. 2-37}$$

$$\boldsymbol{p} = \mathbb{E}[\mathrm{Dir}(\boldsymbol{\alpha})] = \frac{\boldsymbol{\alpha}}{\Sigma_\alpha} \quad \text{Eq. 2-38}$$

using the Dirichlet distribution Dir. This makes it possible to model the aleatoric uncertainties for evidential masses with a network by first learning a Dirichlet distribution as mentioned above and further applying the transformations from eq. Eq. 2-37 [SEN18].

3 Research Approach

The main objective of this thesis is to answer the question of how to extend an already verified ISM with the predictions of an unverified one. More specifically, the scenario of a given geo IRM is considered. It is assumed that this IRM is verified but produces sparse occupancy estimates since it relies on sparse radar detections. This leads to reduced coverage and slow convergence during occupancy mapping. To increase the coverage and convergence speed, the dense interpolations of a learned IRM shall be utilized.

To achieve this, the objective can be divided into the following three goals. First, a model shall be learned from data which is capable of estimating the evidential occupancy state in the close vicinity of the ego vehicle. Here, the focus will lie on measurements in the form of sparse radar detections. However, to showcase the generalizability of the approach, the model will also be applied on two other sensors typically deployed for automated driving, namely camera and lidar data. Next, the deep IRMs estimates shall be fused over time into an evidential occupancy map. Finally, a fusion approach shall be defined to combine the data-driven model's predictions with a geo ISM.

The subsequent sections elaborate the requirements for each of the three aforementioned tasks which is followed by an analysis of the research gap and concluded by the method to solve the problem.

3.1 Requirements

This section details the requirements for both the trained ISM and the fusion approach.

3.1.1 Requirements for deep ISMs

The requirements for the learned ISM are of theoretical as well as practical nature. First, to obtain a generalized measurement representation, BEV grid maps shall be used as inputs. This is the de facto standard for trained ISMs in literature given point cloud inputs (see Section 2.3). Moreover, while it might not be without loss of information, other sensor data e.g. provided by cameras can also be transformed into BEV. Additionally, the outputs shall also be provided as BEV grid maps to ease the later fusion into BEV occupancy maps. Therefore, the following requirement can be formulated.

Requirement 1.1 (R1.1): *The trained ISM must be capable of utilizing the spatial coherence in BEV grid maps and deliver estimates also in the form of BEV grid maps.*

Secondly, the sensor data can include many different sources of noise which is especially true when it comes to radar. In order for the model to learn all these effects,

Requirement 1.2 (R1.2): *the trained ISM must be capable to learn from big data.*

Third, the trained ISM should be obtained as resource efficient as possible. Therefore, the following requirement arises,

Requirement 1.3 (R1.3): *that the amount of manpower, work hours and equipment needed to create the trained ISM should be minimized.*

Also, the data-driven ISM should be able to run in parallel with the already existing geo ISM on the hardware of production-ready vehicles in real-time. The sensor with the highest capture frequency in the NuScenes dataset is the lidar sensor with 20 Hz. To provide some room to perform other computations, it is proposed to aim for a 100 Hz inference time of the trained ISM. To emulate these hardware restrictions, the requirement can be formulated as follows.

Requirement 1.4 (R1.4): *The trained ISM shall be executable with 100 Hz on a single core of a CPU (Intel Core Processor i7-10750H).*

For this work, the width and height of input and output grid maps shall be 128×128 cells for an area of 40 m $\times$ 40 m. This is an acceptable range for low speed scenarios like parking. Additionally, the resolution of 31.25 cm per cell is satisfactory for the trained ISM, as it is mainly used to enhance the geo ISM. Thus,

Requirement 1.5 (R1.5): *the input and output grid maps shall cover an area of* 40 *m* $\times$ 40 *m with* 128×128 *cells.*

Finally, on the theoretical side, the model shall estimate the evidential classes as defined in 2.1, leading to the following requirements.

Requirement 1.6 (R1.6): *The predicted output should capture the amount of free, occupied and unknown information.*

Requirement 1.7 (R1.7): *The unknown mass should be an inverse measure for the overall information content capturing both uncertainty and lack of information.*

Requirement 1.8 (R1.8): *The amount of conflicting information, which is realized by mass being evenly distributed both into the free and occupied class, shall be an indicator for dynamic objects.*

3.1.2 Requirements for Usage of deep ISMs as Priors in Occupancy Mapping

To enable the fusion of the trained ISMs predictions with the outputs of the geo ISMs into a common representation, the trained ISM estimates have to suffice the additional specification for ISMs used in occupancy mapping, as defined in Section 2.1. Namely,

Requirement 2.1 (R2.1): *the trained ISM estimates have to be informational independent over time.*

Additionally, trained ISMs do also provide predictions in regions further away from sensor measurements through means of data-driven interpolation. This poses the potential to overwrite highly certain predictions close to data through many uncertain predictions accumulated over time. Therefore, the accumulation of trained ISM estimates shall be performed in a way that

Requirement 2.2 (R2.2): *regions assigned with high certainty shall not be overwritten by many estimates with low certainty.*

Eventually, as mentioned above, the geo ISM is considered to be a verified, production-ready model which shall solely be enhanced by the trained ISM estimates to increase convergence speed and spatial coverage of the occupancy maps. Thus, given enough measurements, the geo ISM should be trusted over the trained ISM resulting in the following requirements.

Requirement 2.3 (R2.3): *The occupancy map shall be initialized with the trained ISM's estimates up to the point when a definable amount of measurement information has been collected.*

Requirement 2.4 (R2.4): *The occupancy map shall converge to the geo ISM with increasing amount of measurements.*

3.2 Review of the State-of-the-Art and Research Needs

In this section, the SotA, as defined in Chapter 2, is analyzed with respect to the requirements defined in Section 3.1.

3.2.1 Research Needs and Review of State-of-the-Art for geo ILMs and IRMs

The first step in order to analyze trained ISMs is the definition of a reference and baseline model. To generate BEV occupancy targets resource efficiently, as demanded in R1.3, the SotA in all cases relies on automatic label generation using geo ILMs from

360° spinning lidars. For most automated driving test vehicles, these types of lidars are already deployed for verification, making lidar data easily accessible. Moreover, after once manually defining the geo ILM, no additional manual labor is required. Therefore, the automatic generation of BEV occupancy target via geo ILMs suffices R1.3.

To generate geo ILMs, the SotA procedure only contains one problem which revolves around the fact that the radar sensors used in this work have reduced perception capabilities when compared to the lidars. More specifically, the deployed radars can detect objects in 3D but, due to their antenna configuration, can only distinguish the 2D position and velocity. These measurements are additionally filtered and broken down to only a few detections, e.g. 64 detections for the radars in this work. Based on this, the question arises which lidar detections should be filtered out to obtain the best overlap between the two sensor modalities. So far in literature, only threshold-based ground plane removal has been proposed to adapt the lidar detections and only for specific datasets not including NuScenes. Since a proper overlap between target and input information is important to reduce potential outliers and because the specific sensor orientation might have an influence on the perceptive capabilities, a more thorough investigation based on the sensor setup given in NuScenes shall be performed. To narrow the methods down, the often applied threshold-based method shall be compared with different semantic segmentation-based ground plane removal results. This leads to the question of

Research Question 1 (RQ1): *which of the following methods results in the best overlap with respect to Intersection over Union (IoU) between accumulated lidar and radar ISMs: threshold-based or semantic segmentation-based ground plane removal.*

To be able to investigate RQ1, it is also necessary to define the geo IRM algorithm. This geo IRM will be used both, to define the best overlap between the radar and lidar modality and as a baseline to further evaluate trained ISM variants. Regarding the geo IRM's SotA, two problems can be identified. First, the dampening factor, proposed in the SotA, reduces the influence of outliers and multi-path reflections depending on the detection range, angle and ego vehicle velocity. Preliminary experiments have shown that this results in a computationally heavy geo IRM with many hyperparameters. To tune this geo IRM for generalized urban scenarios, many variants of the computational heavy model have to be evaluated on big data. This makes the SotA geo IRM impractical to tune in practice. Therefore, a simplified version of the geo IRM is used in this work which does not specifically scale the IDM based on range, angle and ego vehicle velocity.

The second problem revolves around the enrichment of free space. Here, the SotA proposes to define additional free space between the maximum absolute angles of the

radar's FoV cone and its closest detection. This assumes high certainty of the sensor towards the edges of the FoV cone in order to make the implication of free space based on absence of detections. The remaining literature, however, does not support this assumption, which can be seen by the scaling factor G_φ in Eq. 2-14 which reduces the IDM's certainty towards the FoV's edges in angular direction. Thus, the need arises to provide a free space enrichment method which better suits the sensor certainty, leading to the question of

Research Question 2 (RQ2): *how a procedure can be defined which enriches the baseline IDM ray casting free space predictions in regions lacking detections in a way that strictly increases the mIoU between radar- and lidar-based occupancy maps.*

3.2.2 Research Needs for deep, evidential ISMs

Given the target and baseline model, the creation of trained ISMs can be studied in more detail. Here, the current SotA approaches all tackle the creation of trained ISMs by applying CNNs in the form of UNets with skip connections. While the interpretation of the input varies between it being an image or some kind of multi-channel BEV grid map, CNNs are an appropriate choice for they are designed to leverage spatial context from matrix-like data. Moreover, through breakthroughs like Dropout for regularization, BatchNorm for normalization and skip connections for conservation of information, current CNN models can be designed with increased number of stacked layers. This results in increased modeling capacities allowing them to capture the information from big amounts of data. Thus, the application of UNets as the de facto standard model already suffices R1.1 and R1.2.

Regarding the resource efficiency during inference, literature only discloses run time information on GPUs. Therefore, a rough architecture search shall be conducted for UNets with skip connections and ResNet layers to answer the following question.

Research Question 3 (RQ3): *How should the amount of filters of a UNet architecture be chosen to maximize performance while sufficing the run time requirement as stated in R1.4?*

Additionally, the majority of deep ISMs in the literature model the problem in the probabilistic framework which does not suffice R1.6. On the evidential side, the problem is either modeled as a three class classification or a regression task, both of which suffice R1.6. However, to the best of the author's knowledge, none of the published methods model dynamic objects by distributing mass equally to the free and occupied class. Training on dynamic object targets disqualifies the standard classification approach, since the dynamic object targets are not represented as a one-hot encoding. On the

other hand, regression problems can cope with continuous targets. Thus, the training of deep ISMs will be defined as a regression problem in this work. However, no prior work has provided a thorough comparison of deep ISM's capabilities to estimate evidential masses given different sensor inputs. Which is especially true for the dynamic class which is in most of the literature not modeled for occupancy mapping. Thus, the research question arises

Research Question 4 (RQ4): *to which extent, as measured by the normed confusion matrix (see Section 3.3.8), are the proposed deep ISMs capable to estimate the position of dynamic, free and occupied areas given radar, camera and lidar data respectively.*

Research Question 5 (RQ5): *Additionally, to which extent, as measured by the normed confusion matrix (see Section 3.3.8), does the capability of the proposed deep ISMs to estimate the position of dynamic, free and occupied areas given camera and lidar data respectively change, when radar information is added?*

Regarding the encoding of radar detections for deep ISMs, literature proposes to either encode the positions of a single timestep or use the temporally accumulated point cloud respectively projected into BEV. However, to the best of the author's knowledge, no investigation on how to encode the motion information of the detections has been conducted. Therefore, a rough investigation distinguishing three approaches shall be conducted which is formulated in the following research question.

Research Question 6 (RQ6): *To which extent, as measured by the normed confusion matrix (see Section 3.3.8), does the capability of the proposed deep ISMs to estimate the position of dynamic, free and occupied areas change choosing the input to be BEV projected radar detections...*

- *...of a single timestep encoding the dynamic detections with half intensity?*
- *...accumulated over a certain time horizon only encoding the dynamic detections with half intensity of the latest timestep?*
- *...accumulated over a certain time horizon linearly reducing the intensity of dynamic detections over time starting at half intensity?*

Looking at the uncertainty estimation, neither the classification nor the regression approaches for deep evidential ISMs do explicitly handle occurring aleatoric uncertainties, letting us arrive at the following hypothesis.

Hypothesis 1 (H1): *In case of occurring aleatoric uncertainty, the current state of the art deep ISMs distribute the mass evenly into the free and occupied class rather than shifting it to the unknown class.*

In case H1 holds, these models would, thus, lack the possibility to distinguish between dynamic objects (per definition of evidential occupancy mapping indicated by mass equally distributed into free and occupied class) and regions of high uncertainty. Additionally, the unknown class cannot be used as a measure of information content, since some of the uncertainty is distributed into the free and occupied classes. This behavior would violate the requirements R1.7, hence, raising the following research question.

Research Question 7 (RQ7): *How can a deep, evidential ISM be defined to separate conflicting mass due to aleatoric uncertainty into the unknown class while leaving conflicting mass due to dynamic objects untouched?*

3.2.3 Research Needs for Usage of deep, evidential ISMs as Priors in Occupancy Mapping

With regards to occupancy mapping with deep ISMs, not much literature is available. To the best of the author's knowledge, the only instances of occupancy mapping with deep ISMs use the standard Bayes filtering approach which does not cover evidential models. Thus, the first step consists in analyzing the characteristics and identifying shortcomings when applying the proposed evidential, deep ISMs for occupancy mapping.

First, in contrast to geo ISMs which only provide estimates in regions directly affected by data, deep ISMs additionally perform interpolations in intermediate regions and even extrapolations in regions further away from data. To illustrate the issue arising from this behavior, consider the example depicted in Fig. 3-1. Here, a scenario is shown in which the ego vehicle only partially observes a wall to its left-hand side for the first two time steps. Based on the majority of observations captured in the training dataset, the network might tend to extrapolate the wall as rectangular. In the standard occupancy formulation, this information is treated as independent and, thus, accumulated. When the vehicle finally obtains measurements of the wall's contour in the formerly occluded area, the extrapolation might have already be accumulated to high certainty. Therefore, many estimates based on measurements of this area would have to be accumulated to correct the assigned training data bias. In a similar way, this effect can also lead to overwriting areas with predictions close to data with later occurring extrapolations. This thought experiment leads to the following hypothesis.

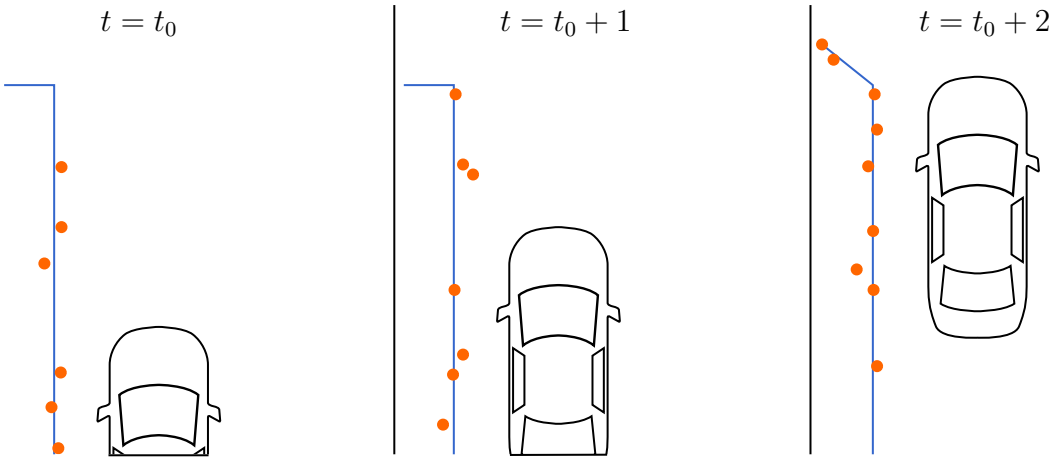


Fig. 3-1: Illustration of informational dependence between deep ISM predictions over time on the example of dataset bias. Here, the ego vehicle (black) drives along a wall and obtains radar measurements (orange) over three time steps. In each time step, the contour of the wall is estimated (blue).

Hypothesis 2 (H2): *Due to inter- and extrapolation in areas not directly measured, deep ISMs contain informational dependence between time steps. This leads to accumulation of bias and or falsification of previously correct assigned areas when their estimates are fused into the occupancy maps using a combination rule that assumes informational independence.*

In case H2 holds, literature in Section 2.4.2 suggests to either remove the redundancy before combination or adapt the combination rule itself to account for the redundancy. This work will focus on removing the redundancy beforehand using Eq. 2-25 since the Yager and Dempster combination rule, as defined in Section 2.4.1, provide well studied, often used fusion methods for evidential occupancy mapping. To remove the redundancy, half of the approaches in the literature focus on defining a constant redundancy weighting between sensor modalities. This is, however, not applicable for the setup in this work, since there is only one sensor modality and the dependency is on the environment.

The other half proposes approaches to measure the amount of information in each to be fused mass and compare them using the mutual information, as stated in Eq. 2-30. However, no general procedure is proposed to measure the mutual information in signals. Thus, the question emerges

Research Question 8 (RQ8): *how can the mutual information be measured to asses the informational redundancy in evidential occupancy mapping.*

Research Question 9 (RQ9): *Also, how can a discount factor be defined based on the mutual information to remove the amount of redundant information between two evidential occupancy masses?*

Additionally, the problem arises that the evidential representation for occupancy mapping slightly differs from the standard in that it defines the conflict state as a meaningful transition state and not as another source of uncertainty based on contradictory information sources. Therefore, the regularly used entropy and the corresponding discount factors cannot be utilized in the proposed form. Also, the second commonly used measure, namely the specificity, quantifies how much mass is distributed into single element classes like the free and occupied class in contrast to the unknown which is both free and occupied. However, the specificity for evidential occupancy mapping is in the interval $[0.5, 1.0]$ which can easily be seen by examining Eq. 2-26. Thus, even for a total lack of information indicated by $\boldsymbol{m} = [0, 0, 1]^\top$, the specificity equals 0.5 disqualifying it as a direct measure for information, too. This leads to the following research question of

Research Question 10 (RQ10): *how the information content in evidential occupancy mapping can be measured.*

Eventually, to utilize the deep ISM estimates as priors according to R2.3 and 2.4, a procedure has to be developed to answer the question of

Research Question 11 (RQ11): *how the influence of the deep ISM can be disabled in case a definable amount of data has been collected, as defined in R2.3 and 2.4.*

Finally, since the evidential occupancy mapping using deep ISMs has not been discussed in the literature so far, the resulting evidential maps given different sensor modalities shall be analyzed and compared. This can be formulated as follows.

Research Question 12 (RQ12): *To which extent, as measured by the normed confusion matrix (see Section 3.3.8), does the capability of evidential occupancy maps to capture the ground-truth change using the proposed deep ISM with different sensor modalities?*

3.2.4 Choice of Dataset

In order to answer the above posed research questions, a dataset containing big amounts of urban recordings of lidar with semantic information, camera, odometry and radar is required. Additionally, it would be nice to use a public dataset, for it allows comparability and reproducability. At the start of the thesis' implementation phase, the

only dataset sufficing these requirements is the publicly available NuScenes dataset [CAE20], which can be seen in the dataset comparison provided in Table 1 in the paper.

The dataset is separated into 1000 so called scenes each containing the data of roughly a 20 second drive during which data from each modality is recorded, which is referred to as sensor sweeps. Additionally, so called samples are defined every 0.5 seconds containing annotations like bounding boxes and semantics for all sensor modalities. To enable comparability, the train-val-test split is predefined by the NuScenes creators. However, all scenes tagged with "night" or "difficult lighting" have been filtered out since they are relatively rare and thus, the networks with camera inputs could not properly adapt. Additionally, some scenes contain little to no ego vehicle movement (e.g. ego vehicle waiting at a red traffic light) which are great scenarios for tracking tasks but largely violate the static environment assumption in occupancy mapping. Thus, only scenes in which the ego vehicle moved at least 20 m are considered. The full list of train, validation and test scenes used in this work is provided in Section 1.1.

Regarding the sensor setup, six cameras three of which face front left, center, right and three of which face back left, center, right are installed. Each camera captures 1.4MP images at a rate of 12 Hz. Additionally, five 77 GHz Frequency-Modulated Continuous-Wave (FMCW) radars are mounted to the car's front left, center, right and back left and right. To obtain ground-truth depth information, a 32 beam spinning lidar is mounted to the roofs center that captures frames with 20 Hz. The pose information is based on a fusion of lidar odometry, Global Positioning System (GPS) and Inertial Measurement Unit (IMU). For more details, the reader is referred to Table 2 in the paper.

3.3 Overview of Methodology

In this section, the framework is presented to address the research questions defined in Section 3.2.3. The framework extents the standard evidential occupancy mapping pipeline [PAG96] to incorporate estimates of a data-driven ISM, as shown in Fig. 3-2. The incorporation of the deep ISM's information is realized by fusing the estimates directly into the map rather than first fusing it with the geo ISM's estimate. This is done, in order to enable an asynchronous fusion of information into the map, which allows for differing execution times of the ISMs. In order to tackle the problem of temporal redundancy, as mentioned in H2, a specific fusion approach is defined for the deep ISM. This general procedure is detailed in the subsequent subsections as follows.

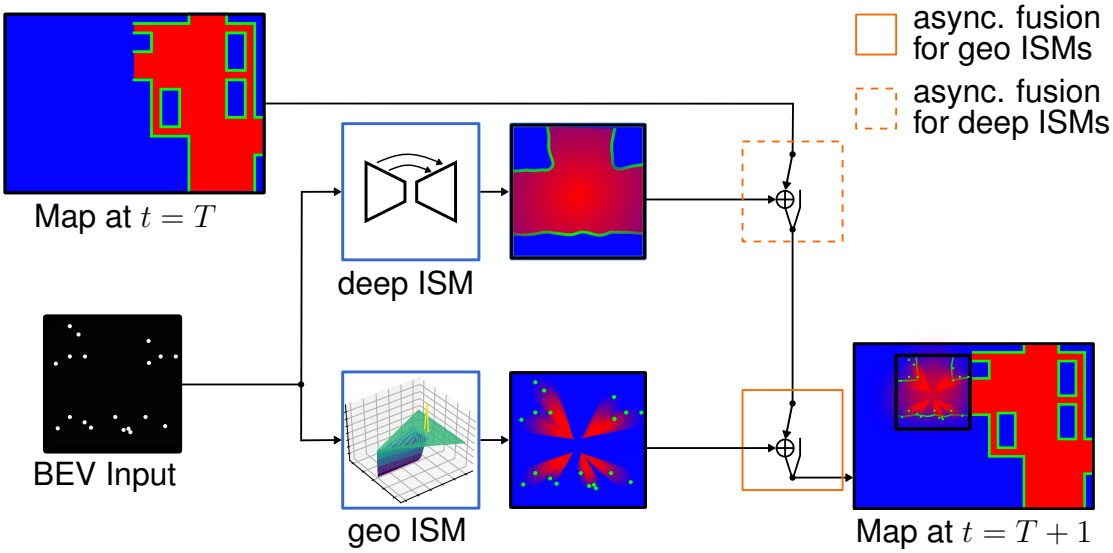


Fig. 3-2: Structural overview of the proposed framework, showing how the BEV input is transformed both by the deep and geo ISM into occupancy estimates.

First, the creation of ground-truth lidar and baseline radar occupancy maps is discussed. Here, a method is shown in Section 3.3.1 to interpolate the annotations of dynamic objects for the higher frequency intermediate lidar detections. This is followed by the definition of the geo ILM and IRM used in this work in Section 3.3.2 and 3.3.3. Eventually, a method is described in 3.3.4 to maximize the overlap between the ground-truth lidar and baseline radar occupancy maps.

The definition of baseline and ground-truth data is followed by defining the deep ISM. Starting with the description of the different inputs and the target inspected in this work in Section 3.3.5, followed by the architecture search in Section 3.3.6, the investigated methods to account for aleatoric uncertainty in Section 3.3.7 and concluded by a description of the used metric in Section 3.3.8. The specific choice of the fusion methods for both geo and deep ISMs are detailed in Section 3.3.9 and 3.3.10.

3.3.1 Method to provide dynamic Information for Lidar Sweeps

To obtain the dynamic information for lidar sweeps, the sample's bounding boxes of dynamic objects are interpolated for intermediate sweeps and all lidar detections intersecting with the boxes are marked as dynamic. Here, the interpolations are obtained as follows. First, corresponding bounding boxes of dynamic objects are identified by their track ids for two subsequent samples. To avoid singularities, the bounding box poses of each identified pair are first transformed into the temporally first bounding box's coordinate frame. Next, a third degree polynomial is used to interpolate the poses. The polynomial is defined to intersect with the positions of the provided poses. Also, the polynomial's derivative at the interpolation points has to equal the tangents of

the pose's angle. Thus intermediate poses should have positions coinciding with the polynomial with orientations given by the arc-tangent of the polynomial's derivative.

To finally obtain the interpolated bounding box pose, the path along the interpolated trajectory is numerically integrated and divided into equidistant segments. Under the assumption that the tracked, dynamic object travels with a constant velocity between the two samples, the interpolated pose is given at the point when the relative traveled distance between the first and second sample's pose is closest to the relative time of the sweep between the two samples. The so identified pose can then be transformed into the coordinate frame of the currently interpolated sweep. The interpolation procedure and the overlay of the resulting interpolated bounding boxes over a lidar sweep's detection image are illustrated in Fig. 3-3 in Cartesian coordinates $[\boldsymbol{e}_x, \boldsymbol{e}_y]$.

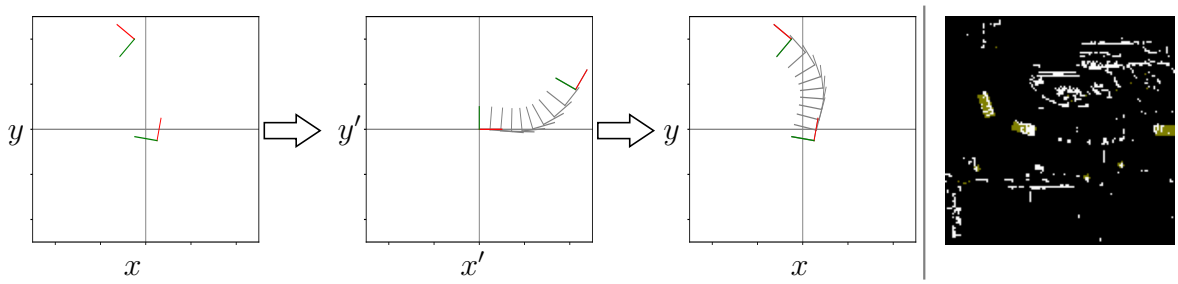


Fig. 3-3: Illustration of the three steps to obtain interpolations of the 2d bounding box poses on the left hand side together with an example of resulting interpolated bounding boxes (yellow) overlayed on a lidar sweep's BEV detection image on the right. The three interpolation stages from left to right show the original bounding box poses of two consecutive samples, the poses transformed into the first poses coordinates together with the interpolated poses (gray) and finally the poses back in the original coordinate frame with the interpolations.

3.3.2 Definition of the geo ILM

Both lidar and radar sensors use a radial measurement principle which is why it makes sense to model them using the ray casting ISM as explained in Section 2.2.1, however in a modified way. In case of the ILM, first the ground detections have to be removed. To do so, either a height-threshold or a semantic segmentation based approach can be chosen. The specific choice used in this work is detailed in Section 4.1.

Afterwards, free space rays with opening angle $\varphi_\triangleleft$ are cast in non-overlapping increments of $\varphi_\triangleleft$ in the range $[0°, 360°]$ around the sensor center either up to a maximum distance r_{max} or until intersecting with a detection. All BEV grid cells affected by the free space ray are set to $[M_F, 0, 1 - M_F]^\top$ using the tunable free space weight $M_F \in [0, 1]$. On the other hand, positions of collided detections are either set to $[0, M_O, 1 - M_O]^\top$ for

static detections with the occupied space weight $M_O \in [0, 1]$ or to $[M_D, M_D, 1 - 2M_D]^\top$ for dynamic detections with the dynamic object weight $M_D \in [0, 0.5]$. Here, the information about the motion state is not provided by the lidar itself but by external sources like additional labels in the dataset. The additional motion information will only be used to generate labels to train the deep ISM variants but not to generate deep ISM inputs! This procedure has the effect that detections occluded in the BEV projection are filtered out which better matches the radar's sensing capability. An algorithmic summary of the geo ILM in pseudo code is shown in Alg. 1.

Algorithm 1: Geo ILM

Hyperparameters: $M_F, M_O, M_D, \varphi_\triangleleft$
function `castFreeSpace`
    `// see` Eq. 3-1 `with` $M_O = M_D = 0$
remove ground-plane detections
initialize all cells in the ILM image to $m_u = 1$
for φ_{center} `in discretizedAngles360` **do**
    if `(at least one detection in current cone area)` **then**
        $(r_{\text{det}}, \varphi_{\text{det}})$ = `getClosestDetectionInsideConeArea(`φ_{center}`,` $\varphi_\triangleleft$`,` r_{max}`)`
        `castFreeSpaceRay(`φ_{center}`,` r_{det}`,` $\varphi_\triangleleft$`,` M_F`)`
        `// mark closest detection`
        if `(detection is static)` **then**
            `assignCellValue(`$(r_{\text{det}}, \varphi_{\text{det}})$`,` $[0, M_O, 1 - M_O]$`)`
        else
            `assignCellValue(`$(r_{\text{det}}, \varphi_{\text{det}})$`,` $[M_D, M_D, 1 - 2M_D]$`)`
    else
        `castFreeSpaceRay(`φ_{center}`,` r_{max}`,` $\varphi_\triangleleft$`,` M_F`)`

3.3.3 Definition of the geo IRM

For the geo IRM, a cone model which is based on the $\text{IDM}_{\text{B-Spline}}$ as proposed in [LOO16] is used with the following adjustments. First, since the whole uncertainty ellipse of a radar detection falls within one of the 31,25 cm $\times$ 31,25 cm grid cells in the BEV grid, a scaled, evidential version of the $\text{IDM}_{\text{ideal}}$ can be used as an approximation of the $\text{IDM}_{\text{B-Spline}}$'s radial component.

In angular direction, it is proposed to linearly approximate the influence of the Gaussian noise, again due to the coarse discretization. However, cones with enlarged opening angles are used to counteract cases of IDM rays passing between detections in sparsely measured areas. While the enlarged cones lead to a desired enrichment

of free space they would also lead to blurring the occupied space and with it, object boundaries. Thus, only the free space follows the linearly approximated Gaussian noise in angular direction while the detection's influence is, again, restricted to its occurring cell. Thus, the following adapted evidential IDM is proposed to be used for radars

$$\text{IDM}_{radar}(r, \varphi, r_{\text{det}}, \varphi_{\text{det}}, d_{\text{det}}) = \begin{cases} [M_{F\triangleleft}, 0, 1 - M_{F\triangleleft}]^\top & , \Delta\varphi \leq \frac{\varphi_\triangleleft}{2}, r < r_{\text{det}} \\ [0, M_O, 1 - M_O]^\top & , \varphi = \varphi_{\text{det}}, r = r_{\text{det}}, d_{\text{det}} = \text{False} \\ [M_D, M_D, 1 - 2M_D]^\top & , \varphi = \varphi_{\text{det}}, r = r_{\text{det}}, d_{\text{det}} = \text{True} \\ [0, 0, 1]^\top & , \text{ else} \end{cases} \quad \text{Eq. 3-1}$$

$$\text{with} \quad M_{F\triangleleft} = M_F(1 - \frac{\Delta\varphi}{\varphi_\triangleleft}) \quad \text{and} \quad \Delta\varphi = |\varphi - \varphi_{\text{det}}| \quad \text{Eq. 3-2}$$

Eq. 3-3

With r and φ being the cylindrical coordinates measured in the radar's sensor coordinate frame, r_{det} and φ_{det} being the cylindrical coordinates of the detection and d_{det} the Boolean flag indicating the dynamic state of the detection. However, for reasons detailed in Section 2.2.4, IDM_{radar} can not be directly applied on each detection to obtain the IRM. Here, the main problem is the assumption that only objects in line of sight are detected. This assumption is violated for the radar since objects can be detected in occluded areas due to multi-path reflections. This leads to big amounts of free space being assigned to occupied areas by IDM rays cast towards detections in occluded areas.

In this work, the occlusion problem is addressed similar to [WER15] with the alteration that the free space part of the IDM is not only ignored in occupied regions of other rays, but ends at it. In case the contours of all objects are densely detected, this would suffice to solve the problem. However, radar detections are also sparse which is why measurements of preceding time steps are additionally used to densify object boundaries. In contrast to the ILM, detections occluded in the BEV projection shall still be marked as occupied in the IRM in order not to further increase the sparseness. The time horizon, in which detections are accumulated, shall be set to a finite value to dampen the influence of potential outliers. This procedure only introduces the time horizon as hyperparameter and, thus, proposes a good candidate to answer RQ2.

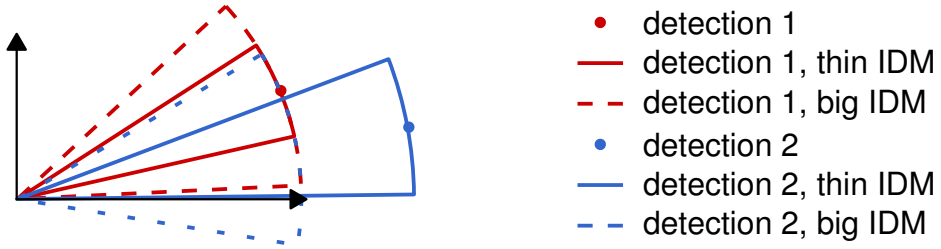


Fig. 3-4: Illustration of the proposed IDM's contours being applied on two detections. It shows the big and thin IDMs being cast for each detection. Also, it shows that the big IDM cone's range for detection two (stripped blue) is limited by the first detection. On the other hand, the thin IDM of the second detection is, for the given scenario, able to assign free space up to the second detection.

Another problem connected with the sparsity is the low coverage of free space. In this work, the method proposed in [PRO18] is altered to find an answer for RQ2 by instead casting the original rays again but with a much wider opening angle. This assumes that the manufacturer's prefiltering of the raw radar signal is done in a way to preserve detections belonging to the object closest to the sensor. The justification for this assumption is based on the fact that radars are applied in safety critical collision avoidance systems. The second iteration of wider IDM rays shall only be used to enrich the free space and, thus, use $M_O = 0$. Additionally, it shall be fused with the first iterations IRM using Dempster's combination rule. An algorithmic summary of the geo IRM in pseudo code is shown in Alg. 2.

Algorithm 2: Geo IRM

Hyperparameters: ${M_F}^{(\text{thin})}, {M_F}^{(\text{big})}, {\varphi_\triangleleft}^{(\text{thin})}, {\varphi_\triangleleft}^{(\text{big})}, M_O, M_D, T$

```
function castFreeSpace
    // see Eq. 3-1 with M_O = M_D = 0
initialize all cells in the IRM image to m_u = 1
transform previous T radar point clouds of all radars to current IRM image
for (r_det, φ_det) in currentDetections do
    // range at which the free space of the ray is stopped
    (r_det, φ_det) = getClosestDetectionInsideConeArea(φ_center, φ_◁, r_max)
    // cast thin and big free space rays
    castFreeSpaceRay(φ_center, r_det, φ_◁^(thin), M_F^(thin))
    DempsterRule(castFreeSpaceRay(φ_center, r_det, φ_◁^(big), M_F^(big)))
    if (detection is static) then
        assignCellValue((r_det, φ_det), [0, M_O, 1 − M_O])
    else
        assignCellValue((r_det, φ_det), [M_D, M_D, 1 − 2M_D])
```

3.3.4 Methodology to define the geo ISM's Hyperparameters and Removal of Ground-Plane Detections

The analysis of which method to chose as a baseline and which as ground truth, according to RQ1 and RQ2, is a chicken and egg problem since one has to be kept fixed to analyze the other. Hence, it is proposed to initially manually define and fix the geo ILM using the threshold-based ground-plane removal as normally applied in the literature. This preliminary geo ILM will act as reference to analyze the free space enrichment (see RQ2) of the geo IRM.

To do so, three variants of the geo IRM (as defined in Alg.2), are compared which subsequently add more of the proposed algorithm's components. First, the $\text{IDM}_{\text{radar}}$ shall only be applied without accumulation of radar detections over time and only as a thin cone. This corresponds to fixing the parameters in Alg. 2 as shown for the first variant in Fig. 3-5. Next, the additional influence of using accumulated detections is analyzed (see Fig. 3-5 #2), followed by the additional application of cones with big opening angles (see Fig. 3-5 #3). These variants are used to create occupancy maps which are compared with maps created from the fixed geo ILM. Here, the maps are created using Yager's rule of combination and the metric for comparison is the mIoU, evaluated within the area reached by the ISMs. To obtain best competitive results for each method, the parameters labeled as "?" for each respective variant in Fig. 3-5 are chosen via a grid search.

	geo IRM Variant	$M_F{}^{(\text{thin})}$	$M_F{}^{(\text{big})}$	$\varphi_{\triangleleft}{}^{(\text{thin})}$	$\varphi_{\triangleleft}{}^{(\text{big})}$	M_O	M_D	T
#1	Ray Casting IRM	?	0	?	0	?	?	1
#2	... + Accumulation	?	0	?	0	?	?	?
#3	... + big Cones	?	?	?	?	?	?	?

Fig. 3-5: Three investigated variants of the geo IRM that subsequently add more features of Alg. 2. The respective parameters marked as "?" are chosen through a grid search. This search is based on mIoU comparison between occupancy maps created with the respective variants and the fixed geo ILM maps.

After the evaluation of the geo IRM, the best performing variant is fixed and used as reference to analyze the effect of different ground-plane removal methods in the creation of lidar-based occupancy maps. Here, a dataset with semantic labels for the lidar detections shall be used to cover the semantic ground-plane filtering approaches. As an alternative, threshold-based filtering shall be considered by removing all lidar detections beneath a certain height threshold. The sensor specifics, considered semantic labels and specification of height thresholds are detailed in Section 4.1.

3.3.5 Deep ISM Targets and investigated Inputs

To analyze the effects and performance of deep ISMs given different sensor inputs, lidar, camera and radar measurements have to be encoded into an input representation. For this work, a common representation for all sensor modalities and the training targets is chosen to be BEV images as further detailed in R1.5. In the following, the steps are detailed to create the lidar, camera and radar BEV image inputs together with the corresponding deep ISM labels.

Starting with the lidar BEV detection images, first, the ground-plane has to be removed. The specific removal method and its parameters are experimentally defined in Section 4.1. Afterwards, the detection positions are discretized into the coordinates of a gray-scale image and marked with value 1.0 as opposed to the default value of 0.0. The so processed lidar BEV image will be referred to as BEV projection of lidar detections without ground-plane (L) and their ISMs as ILM. An example is shown in Fig. 3-6. This encoding has been proposed by the author in [BAU19b].

For the radar images, the sweep information of all corner and the front radar are used. All of the static detections, distinguishable through a flag provided in the NuScenes dataset, are marked as 1.0 opposed to the default pixel value 0.0. To analyze RQ6, three encoding variants will be investigated. First, only the detection information of a single timestep shall be encoded marking the dynamic detections with half intensity (0.5). This encoding is referred to as BEV projection of all radars' detections of the most recent timestep (R). Alternatively, radar detections shall be accumulated over T timesteps, marking the dynamic detections as $0.5(1 - \frac{t}{T})$. This indicates the transition state of dynamic objects between free and occupied by marking the most recent detections as 0.5 and linearly decreasing their influence over time. The time horizon T is chosen by performing a grid search as shown in Section 4.1. This encoding is defined as BEV projection of all radars' dynamic and static detections of the recent T timesteps (R_T). Finally, the third considered encoding again accumulates the static detections over T timesteps. However, only the most recent dynamic detections are marked with 0.5, which is referred to as same as R_T but only projecting the most recent dynamic detection ($\mathrm{R}_{T|1}$). A comparison between R, R_T and $\mathrm{R}_{T|1}$ is presented in Section 4.4 and exemplary illustrations can be found in Fig. 3-6. The corresponding ISMs are referred to as IRM with the "R" being appended by the respective indices per variant. These encodings have been proposed by the author in [BAU19b, BAU19a].

In case of the camera BEV images, the homography projection as well as the MonoDepth projection are considered. To obtain the homography-based BEV images, lidar points are identified which are only 5 cm away from the ideal flat ground-plane. These lidar detections are then transformed both to the BEV and the camera image. After-

wards, the corresponding pixel coordinates in both representations are identified and used to compute the homography matrix using a RANSAC-based filter. After identifying the homography for each camera, the images are projected into the BEV image where overlapping areas are being replaced and areas with no detections remain black. This BEV projection is referred to as homography BEV projection of the RGB values for each camera (C_{RGB}). Additionally, a variant named homography BEV projection of the semantic segmentation for each camera (C_S) is defined where the semantic annotations are transformed using the homography transformation. The semantic labels are being obtained by applying DeepLab V3+ [CHE18] using the Xception network [CHO17] as a backbone trained on the Cityscapes dataset [COR16] without further finetuning. The color coding is taken over from Cityscapes. Examples of C_{RGB} and C_S are shown in Fig. 3-6. The respective ISMs are referred to as Inverse Camera RGB Model ($IC_{RGB}M$) and Inverse Camera Semantic Model (IC_SM). These encodings have been used in previous works, as described in Section 2.3.2.

For the MonoDepth projection, the semi-supervised model as proposed in [GUI20] is used. Here, a model pretrained on Cityscapes is finetuned in a semi-supervised way on the NuScenes data. The resulting point cloud of all cameras is then projected into the BEV image while the pixel intensities represent the scaled height information. More specifically, the height is clipped into the interval $[-0.5, 1.0]$m and scaled to the intensity interval $[0, 1]$. This encoding and ISM will be referred to as BEV projection of MonoDepth prediction for each camera (C_D) and Inverse Camera Depth Model (IC_DM) and an example is shown in Fig. 3-6. To the best of the author's knowledge, this encoding is first used in this work.

To analyze the fusion of camera and lidar respectively with radar measurements, the afore described input representations will be concatenated channel-wise. The combined BEV image is further directly used as an input to perform an implicit fusion inside the deep ISM. For reasons explained in Section 4.5, only the fusion of R_{20} with L and C_D respectively will be investigated in this work. The resulting deep ISMs are referred to as Inverse Lidar and Radar (20) Model ($ILR_{20}M$) and Inverse Camera Depth and Radar (20) Model ($IC_DR_{20}M$). Examples of the combined input for L & R_{20} and C_D & R_{20} are visualized in Fig. 3-6. To the best of the author's knowledge, these specific input concatenations are introduced for the first time in this work.

Finally, the occupancy map patches used as targets to train the deep ISMs are generated as follows. First, the geo ILM, as defined in Section 3.3.2 and 4.1, is used to create an occupancy map of the considered scene. Here, the mapping is conducted by using the interpolated dynamic information, as described in Section 3.3.1, to reduce the effect of artifacts due to dynamic objects. After the creation of the occupancy map, patches, centered around each ego vehicle's position during mapping, are cut from the

map. As a last step, to regain the information of dynamic objects filtered out during mapping, the interpolated bounding boxes are marked in the occupancy map patches. An alternative to assigning the whole bounding box area as dynamic would be to only mark pixels corresponding to dynamic detections. This, however, provides only partial contours of the moving objects and is therefore disregarded. An example of this projection, which will be referred to as ground-truth occupancy map patch used as labels to train the deep inverse sensor models (GT), is shown in Fig. 3-6.

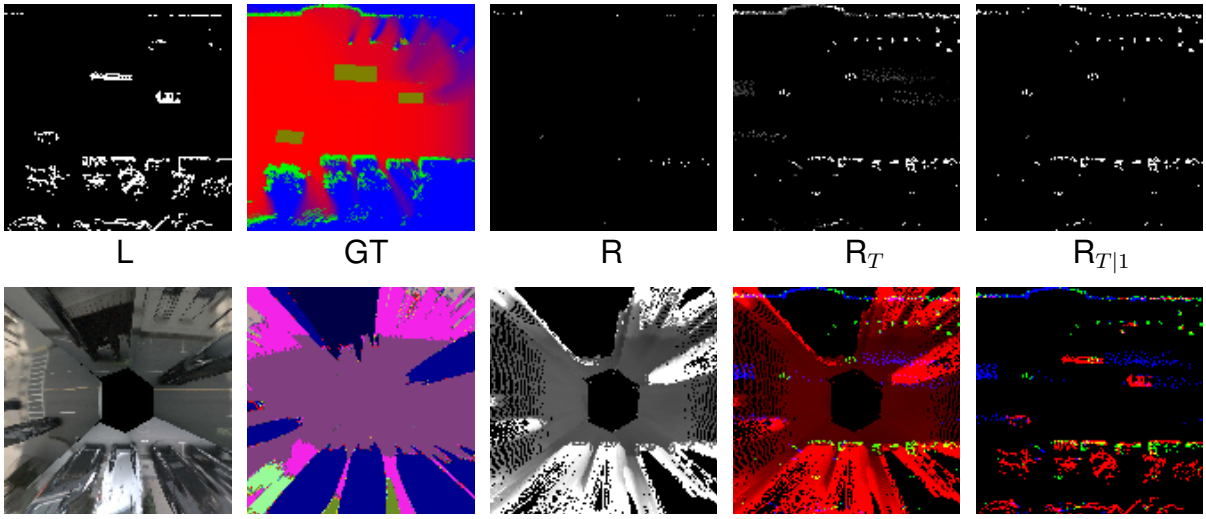
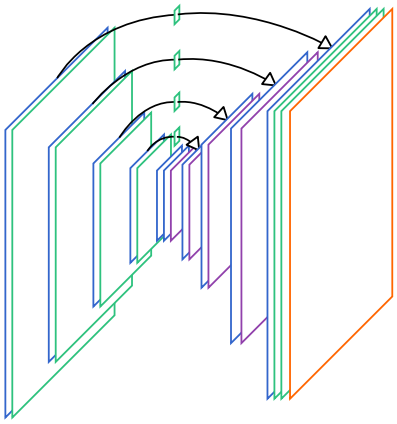


Fig. 3-6: Illustration of the inputs and target images used to train the different deep ISM variants.

3.3.6 Methodology to define the deep ISM Architecture

Next, the network architecture has to be analyzed to suffice RQ3. To restrict the search space to a feasible subset, the architecture as illustrated in Fig. 3-7 is considered. Here, the initial grid dimension is halved after each of the four encoder steps using strided 3×3 convolutions while the number of channels is doubled up to a maximum. This is inverted in the decoder using bilinear upsampling. All convolution layers are structured as suggested in the literature (see Section 2.3.1 and Fig. 2-5). Whenever the grid's height and width remain constant, ResNet layers, as shown in Fig. 2-5, are deployed to increase efficiency. The last three layers consist of two additional convolution layers without Dropout and the output layer, specifically designed for the applied loss. A detailed description of the hyperparameters and the choices made for the architecture search are elaborated in Section 4.2. The layer depth of the network is chosen in a way that a receptive field of 125 is reached which almost covers the whole image (for receptive field computation please refer to Eq. 2-15). This receptive field is chosen to enable the network to learn the principle of occluded areas based on the ego vehicle positioned in the center.

layers	abbr.	parameters
ResNet	$\mathsf{res}(\cdot)$	$\kappa = 3, C_{\text{out}}, D$
Convolution	$\mathsf{conv}_s(\cdot)$	$\kappa = 3, C_{\text{out}}, s$
conv without Dropout	$\underline{\mathsf{conv}}_s(\cdot)$	$\kappa = 3, C_{\text{out}}, s$
bilinear upsampling	$\mathsf{up}_u(\cdot)$	$\kappa = 3, C_{\text{out}}, u$
Concatenation	$\mathsf{cat}(\cdot,\cdot)$	—
Input / Output	x / $\mathsf{y}(\cdot)$	— / —



UNet Architecture

Layers

layers	abbr.	dimension		layers	abbr.	dimension
x	e_{00}	$128 \times 128 \times C_{\text{in}}$		$\mathsf{y}(d_{04})$	d_{05}	$128 \times 128 \times C_{\text{out}}$
$\mathsf{res}_D(e_{00})$	e_{01}	$128 \times 128 \times C_0$		$\underline{\mathsf{conv}}_1(d_{03})$	d_{04}	$128 \times 128 \times 4$
				$\underline{\mathsf{conv}}_1(d_{02})$	d_{03}	$128 \times 128 \times 4$
				$\mathsf{res}_D(d_{01})$	d_{02}	$128 \times 128 \times C_0$
$\underline{\mathsf{conv}}_1(e_{01})$	s_0	$128 \times 128 \times 4$	$\longrightarrow$	$\mathsf{cat}(d_{00}, s_0)$	d_{01}	$128 \times 128 \times C_0$
				$\mathsf{up}_2(d_{12})$	d_{00}	$128 \times 128 \times C_0$
$\mathsf{conv}_2(e_{01})$	e_{10}	$64 \times 64 \times C_1$				
$\mathsf{res}_D(e_{10})$	e_{11}	$64 \times 64 \times C_1$		$\mathsf{res}_D(d_{11})$	d_{12}	$64 \times 64 \times C_1$
$\underline{\mathsf{conv}}_1(e_{11})$	s_1	$64 \times 64 \times 4$	$\longrightarrow$	$\mathsf{cat}(d_{10}, s_1)$	d_{11}	$64 \times 64 \times C_1$
				$\mathsf{up}_2(d_{22})$	d_{10}	$64 \times 64 \times C_1$
$\mathsf{conv}_2(e_{11})$	e_{20}	$32 \times 32 \times C_2$				
$\mathsf{res}_D(e_{20})$	e_{21}	$32 \times 32 \times C_2$		$\mathsf{res}_D(d_{21})$	d_{22}	$32 \times 32 \times C_2$
$\underline{\mathsf{conv}}_1(e_{21})$	s_2	$32 \times 32 \times 4$	$\longrightarrow$	$\mathsf{cat}(d_{20}, s_2)$	d_{21}	$32 \times 32 \times C_2$
				$\mathsf{up}(d_{32})$	d_{20}	$32 \times 32 \times C_2$
$\mathsf{conv}_2(e_{21})$	e_{30}	$16 \times 16 \times C_3$				
$\mathsf{res}_D(e_{30})$	e_{31}	$16 \times 16 \times C_3$		$\mathsf{res}_D(d_{31})$	d_{32}	$16 \times 16 \times C_3$
$\underline{\mathsf{conv}}_1(e_{31})$	s_3	$16 \times 16 \times 4$	$\longrightarrow$	$\mathsf{cat}(d_{30}, s_3)$	d_{31}	$16 \times 16 \times C_3$
				$\mathsf{up}_2(d_{40})$	d_{30}	$16 \times 16 \times C_3$
$\mathsf{conv}_2(e_{31})$	e_{40}	$8 \times 8 \times C_4$				
$\mathsf{res}_D(e_{40})$	e_{41}	$8 \times 8 \times C_4$	$\longrightarrow$	$\mathsf{res}_D(e_{41})$	d_{40}	$8 \times 8 \times C_4$

Encoder Architecture **Decoder Architecture**

Fig. 3-7: Illustration of the UNet variant's architecture used in this work. The skip connections between encoder and decoder are realized with a convolution, compressing the amount of features to 4 channels and, afterwards, concatenating them with the features of a subsequent layer. Convolution and ResNet layers are structured as shown in Fig. 2-5. The skip connection in the ResNet layer is realized by adding the input to the ResNet layers output before applying the non-linearity.

It shall be mentioned that a receptive field covering half the input image size might already suffice for the task at hand. However, since the computation cost shrinks quadratically when downsampling the feature size, adding the additional layers to arrive at the full input image's receptive field is almost negligible.

3.3.7 Methodology to account for aleatoric Uncertainties in deep ISMs

As explained in Section 3.2.2, the baseline deep ISM considered in this work shall model the evidential occupancy estimation using a Softmax output activation (see Eq. 3-4) to transform the last layer's features $\boldsymbol{z}$ normed evidential output $\tilde{\boldsymbol{m}}$. The training loss between the prediction $\tilde{\boldsymbol{m}}$ and the targets $\xi_{\boldsymbol{m}}$ shall be the standard MSE regression loss. This configuration is from here on referred to as **SoftNet** which has been proposed by the author in [BAU19b, BAU19a]. A visualization is shown in the first row of Fig. 3-8.

$$\tilde{m_{ki}} = \frac{e^{\boldsymbol{z}_i}}{\sum_{k\in[f,o,u]} e^{\boldsymbol{z}_k}} \quad \text{Eq. 3-4}$$

$$\mathcal{L}_2 = \frac{1}{N}\sum_{i=1}^{N} \|\xi_{\boldsymbol{m}i} - \boldsymbol{m}_i\|_2^2 \quad \text{Eq. 3-5}$$

In case of occurring aleatoric uncertainty, the network is confronted with both information indicating a pixel to be free and occupied, leading to H1. To investigate H1 further, first, the architecture as specified in Section 3.3.6 will be trained in the SoftNet configuration using the data as specified in Section 3.3.5. Next, the normed confusion matrix, as detailed in Section 3.3.8, will be computed for the test data predictions. The normed confusion matrix is an expansion of the commonly used IoU and thus provides more detailed insights. Additionally, two alternatives to SoftNet shall be investigated to potentially answer RQ7.

For the first variation, the evidential occupancy classes are extended to separately model the dynamic class instead of mixing it into the free and occupied classes(e.g. $[0.5, 0.5, 0] \rightarrow [1, 0, 0, 0]$). With regards to the network, the Softmax output as well as the MSE loss have to be extended by one dimension to realize this configuration. Additionally, to adapt the targets, an operation which shifts the occupancy labels from the evidential $\boldsymbol{m} = [m_f, m_o, m_u]^\top$ to the extended representation $\mathring{\boldsymbol{m}} = [\mathring{m}_d, \mathring{m}_f, \mathring{m}_o, \mathring{m}_u]^\top$, sufficing R 1.6, 1.7 and 1.8, can be defined as follows

$$\text{shift extension} \quad \mathring{\boldsymbol{m}} \succ (\boldsymbol{m}) \quad \text{Eq. 3-6}$$

$$\mathring{m}_d = 2 \cdot \min(m_f, m_o) \quad \text{Eq. 3-7}$$

$$\mathring{m}_k = m_k - \min(m_f, m_o), \quad k \in [f, o] \quad \text{Eq. 3-8}$$

$$\mathring{m}_u = m_u \quad \text{Eq. 3-9}$$

Here, $\min(\mathring{m}_f, \mathring{m}_o)$ describes the amount of mass being equal in both the free and occupied class. This portion is extracted from both classes, which doubles its amount, and shifted to the newly created dynamic class, leaving the unknown class untouched. Moreover, to use estimates in the extended representation $\tilde{\mathring{\boldsymbol{m}}}$ in the evidential occupancy mapping pipeline, a compression operation can be defined as follows

$$\text{shift compression} \quad \boldsymbol{m} \prec (\mathring{\boldsymbol{m}}) \tag{Eq. 3-10}$$

$$m_k = \mathring{m}_k - \min(\mathring{m}_f, \mathring{m}_o) + \frac{\mathring{m}_d}{2}, \quad k \in [f, o] \tag{Eq. 3-11}$$

$$m_u = \mathring{m}_u + 2 \cdot \min(\mathring{m}_f, \mathring{m}_o) \tag{Eq. 3-12}$$

This operation quantifies the learned aleatoric uncertainty between the free and occupied class as the amount of mass equally distributed into free and occupied ($\min(\mathring{m}_f, \mathring{m}_o)$) following H1. Afterwards, the equal portion is extracted from their respective classes and shifted to the unknown class. Also, the dynamic mass is split into equal portions and added to the free and occupied class respectively, to account for R 1.8. The network trained on the extended labels $\xi_{\mathring{\boldsymbol{m}}}$ and capable of producing evidential estimates $\tilde{\boldsymbol{m}}$ by applying the extend operation defined in Eq. 3-10 is from here on referred to as **ShiftNet**. To the best knowledge of the author, this method is investigated in this work for the first time. The ShiftNet method is depicted in the second row of Fig. 3-8.

The third method is heavily based on the method proposed by Sensoy et al. [SEN18]. In their method, a Dirichlet network is trained on the Bayes risk of the MSE (see Eq. 2-35) to model aleatoric uncertainty and SL (see Eq. 2-37 and 2-38) is used to transform the Dirichlet PDF to evidential masses. This will be referred to as **DirNet** and has been proposed by the author in [BAU19a]. A depiction is shown in the last row of Fig. 3-8. However, in the original formulation of Sensoy et al., all unknown mass is solely due to aleatoric uncertainty. But, in the evidential occupancy formulation, the unknown mass both represents uncertainty and lack of information. Thus, to additionally provide labels for the unknown mass e.g. in unobserved areas, the loss from Eq. 2-35 shall be altered as follows

$$\mathcal{L}_i = (\xi_{m_u i} - \tilde{m}_{ui})^2 + \sum_{k \in [f,o]} (\xi_{m_k i} - \tilde{m}_{ki})^2 + \frac{\tilde{m}_{ki}(1 - \tilde{m}_{ki})}{\Sigma_{\alpha i} + 1} \tag{Eq. 3-13}$$

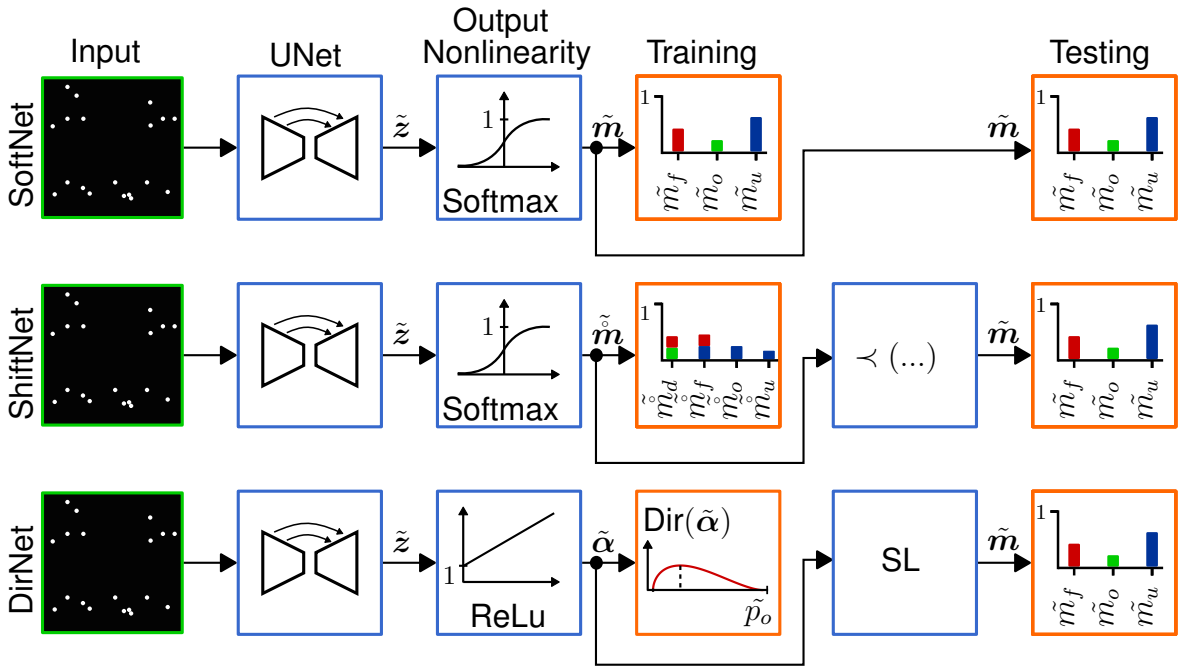


Fig. 3-8: Illustration of the three deep ISM configurations investigated in this work.

To account for the class imbalance in all of the above mentioned deep ISM variants, the mean loss is computed separately for the labels of each class and afterwards summed up over all classes to obtain a final score. The performance of these variants is compared in Section 4.3 to analyze H1 and answer RQ7.

3.3.8 Metric to evaluate the ISM Variants

To quantify the ISM and occupancy mapping results, a confusion matrix variant is deployed since it provides a more detailed means of analysis as compared to the mIoU score. The confusion matrix variant is constructed in the following steps

1. obtain the current lidar occupancy map patch together with the current model's estimates
2. define the ground-truth pixels by assigning each pixel to the class with the highest mass in the lidar occupancy map patch
3. for each ground-truth pixels belonging to a class, sum up how much weight has been estimated to belong to each class (e.g. for all ground-truth occupied pixels, how much mass is estimated to be free, how much occupied, ...)
4. divide the sum of estimated mass for each class by the amount of ground-truth pixels for the current class
5. to cope with the amount of pixels per class over the whole dataset, the above is done on an image basis and averaged over all test samples

This leads to the probability $p(\tilde{\hat{m}}_k|\xi_{\hat{m}_k})$ to estimate class k for a pixel given the pixel belongs to the actual class k with $k \in [d, f, o, u]$. This probability will be abbreviated as $p(\tilde{k}|k)$.

Since all of the investigated models behave severely differently in the visible and occluded area, it is necessary to provide independent metrics for these two cases. Thus, the metrics are additionally computed for visible and occluded areas, while the occluded area is defined as the pixels where the geo ILM's unknown mass surpasses the other class' masses. Thus, only a stochastically insignificant portion of unknown labels remains in the visible area, which is why these scores are excluded from consideration.

To help the interpretation of the confusion matrix, values are marked in green to show true positives while red shows the percentages of false predictions. The unknown class predictions, marked in black, are treated as the safe state and, as such, do not add to the false rates. Additionally, since the network is permitted to extrapolate the state in unknown areas, all predictions deviating from the unknown class given unknown labels should also not be treated as false.

3.3.9 Methodology to use deep ISMs in Occupancy Mapping

In order to investigate H2, an alternative to the standard combination rules for evidential occupancy mapping will be proposed in this section. This altered combination has been proposed by the author in [BAU20].

The problem addressed in H2 revolves around accumulation of redundant information over time. This directly leads to the question of how to measure the information content in deep ISMs in the first place, as formulated in RQ10. As stated in Section 3.2.3, the commonly used entropy and specificity measure cannot be used to quantify the information in evidential occupancy mapping. However, since the deep ISMs in this work are constructed in a way to suffice R1.7, the unknown mass should correlate with the information content. This assumption is verified in the experiments discussed in sections 4.3 and 4.5.

Given the unknown mass as a measure for information, the problem of how to quantify the temporal redundancy shall be tackled (RQ8). In this work, the redundancy shall be assessed through mutual information, for reasons explained in Section 3.2.3. Alternatives are briefly discussed in Section 5.2.3. To obtain the mutual information, this work proposes to use temporally accumulated measurement signals as inputs for the deep ISMs. In doing so, the deep ISM learns to directly approximate the occupancy state $\boldsymbol{m}_{0:T}$ given the information $I_{0:T}$ of time step zero up to the current step T. On the other hand, before fusion, the map provides a successively obtained estimate of

the environment state $\boldsymbol{m}_{0:T-1}$ of all previous measurements. Ideally, this means that the deep ISM prediction contains all of the information stored in the occupancy map around the current ego vehicle's location with the addition of the information captured in the current time step. Thus, using the unknown mass as an inverse measure for information, the non-redundant part of the information can be approximated as follows

$$I^{T|0:T} = I^{0:T} - I^{0:T-1} \approx \underbrace{{m_u}^{0:T-1} - {m_u}^{0:T}}_{\equiv \Delta m_u} \qquad \text{Eq. 3-14}$$

However, a problem arises in case the static environment assumption of occupancy mapping is violated. When solely using Δm_u as a measure for information, a shift between free and occupied can only be achieved if the new signal is more certain. To allow for class shifts even if measurement signals are equally certain, it is proposed to add the conflict K as defined in Eq. 2-17 to the difference in unknown mass leading to the following improved approximation of novel information

$$I^{T|0:T} \approx \Delta m_u + K \in [-1, 1] \qquad \text{Eq. 3-15}$$

In case a new measurement contains conflicting information K but is also less certain than the current map state, Δm_u will become negative and reduce the amount of novelty in the update. This effect is desired to reduce the influence of outliers. The qualitative evaluation in the bottom plot of Fig. 5-1 in Section 5.1 shows the approximated amount of novel information in several fusion constellations.

At this point, it should be mentioned that the map region around the vehicle could simply be updated by replacing the old map state with the new deep ISM estimate since it already predicts ${m_u}^{0:T}$. This, however, would remove all of the geo ISM's influences. Therefore, it is rather proposed to discount the deep ISM estimate according to the amount of non-redundant information and combine it with the current map state. To do so, RQ9 has to be answered to obtain a discount factor for the discount operation defined in Eq. 2-25. The requirements for the discount factor γ can be formulated as follows

$$\gamma = \begin{cases} 0, I^{T|0:T} \leq 0 \\ 0, \lim\limits_{I^{T|0:T} \to 0} \\ 1, \lim\limits_{I^{T|0:T} \to 1} \\ 1, I^{T|0:T} \geq 1 \end{cases} \qquad \text{Eq. 3-16}$$

It should be noted that, ideally, $I^{T|0:T} < 0$ never occurs since the next time step's estimate is based on at least the same amount of information as the previous one. Since the proposed approximation of $I_{T|0:T}$ in Eq. 3-15 lies within the interval $[-1, 1]$,

the behavior defined in Eq. 3-16 between 0 and 1 can e.g. be achieved in a linear way as follows

$$\gamma = \text{ReLU}(\Delta m_u + K) \qquad \text{Eq. 3-17}$$

However, alternatives like a scaled hyperbolic tangent non-linearity can also be applied to e.g. dampen the influence of highly certain and highly redundant estimates. Since no clear preference is given, this work will focus on the discount factor as defined in Eq. 3-17. Finally, the decision of which combination rule to chose to combine the deep ISM estimate after discounting the redundancy will be detailed in Section 3.3.10.

3.3.10 Methodology to use deep ISMs as Priors in Occupancy Mapping

In this section, the requirements R2.3 and 2.4 to initialize a map with the deep ISM while later converging to the geo ISM shall be tackled. To do so, it shall be first discussed how the prior information can be integrated into the map in a way that its influence is disabled after enough data has been collected (see RQ11). This procedure has been proposed by the author in [BAU20].

Normally, as explained in Section 2.1, the map's state is initially set to the prior e.g. obtained through previous mapping of the environment. In this work, however, the map shall be initialized using a deep ISM during mapping. A simple procedure to do so would be to initially set the state of all grid cells to unknown. Afterwards, if a grid cell is still in the initial state and falls in the deep ISM's field of view, replace the grid cell's value by the deep ISM's estimate and leave it untouched otherwise. However, since the deep ISM's estimates are noisy, potentially prone to errors and to account for the fact that the estimates might improve due to better measurement coverage at later time steps, the following three stepped procedure is proposed to filter these effects.

Start Phase
First, in the start phase, all cells shall be set to $m_u = 1$ as illustrated on the left side in Fig. 3-9.

Initialization Phase
Afterwards, in the initialization phase, both the geo and deep ISM estimates are integrated into the map. This phase lasts until a definable amount of measurement information has been collected. In this work, the information content is described using the unknown mass (see Section 3.3.9). Therefore, a threshold on the unknown mass $\underline{m_u}$ is introduced to quantify whether enough data has been collected in a parameterizable way. Since the deep ISM shall only be used to initialize the map, its estimates should be integrated in a way that the unknown mass never falls below $\underline{m_u}$. At the same time, in

case the true occupancy state changes, the combination should be conducted in a way to allow shifting mass between occupied and free while keeping or even recuperating unknown mass. The possibility to recuperate unknown mass is important since once the unknown mass has reached $\underline{m_u}$, the deep ISM's influence is suppressed given the redundancy removal is applied as proposed in Section 3.3.9. This potentially prohibits the state change to be fully successful. To achieve the above described properties, the following procedure is proposed.

First, restrict the certainty of the deep ISM estimates to $\underline{m_u}$ using the discounting operation as follows

$$\underline{\tilde{\boldsymbol{m}}}^{0:T} = (1 - \underline{m_u}) \otimes \tilde{\boldsymbol{m}}^{0:T} \quad \text{Eq. 3-18}$$

with $\tilde{\boldsymbol{m}}^{0:T}$ being the evidential mass estimate of one grid cell provided by the deep ISM given accumulated measurement information from timestep zero till the current timestep T. This restriction of certainty, however, does not influence the amount of redundancy in the deep ISM estimates. Thus, the next step consists in removing the redundancy as proposed in Section 3.3.9 by an additional discounting operation. The restricted deep ISM certainty together with the discounting of non-redundant information makes sure that once a map cell's unknown mass has fallen beneath $\underline{m_u}$, all further deep ISM estimates are ignored. However, it is still possible for the deep ISM to reduce the unknown mass below the lower limit in the combination step as long as it has not been reached. Therefore, the discount factor to remove the redundancy has to be adapted to suffice the following condition

$$\boldsymbol{m}^{0:T} = \boldsymbol{m}^{0:T-1} \oplus \underbrace{(\gamma \otimes \underline{\tilde{\boldsymbol{m}}}^{0:T})}_{\approx \tilde{\boldsymbol{m}}^{T|0:T}} \quad \text{Eq. 3-19}$$

$$\underline{m_u} \leq {m_u}^{0:T} \quad \text{Eq. 3-20}$$

with $\boldsymbol{m}^{0:T-1}$ and $\boldsymbol{m}^{0:T}$ being the occupancy map's state accumulated from timestep zero to timestep T and $T-1$ respectively and $\tilde{\boldsymbol{m}}^{T}$ being the redundancy-removed evidential mass estimate at timestep T given the former timesteps information. Here, the adaption of γ, defined in Eq. 3-17, in a way that the fusion respects the lower bound $\underline{m_u}$, depends on the choice of combination rule. Since the combination requires the possibility to recuperate unknown mass, Yager's rule is chosen over Dempster's (for details on the properties of evidential combination rules see Section 2.4.1 and Section 5.1). Given Yager's combination rule, γ shall be chosen as follows

$$\underline{m_u} \leq {m_u}^{0:T} \underbrace{=}_{\text{Eq. 2-24}} {m_u}^{0:T-1} {\tilde{m}_u}^{T} + K \quad \text{Eq. 3-21}$$

Eq. 3-22

Including the definition of K and the discount operation, it follows that

$$\underline{m}_u \underbrace{\leq}_{\text{Eq. 2-25 \& Eq. 2-17}} {m_u}^{0:T-1}(1-\gamma+\gamma\tilde{m}_u^{0:T}) + {m_o}^{0:T-1}\gamma\tilde{m}_f^{0:T} + {m_f}^{0:T-1}\gamma\tilde{m}_o^{0:T} \quad \text{Eq. 3-23}$$

$$\underline{m}_u \leq {m_u}^{0:T-1} + \gamma(\underbrace{{m_u}^{0:T-1}\tilde{m}_u^{0:T} - {m_u}^{0:T-1} + {m_o}^{0:T-1}\tilde{m}_f^{0:T} + {m_f}^{0:T-1}\tilde{m}_o^{0:T}}_{=:\zeta}) \quad | - {m_u}^{0:T-1} \quad \text{Eq. 3-24}$$

$$\gamma\zeta \geq \underline{m}_u - {m_u}^{0:T-1} | \div \zeta \quad \text{Eq. 3-25}$$

Since the above derivation is made for the case of combination in the initialization phase, both the map cell's and deep ISM estimate's unknown masses are in the interval ${m_u}^{0:T-1}, \quad \tilde{m}_u^{0:T} \in [\underline{m}_u, 1]$. Thus, for $\zeta \equiv 0$ Eq. 3-20 is satisfied.
In case $\zeta \neq 0$, the two following solutions can be found

$$\text{case: } \zeta > 0 \rightarrow \gamma \geq \frac{\underline{m}_u - {m_u}^{0:T-1}}{\zeta} \quad \text{Eq. 3-26}$$

$$\text{case: } \zeta < 0 \rightarrow \gamma \leq \frac{\underline{m}_u - {m_u}^{0:T-1}}{\zeta} \quad \text{Eq. 3-27}$$

Here, the case of $\zeta > 0$, can be further simplified. Since $\underline{m}_u - {m_u}^{0:T-1} \leq 0$ and $\gamma \in [0, 1]$, γ already fulfills the requirement and, thus, does not need to be adapted. Therefore, the final discount factor $\underline{\gamma}$ that both reduce the informational redundancy in the deep ISM estimate and, at the same time, respects a lower bound on the unknown mass $\underline{m}_u$ after combination using Yager's rule can be written as

$$\text{case: } \zeta > 0 \rightarrow \underline{\gamma} = \gamma \quad \text{Eq. 3-28}$$

$$\text{case: } \zeta < 0 \rightarrow \underline{\gamma} = \min\left(\gamma, \frac{\underline{m}_u - {m_u}^{0:T-1}}{\zeta}\right) \quad \text{Eq. 3-29}$$

This modified Yager rule shall be referred to as lower bounded Yager rule. To fuse the geo ISM estimates into the map during the initialization phase, it is again proposed to use Yager's combination rule. The reason is the same as for the deep ISM fusion and revolves around the fact that Yager's rule is better suited to cope with changes in the true occupancy state.

In the center of Fig. 3-9, the main properties of the initialization phase to alter the map's state up the the lower bound $\underline{m}_u$ are depicted. It also shows the capability to move mass between free and occupied while recuperating unknown mass. Furthermore, a summary of the combination rule's desired properties both for the deep and geo ISM in the initialization phase are listed in Fig. 3-10.

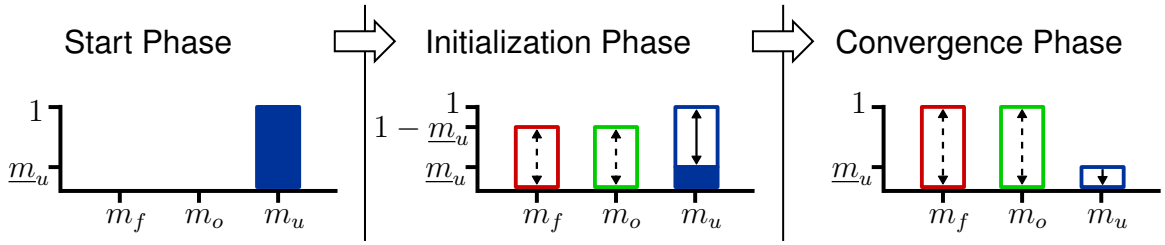


Fig. 3-9: Illustration of the three phases of a grid cell's occupancy state. Beginning with the start phase where the evidential mass is set to be all unknown. Afterwards, in the initialization phase, the unknown mass can be shifted into the free and occupied class and be recuperated using the deep and geo ISM's predictions. Finally, in the convergence phase, the geo ISM's estimates are used to strictly reduce the unknown mass while the deep ISM is being disabled.

Convergence Phase

Once the unknown mass has fallen below the threshold $\underline{m}_u$, the convergence phase starts. Here, the geo ISM should still be integrated into the map while the influence of the deep ISM should be disabled, guaranteeing a convergence towards the geo ISM. The combination rule for the deep ISM as defined for the initialization phase already makes sure that the influence of the deep ISM is being disabled in the convergence phase.

For the geo ISM, a combination rule has to be chosen that integrates the estimates in a way to strictly reduce the unknown mass but at the same time is capable to shift mass between the free and occupied class to, again, account for changes in the occupancy state or for correction purposes. To suffice these requirements, the usage of an adapted Yager's rule is proposed. The adaptation seeks to disable the capability to recuperate unknown mass to ensure that the unknown mass always remains below $\underline{m}_u$. To do so, it is proposed to assign the conflict K in equal portions to the free and occupied mass instead of assigning it to the unknown mass. This can be written as follows

$$\boldsymbol{m}_1 \oplus_{YD} \boldsymbol{m}_2 = \begin{bmatrix} m_{f1}m_{f2} + m_{f1}m_{u2} + m_{u1}m_{f2} + K/2 \\ m_{o1}m_{o2} + m_{o1}m_{u2} + m_{u1}m_{o2} + K/2 \\ m_{u1}m_{u2} \end{bmatrix} \quad \text{Eq. 3-30}$$

This combination rule shall be referred to as **YaDer**'s rule since it combines the properties of Yager's rule to model conflicting mass and of Dempster's rule to strictly reduce unknown mass.

An overview of the desired combination rule properties in each phase is provided in Fig. 3-10. For a qualitative evaluation of the combination approaches for each phase

together with a comparison against the two baseline combination rules, please be referred to Section 5.1. Furthermore, Section 5.4 shows the experimental results of applying this procedure for occupancy mapping on real world data.

Phase	Model	No Conflict	Conflict
init.	geo ISM	• fully converge • strictly reduce m_u	• fully converge • recuperate m_u
	deep ISM	• converge up to $\Delta m_u = 0$ • strictly reduce m_u	• converge up to $\Delta m_u = 0$ • recuperate m_u
converg.	geo ISM	• fully converge • strictly reduce m_u	• fully converge • strictly reduce m_u
	deep ISM	• unchanged	• unchanged

Fig. 3-10: Overview of the desired combination rules' properties in each mapping phase.

4 Deep ISM Experiments

As explained in Section 3.2.4, the experiments in this work will be conducted based on the NuScenes dataset. To obtain denser measurements for occupancy mapping, the sweeps are used to create the occupancy mapping dataset. Here, the sensor modality with the fewest sweeps per scene is identified and chosen as reference. Next, the temporally closest sweeps of the remaining sensors towards the reference are processed. Afterwards, sensor-dependent procedures are applied to create the different baselines, inputs and targets for the investigated geo and deep ISMs in form of a 128×128 grid map centered around the hind axle of the ego vehicle and spanning an area of 40×40 m^2.

4.1 Parameterization of geo ILM and IRM

This section details the comparison of different approaches to adapt lidar to radar BEV information. First, the sensor characteristics in the chosen dataset will be listed together with the applied methods to adapt the lidar. Afterwards, the different lidar filtering approaches will be evaluated based on the overlap between the lidar and radar maps in the mapped areas quantified by the mIoU score.

4.1.1 Experimental Setup

In the following, evidential occupancy maps of scenes as defined in the NuScenes dataset are created based on the geo IRM and ILM as defined in Section 3.3.2 and 3.3.3. Because of the large space of parameter configurations, the search is only conducted on the first 10% of the available training scenes of Fig. 1-1. To fix the parameters and identify the best performing ISM variant, a two stepped approach is proposed.

The first steps consists of estimating the best parameters for the geo ILM and IRM. Since no additional occupancy ground truth is available to tune both the lidar and radar models, a temporary ILM is manually configured by the author to provide a reference. This reference is further used to perform a parameter grid search for each of the geo IRM variants mentioned in Section 3.3.4. Here, the parameters, if not fixed for the given variant, are searched as shown in Fig. 4-1. These ranges are based on prior experience of the author. The temporary geo ILM uses the height-threshold-based ground-plane removal since it is the most widely used method in the literature.

$M_F{}^{\text{(thin)}}$	$M_F{}^{\text{(big)}}$	$\varphi_{\sphericalangle}{}^{\text{(thin)}}$	$\varphi_{\sphericalangle}{}^{\text{(big)}}$
$[0.1, 0.2, ..., 0.6]$	$[0.3, 0.4, ..., 0.8]$	$[1°, 1°, ..., 5°]$	$\{10°, 20°, 30°, 40°\}$

M_O	M_D	T
$[0.3, 0.4, ..., 0.8]$	$[0.1, 0.2, ..., 0.5]$	$\{1, 5, 10, 15, 20, 25\}$

Fig. 4-1: Ranges for geo IRM variants' parameter grid search given the parameters are not fixed for the respective variant.

In the second step, the afore best performing geo IRM variant will be kept fixed and used as a reference to analyze the ground-plane removal variants. Here, the two filters under investigation are a purely geometric height threshold-based filter and a semantic filter. The height thresholds are compared for different heights in the interval $[0, 0.1]$ with step size 0.025 and in the interval $[0.1, 2.0]$ with step size 0.1 in order to have a higher resolution close to zero. For the semantic filters, in the majority of cases the street detections are closest to the ego vehicle in the BEV projection followed by sidewalks and terrain. Since the removal of detections in occluded areas has little to no effect for geo ISMs, a three stepped removal of the semantics is proposed which removes pixels of classes on average increasingly further away from the vehicle. The three removal steps are as follows [no street; no street or sidewalk; no street, sidewalk or terrain].

In order to obtain lidar occupancy maps closer resembling the ground-truth occupancy state, the dynamic state provided by the bounding box labels is propagated to corresponding lidar detections. All detections marked as dynamic are only used to provide boundaries for free space rays but not to define occupied space. Since the sweep information is used for mapping but the labels are only available on a sample level, the bounding box poses of dynamic objects have to be interpolated as described in 3.3.1.

4.1.2 Experimental Results for geo ILM and IRM Parameter Tuning

In order to parameterize the temporary geo ILM, the height-threshold will be defined first. Here, the parameter is increased up to the point where additional structure besides street is being removed. Using the so found height threshold of 0.6 cm, the free and occupied weight M_F, M_O, M_D and the opening angle $\varphi_{\sphericalangle}$ of Alg. 1 are balanced in a way that

- remaining ground points and dynamic object artifacts are being filtered out,
- IDM rays don't cut through object boundaries,
- object boundaries are as dense as possible,
- free and occupied space has maximal assigned evidential mass

leading to the parameters summarized in Tab. 4-4. For an illustration of the ILM tuning, refer to Fig. 4-2.

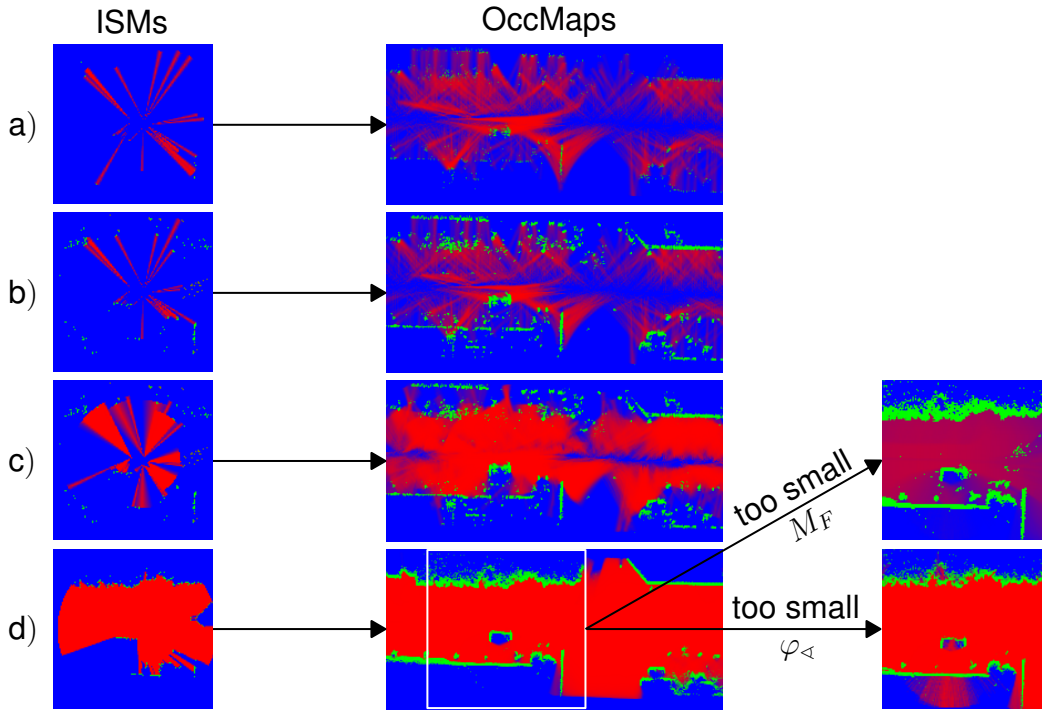


Fig. 4-2: Examples of the chosen ISM parameterization. The first column shows the ISM results and the second one the resulting occupancy maps. The rows show the different investigated ISM variants. Row a) shows the rays casting IRM which is extended in b) with accumulated detections and further extended in c) with additional free space rays. Row d) shows the geo ILM. Additionally, examples of the ILM with sub-optimal free mass M_F and ray opening angle $\varphi_\triangleleft$ are shown on the right.

After fixing the temporary ILM, it can be used to tune the IRM using the mIoU between their occupancy maps. As mentioned in Section 3.3.3, three IRM variants are separately tuned and compared against each other. As shown in Tab. 4-3, the accumulation of detections results in additional information about occupied space and helps to correct the free space leading to a better overall mIoU. Additionally, enriching the free space with the larger IDM rays keeps the occupied score untouched while moving unknown mass to the free class, leading to the best mIoU score of the considered IRM variants. These variants are illustrated in Fig. 4-2 together with resulting occupancy maps. The geo ILM defines a pseudo ground-truth that is then used for tuning the geo IRM and alter to evaluate the different learned IRM designs.

IRM variants	mIoU free	occupied	unknown	overall
ray casting	67.0	16.8	21.5	35.1
ray casting + acc. detections	70.6	26.6	21.5	39.9
ray casting + acc. detections + free rays	76.2	26.6	22.6	42.1

Fig. 4-3: Comparison of mIoU of occupancy maps generated using three IRM variants and lidar occupancy maps.

ISM	$M_F{}^{(\text{thin})}$	M_O	M_D	$\varphi_{\sphericalangle}{}^{(\text{thin})}$	$M_F{}^{(\text{big})}$	$\varphi_{\sphericalangle}{}^{(\text{big})}$	T
ILM	0.025	0.5	0.3	3°	0	0	
IRM #1	0.1	0.8	0.3	5°	0	0	1
IRM #2	0.2	0.3	0.3	5°	0	0	20
IRM #3	0.2	0.3	0.3	5°	0.2	30°	20

Fig. 4-4: Parameters used for geo ILM and IRM (see Section 3.3.4) to produce the qualitative and quantitative results in Fig. 4-2 and 4-3. The parameters are chosen via grid search.

4.1.3 Experimental Results for Lidar Ground Plane Removal Variants

The quantitative comparison in Fig. 4-6 shows that the successive semantic-based removal up to the terrain level leads to increasingly better overlap up to the best reached mIoU score of 28.18%. On the other hand, the height threshold-based filters show improved performance up to a height threshold of 0.5 m with a score of 26.96% after which the performance starts to decrease. This suggests that for the street and sidewalk level semantics as well as low height threshold large portions of the areas detected by the radar are occluded in the lidar BEV. This is also qualitatively shown in the lower left white boxes in Fig. 4-5. Moreover, when the height threshold is set too high, portions of the areas detected by the radar are increasingly filtered out, as illustrated in the upper right boxes in Fig. 4-5. Thus, a height threshold of about 0.5 m or the semantic-based removal up to the terrain level provide the best compromise of the compared methods. However, since the semantic information is only available for keyframes in the NuScenes dataset and because the 0.5 m height threshold filter rivals the best semantic filter in its performance, it is proposed to use the height threshold-based filter to obtain the labels for further experiments.

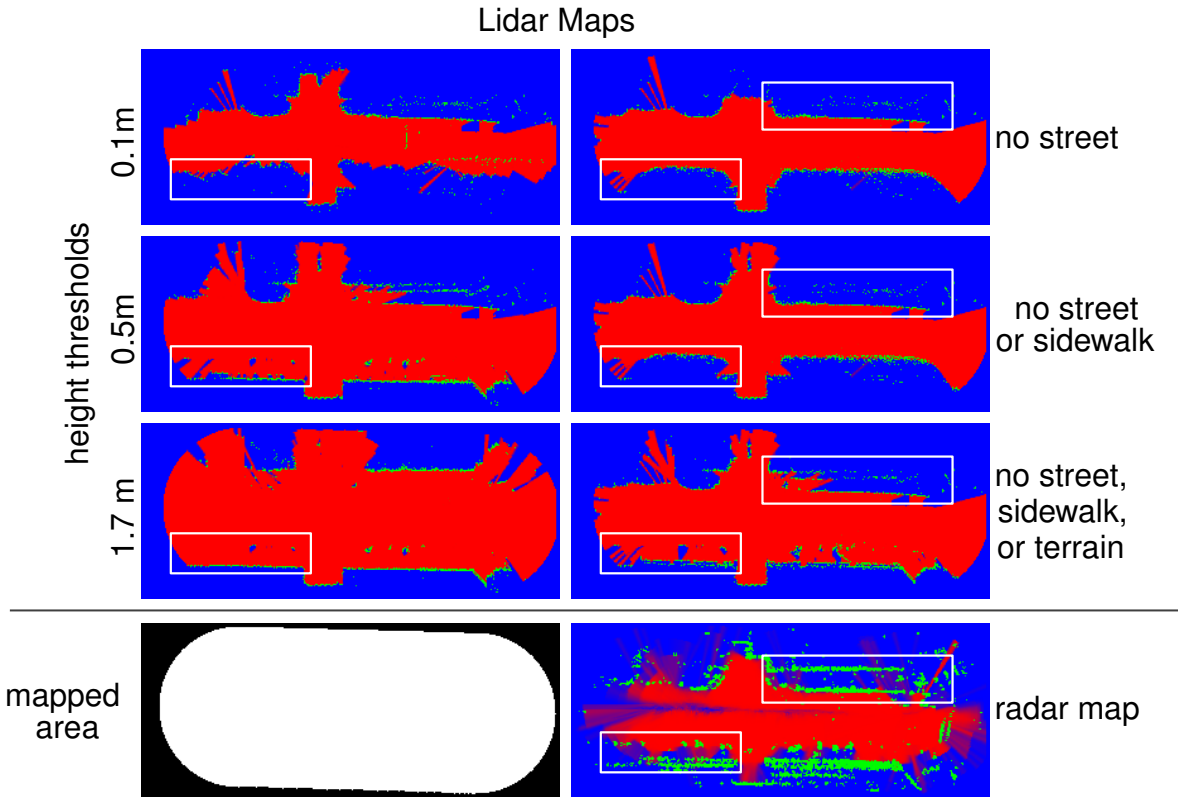


Fig. 4-5: Example of lidar maps created by successively removing ground-plane semantics and three threshold-based filters, together with the mapped area (white) and the radar map.

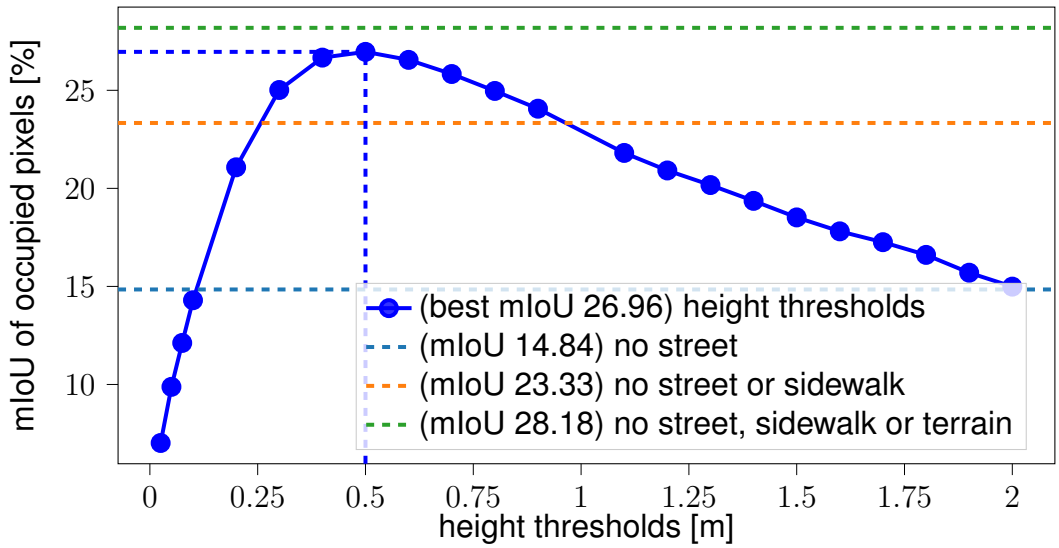


Fig. 4-6: Results of mIoU between different evidential occupancy maps create with variations of geo ILMs and the geo IRM computed in the mapped area.

4.1.4 Discussion

Looking at the results of the geo IRM variant comparison, it becomes clear that the two proposed improvements, namely the accumulation of detections over time and the additional application of the IDM with a wide cone, indeed outperform the baseline by a margin of 7% in mIoU. However, there have been cases in which the accumulation leads to a persistence of outliers increasing their influence. But, those cases are statistically outweighed by the benefit of restricting rays from cutting through occupied regions. It shall also be mentioned that there are still large areas marked as unknown which are clearly free. One noticeable example is the vehicle's trajectory which remains untouched by the geo IRM. Here, the geo IRM can be changed to set the region beneath the ego vehicle to free. For the other regions further away from detections and ego vehicle further investigations can be done to improve the free space coverage. Overall, this procedure has shown to suffice RQ2, since it strictly improves the mIoU scores of all classes over the baseline model.

With regards to the analysis of ground-plane removal techniques according to RQ1, the comparison of lidar ground plane removal methods demonstrates that the removal of detections up to the terrain level increases the overlap between lidar and radar sensing modalities and provides the best result. Additionally, it can be seen that the application of the simple height threshold removal method can reach similar performance when it comes to aligning the sensing overlap. Thus, since the lidar labels are only provided with low frequency on the sample level and an interpolation down to the higher frequent sweep level is non-trivial, it is proposed to use the height threshold method for the further experiments.

4.2 Choice of UNet Architecture

This section details the experimental determination of hyperparameters of the deep ISM's base architecture, as defined in Section 3.3.6.

4.2.1 Experimental Setup

Since the focus of this work lies on the investigation of radar ISMs, the architecture search is performed based on radar input images with occupancy map patches for targets (see Section 3.3.5). More specifically, radar BEV images based on one sweep's information R_1 are used, since they contain the least information and, thus, provide the most difficult task for the network to handle. Moreover, the considered UNet (see Section 3.3.6) is trained in SoftNet configuration (see Section 3.3.7), since it is the baseline configuration as proposed by the literature.

The hyperparameters searched with the following procedure are the downsample fac-

tor D of each ResNet layer in the UNet and the amount of filters for each stage $C_k, k \in [0, 4]$. With regards to the amount of channels per layer, the approach proposed in the literature to double the amount after each encoder and halve it after each decoder stage respectively up to a maximum number of channels is adapted in this work (see Section 2.3.1). Thus, reducing the search space for the number of channels to two hyperparameters, namely the initial number of filters C_0 and the maximum number of filters $C_{\max}$. These hyperparameters shall be investigated in a two stepped approach. First, D is set to 0.5 which is half of the most conservative compression rate reported to work without loss in performance (see Section 2.3.1). On the other hand, for C_0, the following variations are evaluated $[4, 8, 16, 24, 32, 40]$ with $C_{\max} = \infty$. Given the results of these variations, a configuration with a good trade-off between inference speed and accuracy is chosen for further optimization. Here, based on personal experience and the architectures reported in the literature, $C_{\max}$ is set to 128 filters. Additionally, D is successively halved starting from 0.25, which is still reported throughout the literature to work without significant loss of accuracy, up to the point of collapse in performance.

All these experiments are conducted using Tensorflow 2.1 [ABA16], the ADAM optimizer [KIN14] with a learning rate of 0.001 and a dropout rate of 0.3. The layers are initialized using the HeNormal initializer [HE15] and trained on the scenes in Fig. 1-1 and 1-2 until convergence, as indicated by the validation set (see Fig. 1-3), to remove the bias due to the random initialization. The experiments are conducted on two NVIDIA Tesla V100 (32 GB) GPUs and evaluated on a single core of an Intel Core Processor i7-10750H CPU. It shall be mentioned that the test set (see Fig. 1-4) is not utilized during the network tuning process.

4.2.2 Experimental Results

As explained above, the architecture search is split into two parts. Here, the first part fixes the ResNet layer's downsample rate D and alternates the initial number of filters C_0 with $C_{\max} = \infty$. These hyperparameter configurations are summarized in the experiments $1 - 6$ in Fig. 4-7 while the results are shown in Fig. 4-9 marked in orange. The experiments show an exponential decrease in inference time to improve the mIoU. With regards to the time till convergence of training, the time remains about the same up to $C_0 = 24$. Afterwards, for $C_0 = 32$, it more than triples and doubles again for $C_0 = 40$.

Based on these results, $C_0 = 32$ is chosen and further used to fine tune D for the following reasons. First, further increasing C_0 shows to move the inference speed too far away from the goal of 100 Hz as defined in R1.4. However, increasing C_0 from 24 to

32 showed a significant improvement in qualitative predictions as can be seen in Fig. 4-8.

experiment number	C_0	$C_{\max}$	D
1	4	∞	0.5
2	8	∞	0.5
3	16	∞	0.5
4	24	∞	0.5
5	32	∞	0.5
6	40	∞	0.5
7	32	128	0.25
8	32	128	0.125

Fig. 4-7: Hyperparameter setting for all network tuning experiments

To fine tune the capacity of the network, C_0 is fixed to 32 while an upper limit on the number of filters is set to $C_{\max} = 128$. Given this setup, D is halved, starting from 0.5 up to the point where the performance collapses. It can be seen that this is the case after the second reduction, as shown by the blue dots in the left plot of Fig. 4-9. The parameter settings for the two experiments labeled "7" and "8" are shown in the lower part of 4-7.

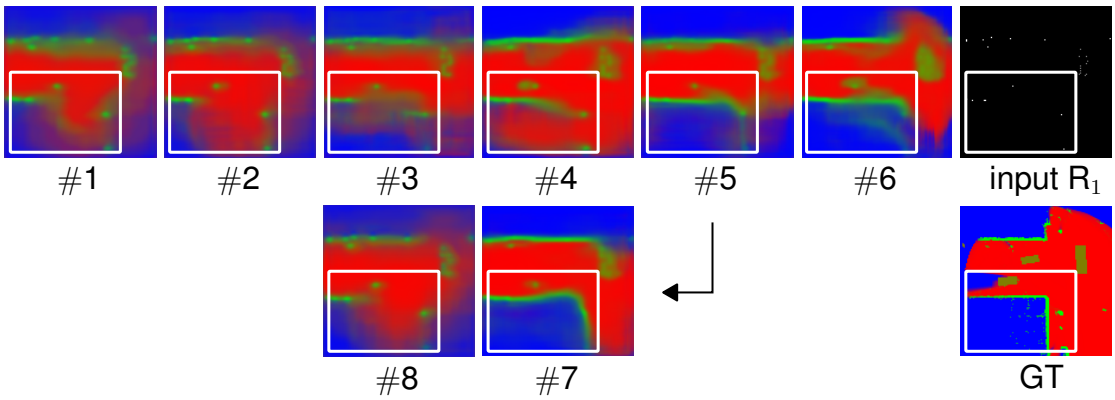


Fig. 4-8: Qualitative results of the different models trained in the experiments as numbered in Fig. 4-7 with the radar input R_1 and the ground-truth GT.

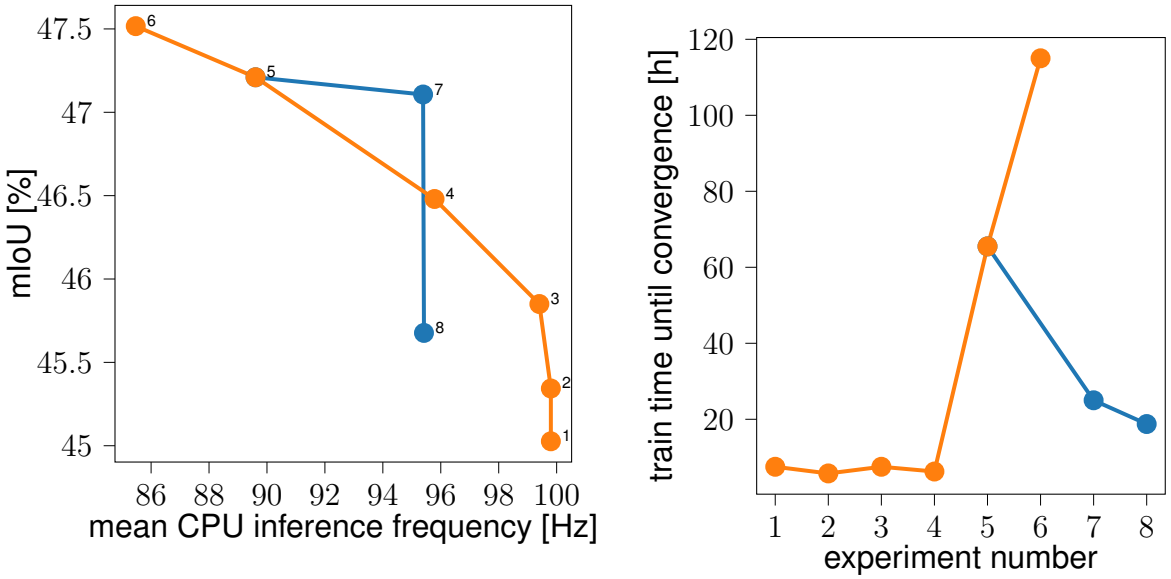


Fig. 4-9: Experimental results of the network tuning experiments. On the left-hand-side, the mIoU over the CPU inference frequency is shown while on the right-hand-side, the training time is plotted for each experiment. Here, orange marks the first part of the experiments in which the downsamlping rate D is fixed and the initial amount of filters C_0 is searched. On the other hand, blue marks the second stage of the parameter search in which D is finetuned.

The experimental results show that a reduction of D from 0.5 to 0.25 brings the network about 6% closer to the goal of 100 Hz while roughly keeping its performance. Also, the interpolation capability seems to deteriorate only after the second reduction step, as shown qualitatively in Fig. 4-8. At the same time, the first reduction more than halves the training time.

4.2.3 Discussion

First of all, the choice of only conducting each experiment once shall be discussed. It shall be argued that this choice is justified since the trend of the experiments clearly follows the expected and reported behavior in the literature for experiments $\#1$ - $\#7$. Moreover, all models have been trained until convergence, thereby reducing the influence of random initialization. Regarding experiment $\#8$, the severe drop in performance can be explained as the point for which the bottleneck starts to impede the flow of information. However, since the experiments have been conducted only once for each parameter setting, the effect might also be due to outlier behavior. In any case, since the mean inference frequency did not significantly increase over its predecessor, the configuration in $\#7$ is chosen and $\#8$ is not repeated.

The experiments to fine-tune the architecture for this work clearly verify the results as stated in the state-of-the-art. It can be seen that the UNet architecture is capable

of learning geometric interpolations (see interpolated corner in white box in Fig. 4-8). Also, the choice of network depth, and with it the network's receptive field, suffice to learn the concept of occluded areas. This can be seen in Fig. 4-8 by the unknown areas (blue) behind occluded areas (green) given the reference point is the image center. Therefore, the model can clearly utilize the spatial coherence in huge amounts of data and, thus, suffices R1.1, 1.2 and 1.6.

Moreover, it has been shown that the reduction in the ResNet layer's bottleneck can be done up to factor of 0.25 while almost keeping the performance, which validates the results reported in the state-of-the-art. A further reduction leads to serious performance degradation (might be due to outlier behavior). Here, the variant $\#7$ is still able to predict geometric interpolations while providing an inference speed close enough to suffice R1.4. While the performed architecture search used many assumptions to reduce the search space and only performed the search on rough intervals, clear trends can be shown leading to $\#7$ as the best performing model according the the given requirements. Thus, it can be argued that this procedure suffices to answer RQ3.

4.3 Aleatoric Uncertainties in deep ISMs

This section describes the experiments to analyze to which extent the different deep ISM variants, described in Section 3.3.7, are capable to capture aleatoric uncertainties.

4.3.1 Experimental Setup

To investigate how to best learn and shift the aleatoric uncertainty as described in RQ7, the basic UNet architecture as tuned in Section 4.2 is modified and trained on the training scenes listed in Fig. 1-1 and 1-2 in the SoftNet, ShiftNet and DirNet configuration as described in Section 3.3.7. These configurations are trained using the radar inputs R_1 and R_{20} with occupancy map targets GT as described in Section 3.3.5 until convergence as indicated by the validation set (see Fig. 1-3). The comparison of the deep IR_1M and $IR_{20}M$ results should help to investigate the effect of data uncertainty. Additionally, the geo IRM's performance for both radar inputs is evaluated to provide a reference. The evaluation is based on the confusion matrix variant as explained in Section 3.3.8 and computed for the whole test set defined in NuScenes (see Fig. 1-4).

4.3.2 Experimental Results

For the following interpretation of quantitative results, it shall be noted that unknown mass in other classes is not seen as false predictions but rather as an indicator for certainty. Thus, the overall false rate only equals the sum over the red scores per row for each class.

Starting with the R_1 scores, the usage of the modified confusion matrix, as opposed to the mIoU used in 4.1, allows a more detailed analysis of the geo IRM. It can be seen that the geo IRM provides less then 1% correct predictions both for dynamic and occupied space due to the sparseness in detections, which can also be seen in Fig. 4-11. On top of that, the false free space predictions in these two categories are almost 20% showing the effect of rays cutting through occupied space due to multi-path reflections. On the other hand, while the free space predictions are sparse with about 70% of the space being predicted as unknown, the rest is predicted correctly. In an overall comparison with the deep IRM variants, the geo IRM provides the least false rates in all categories but also the smallest positive rates. This experimentally shows the reason for the approach to use a deep IRM to initialize the maps and later converge to the geo IRM.

Next, the SoftNet configuration treats the aleatoric uncertainty by equally distributing mass into the two classes between which the uncertainty occurs, as stated in H 1. In this case, the majority of uncertainty in the visible area is at the boundaries between occupied and free areas leading to huge portions of the actually free and occupied class respectively being estimated as dynamic. This effect can be seen in Fig. 4-11 by the blurriness of the occupied space. Due to this bias towards the dynamic class, the true rate for dynamic objects is also the best among the investigated methods. In occluded areas, in addition to huge false rates of dynamic objects, an increase of unknown mass can be observed over all classes. This might be due to a combination of bias towards the unknown class in occluded areas and the fact, that the aleatoric uncertainty now also occurs at boundaries between free, occupied and the unknown class.

In contrast to SoftNet, DirNet is capable of shifting the aleatoric uncertainty into the unknown class which can be seen by less than half of the overall false dynamic predictions in the free and occupied categories while at the same time increasing the unknown mass portion over all classes. Moreover, in the occluded area, where more aleatoric uncertainty can be expected, larger portions are shifted into the unknown class which also results in smaller false rates for free and occupied cells. These effects can also been seen in Fig. 4-11 by the clearer boundaries between free and occupied space and overall more unknown space in occluded regions. This, in contrast to the almost steady false rates in the visible and occluded area for SoftNet, additionally highlighting the uncertainty awareness of the DirNet configuration. On top of that, DirNet surpasses the positive rates of SoftNet in all but the biased dynamic class. The improvement in performance might be due to the effect that by modeling the aleatoric uncertainty, the loss function is weighted in a way to increasingly ignore predictions for which the network cannot find a sufficient solution. This leads to more network capacity

being focused on the majority of data which leads to better average performance and, thus, better scores. It shall be noted that, while the scores improve, this behavior might lead to a neglection of edge cases.

Finally, ShiftNet demonstrates even better capabilities in shifting the aleatoric uncertainties to the unknown class as compared to DirNet which is indicated by the highest unknown mass rates of all model variants. This uncertainty-awareness can again also be observed by higher unknown mass rates for the occluded as compared to the visible areas. The effect of shifted uncertainty is so dominant that it can even be seen in the qualitative results in Fig. 4-11 by unknown space (blue) around all occupied boundaries. Regarding the occupied and free space, ShiftNet provides the least false rates and best free space predictions for all classes compared to all learned IRMs. However, it lacks the capability to predict occupied areas with high certainty.

For the R_{20} scores, an expected overall improvement of all models can be observed. Here, the free space predictions of SoftNet even improve to the point of surpassing the other variants. With regards to aleatoric uncertainty, the unknown mass portions for the DirNet and ShiftNet decrease compared to the R_1 scores as is desired in the light of more input information. Additionally, similar to the R_1 scores, an increase in unknown mass can be observed in occluded as compared to visible areas. Another notable change is the improvement in occupied predictions given by ShiftNet, now surpassing SoftNet and almost closing the gap to DirNet.

	$\tilde{k}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$
geo IRM	$p(k\|d)$	0.6	19.6	0.4	79.3	0.6	34.3	0.4	64.5	0.6	12.6	0.4	86.3
	$p(\tilde{k}\|f)$	0.0	28.9	0.0	70.9	0.0	35.9	0.0	63.9	0.0	9.6	0.0	90.2
	$p(\tilde{k}\|o)$	0.2	18.4	0.8	80.5	0.3	30.6	0.9	68.1	0.2	14.5	0.8	84.4
	$p(\tilde{k}\|u)$	0.0	5.4	0.1	94.4	-	-	-	-	0.0	5.4	0.1	94.5
SoftNet	$p(k\|d)$	49.7	16.7	10.5	23.2	52.5	18.8	11.5	17.2	48.8	15.8	9.6	25.8
	$p(\tilde{k}\|f)$	35.8	37.0	3.3	24.0	36.2	42.6	3.1	18.2	35.5	21.5	4.0	39.1
	$p(\tilde{k}\|o)$	37.8	9.6	17.2	35.5	43.7	11.5	20.0	24.8	35.9	9.0	16.2	38.9
	$p(\tilde{k}\|u)$	25.7	8.7	4.5	61.0	-	-	-	-	25.7	8.7	4.5	61.1
DirNet	$p(k\|d)$	31.0	21.9	12.8	34.3	35.4	27.4	13.0	24.2	29.0	19.9	12.3	38.7
	$p(\tilde{k}\|f)$	13.6	47.5	5.6	33.3	15.5	56.8	4.6	23.2	9.0	22.3	8.4	60.3
	$p(\tilde{k}\|o)$	13.5	9.7	20.7	56.1	22.4	14.3	22.9	40.3	10.8	8.4	19.9	60.9
	$p(\tilde{k}\|u)$	2.7	3.7	12.0	81.6	-	-	-	-	2.7	3.6	12.0	81.7
ShiftNet	$p(k\|d)$	31.8	19.0	7.2	41.9	30.7	23.3	7.5	38.6	33.3	17.2	6.9	42.7
	$p(\tilde{k}\|f)$	7.4	48.6	2.0	42.0	7.5	56.0	1.6	34.9	7.6	28.1	3.2	61.1
	$p(\tilde{k}\|o)$	8.4	15.1	13.7	62.8	10.0	19.2	14.9	55.9	7.9	13.9	13.4	64.8
	$p(\tilde{k}\|u)$	4.0	9.4	4.7	81.9	-	-	-	-	4.0	9.3	4.7	82.0
R_1 Scores		overall				visible				occluded			

	$\tilde{k}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$
geo IRM	$p(k\|d)$	7.1	24.5	6.8	61.4	4.5	43.7	7.5	44.1	8.2	15.9	6.5	69.3
	$p(\tilde{k}\|f)$	0.4	47.0	0.7	51.7	0.4	57.8	0.5	41.1	0.6	17.5	1.2	80.6
	$p(\tilde{k}\|o)$	2.6	16.8	13.5	67.0	2.7	31.0	15.4	50.7	2.5	12.3	13.0	72.1
	$p(\tilde{k}\|u)$	0.6	3.6	1.7	94.1	-	-	-	-	0.6	3.5	1.7	94.1
SoftNet	$p(k\|d)$	52.4	21.8	12.0	13.7	54.6	25.2	12.7	7.6	51.9	20.6	11.3	16.2
	$p(\tilde{k}\|f)$	21.6	64.1	2.0	12.2	19.5	72.0	1.6	6.9	28.1	42.6	3.3	26.0
	$p(\tilde{k}\|o)$	36.2	12.9	22.8	28.1	43.2	15.3	25.8	15.6	33.9	12.2	21.9	32.0
	$p(\tilde{k}\|u)$	19.2	9.9	4.6	66.3	-	-	-	-	19.1	9.8	4.6	66.5
DirNet	$p(k\|d)$	37.1	22.7	17.4	22.8	39.6	29.2	18.3	12.8	36.2	20.2	16.5	27.1
	$p(\tilde{k}\|f)$	10.9	61.5	5.1	22.4	11.3	71.4	3.9	13.5	10.6	35.4	8.3	45.7
	$p(\tilde{k}\|o)$	15.2	10.5	31.8	42.6	25.0	15.1	35.2	24.6	12.0	9.1	30.7	48.2
	$p(\tilde{k}\|u)$	3.0	5.4	12.7	78.9	-	-	-	-	3.0	5.4	12.6	79.0
ShiftNet	$p(k\|d)$	38.5	18.1	12.7	30.7	35.3	22.9	13.7	28.1	40.9	15.9	12.0	31.2
	$p(\tilde{k}\|f)$	4.7	62.9	3.2	29.2	4.3	71.1	2.4	22.1	6.1	40.8	5.3	47.8
	$p(\tilde{k}\|o)$	6.0	13.6	28.5	51.8	7.4	17.6	30.9	44.1	5.7	12.4	27.8	54.1
	$p(\tilde{k}\|u)$	3.0	9.0	7.8	80.2	-	-	-	-	3.0	8.9	7.8	80.3
R_{20} Scores		overall				visible				occluded			

Fig. 4-10: Results of deep ISM variants trained on R_1 and R_{20} images for visible, occluded and overall areas. For a detailed explanation of the table format kindly refer to Section 3.3.8.

The geo IRM is also capable of improving the dynamic true rate by about seven times and the occupied rate by about 13 times while reducing the respective overall false rates. This is as expected since it can leverage the information of 20 times as many timesteps. Also, the free space true rate reaches about 50% while only minimally

increasing the false rates. These general improvements over the R_1-based predictions are also clearly visible in Fig. 4-11.

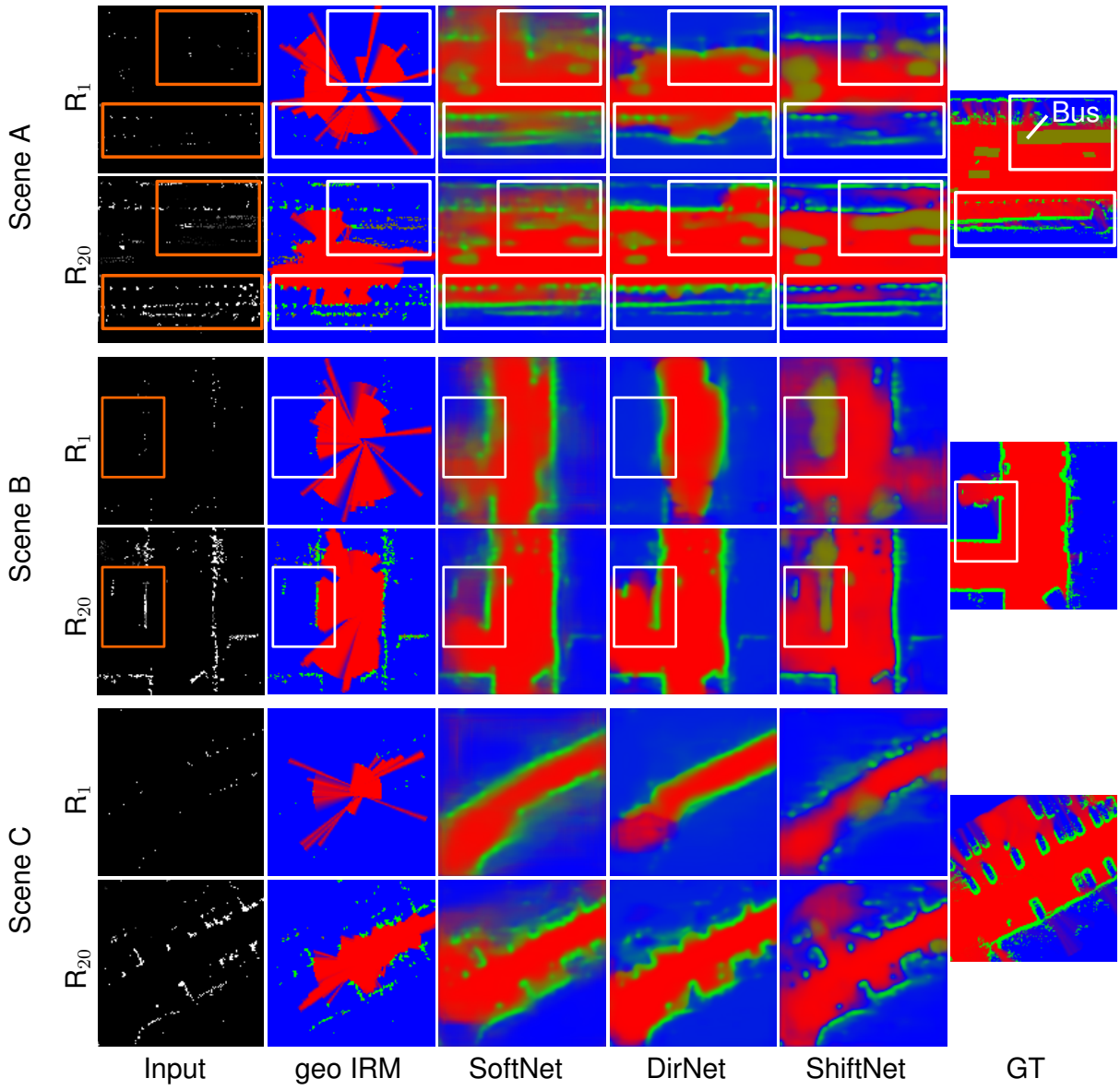


Fig. 4-11: Qualitative results of the different compared IRMs (column-wise) for three validation scenes showing the predictions based on R_1 and R_{20} in the respective rows with the corresponding label. For a detailed explanation of the abbreviations and color format kindly refer to Section 3.3.5.

4.3.3 Discussion

The experiments have shown that the current state-of-the-art deep ISM (SoftNet) indeed models occurring uncertainty as conflicting mass leading to a huge bias in the dynamic class (see Fig. 4-10), proving H1. The two compared methods to shift the uncertainty to the unknown class both clearly reduce the amount of bias in the dynamic

class. Additionally, the amount of shifted mass to the unknown class clearly correlates with the amount of certainty in the data which can be seen by comparing the unknown mass assigned in visible and occluded areas and also the overall assigned unknown mass between R_1 and R_{20} inputs, providing an answer to RQ7.

Since ShiftNet provides the least false rates per class over all classes for the R_{20} compared to all models while providing similar true positive rates compared to DirNet it largely suffices R1.7 and 1.8. Thus, ShiftNet will be used as the baseline method for further experiments and analysis.

Regarding the ISMs capability to estimate dynamic objects, the qualitative results in Fig. 4-11 Scene A shall be analyzed. These show that the R_1 inputs contain sometimes only a single detection belonging to a dynamic object. This leads the networks to predict a dynamic object prior with size and shape of a car oriented in the direction of the road (compare lower right dynamic object in upper marked box in input and ShiftNet prediction of Scene A in Fig. 4-11). For R_{20}, all of the dynamic object's detections over time are available leading to an overall improvement in dynamic class true rates. However, the false occupied predictions in the dynamic class are also consistently increased and in case of ShiftNet almost doubled. This might hint that the networks can decide that there is an object but are uncertain about its motion state. This problem might be due to the choice of encoding the dynamic detection or due to errors in the dynamic flags of the radar detections. Additionally, the approach to decay the dynamic detections over time to encode the motion direction seems to work for some vehicles (see ShiftNet prediction and label of car in lower left for Scene A) while other objects are elongated (see ShiftNet prediction bicycle in the lower right of the upper marked box in Scene A). Thus, further improvement needs to be done in the future to find better ways of encoding the dynamic detections over time.

The above analysis and experimental results provide a partial answer to RQ4, by analyzing only the deep ISM's capability to estimate evidential masses for radar inputs. The remaining analysis is provided in the following Section 4.4. Moreover, two of the three possible radar encodings have been analyzed providing a partial answer to RQ6. The analysis of the remaining variant is provided in Section 4.4 to provide the full answer. Regarding RQ10, it can be seen that the unknown mass, as proposed in Section 3.3.9 indeed correlates with the information content in the ShiftNet inputs. To further verify the validity to use the unknown estimated by the ShiftNet, the correlation is additionally analyzed for camera, lidar and the fused inputs in Section 4.5.4.

4.4 Analysis of dynamic Detection Encoding for R_{20} ShiftNet Inputs

In this section, the effect of encoding all static and dynamic radar detections of the temporal horizon is compared against encoding all static but only the most recent dynamic radar detections as deep IRM inputs.

4.4.1 Experimental Setup

The first two variants, namely the radar BEV projection of a single timestep R_1 and the temporally accumulated radar projection with decay on the dynamic detections' intensity R_{20} are already discussed in Section 4.3. Thus, this section focuses on encoding temporally accumulated radar detections while only marking the latest dynamic detections, abbreviated as $R_{20|1}$. The three variants are explained in more detail in Section 3.3.5. For the analysis, the ShiftNet deep ISM is trained on the scenes in Fig. 1-1 and 1-2 until convergence as marked by the validation set (see Fig. 1-3) defined in NuScenes. The qualitative and quantitative results are purely based on the test set (see Fig. 1-4).

4.4.2 Experimental Results

First, the scores shown in Fig. 4-10 for the inputs R_1 and R_{20} are compared with the scores for $R_{20|1}$, shown in Fig. 4-12. Here, deep ISMs trained on R_{20} show an overall increased performance for the dynamic class over $R_{20|1}$. This shows that the encoding of dynamic detections with decay is beneficial over only including the latest dynamic detection for moving object prediction. Qualitatively, the improvement of the estimated dynamic object shapes is clearly visible in the white box in scene A and the left white box in scene B of Fig. 4-13. More specifically, it is shown that the deep ISM is capable of producing reasonable shape, position and orientation predictions of the dynamic objects based on a single detection point and the scene layout for R_1 and $R_{20|1}$ compared to the cloud of dynamic detections for R_{20}. Therefore, the improvement for dynamic objects is to be expected. Nevertheless, accumulation of dynamic detections in R_{20} also leads to failure cases.

	$\tilde{k}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	
ShiftNet	$p(k\|d)$	29.2	21.7	14.4	34.6	27.8	26.3	15.4	30.6	31.0	20.3	13.2	35.5
	$p(\tilde{k}\|f)$	4.7	63.0	3.3	29.0	4.7	71.9	2.5	20.9	5.0	38.8	5.7	50.5
	$p(\tilde{k}\|o)$	4.5	14.1	30.0	51.4	5.7	18.4	32.4	43.5	4.2	12.8	29.3	53.6
	$p(\tilde{k}\|u)$	2.1	8.2	7.7	82.0	-	-	-	-	2.1	8.1	7.7	82.1
$R_{20\|1}$ Scores		overall				visible				occluded			

Fig. 4-12: Normed confusion matrix evaluated on ShiftNet model which was trained on $R_{20|1}$ inputs. For a detailed explanation of the table format kindly refer to Section 3.3.8.

Here, the right white box in scene B of Fig. 4-13 shows a predicted dynamic object for R_{20} which is already outside the ISM's FoV. This is caused by the trailing dynamic predictions which does not occur for the other deep IRM variants. On the other hand, the overall occupied class performance is slightly but consistently improved by only taking the latest dynamic detection for $R_{20|1}$. One potential cause can be seen in the lower orange box in scene B of Fig 4-13. Here, static detections are missing in R_{20} which are present in $R_{20|1}$. This is caused by some detections almost outside of the time horizon that have been falsely identified as dynamic and are being overlayed over the static predictions. Hence, the accumulation of dynamic detections can cause outliers to deteriorate static detections. Eventually, the free scores are mainly the same, showing that the changes only affect the occupied and dynamic class. Finally, comparing $R_{20|1}$ and R_{20} with R_1 it can be seen that the problem of decrease in dynamic prediction performance, indicated by the quantitative results while the qualitative results seem to be sharper, remains.

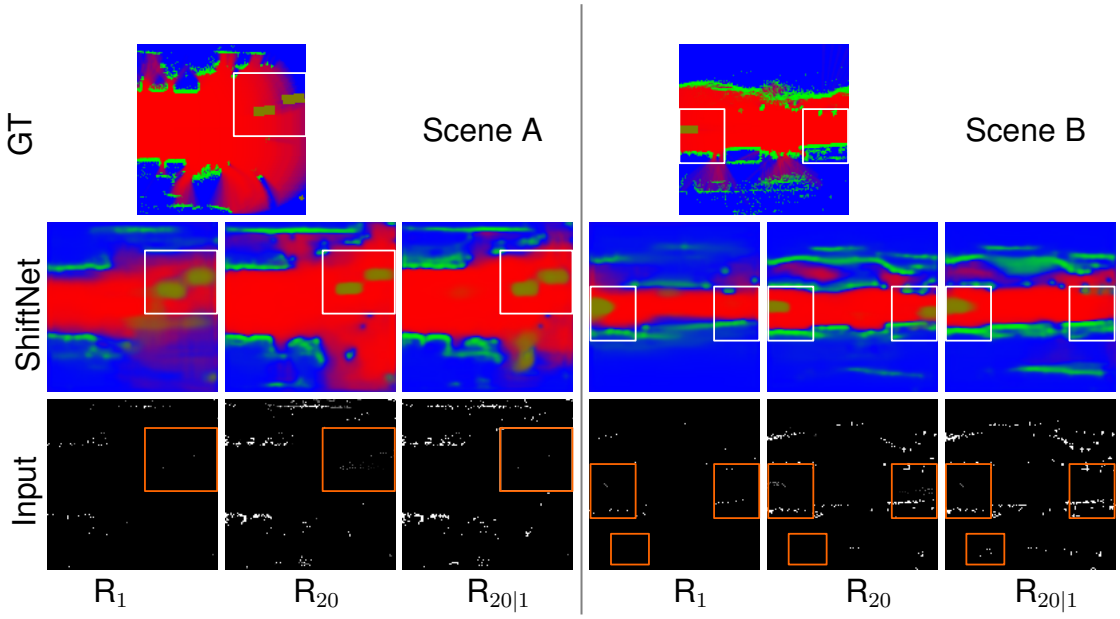


Fig. 4-13: Qualitative comparison of ShiftNet predictions (middle) trained on radar inputs with three different dynamic detection encodings (bottom) shown for two scenes with the respective GT (top). For a detailed explanation of the abbreviations and color format kindly refer to Section 3.3.5.

4.4.3 Discussion

The above results show that the radar encoding $R_{20|1}$ significantly decreases the capability to estimate dynamic objects, slightly increases the occupied area predictions while maintaining about constant for free areas. Here, the decrease for the dynamic class is to be expected, since in some cases only a single detection point is assigned to a dynamic object. Therefore, the $R_{20|1}$, similar to the R_1, based deep ISMs need

to rely on scene understanding and prior knowledge from the dataset to estimate the dynamic object shapes.

Considering the occupied class, the qualitative results indicated one of the causes to be the increased influence of detections being falsely marked as dynamic. Through the temporal accumulation in R_{20}, the outliers influence can persist over time and cause static boundaries to deteriorate through the temporal decay. Also, the quantitative results both for $R_{20|1}$ and R_{20} show decreased quantitative performance over the R_1 inputs while the qualitative results seem to get sharper. This might be caused by overall more occupied mass being estimated that might leak into dynamic predictions. Here, deep ISM based on $R_{20|1}$ and R_{20} increase their occupied true rates by more than double the amount that is predicted for R_1 inputs.

Overall, the quantitative scores for R_{20} exceed the ones for $R_{20|1}$ and R_1 or are in a similar range. Also, the qualitative results show the arguably best edge accuracy for occupied and shape estimation for dynamic objects. Thus, the R_{20} encoding will be used for further comparisons and experiments, concluding the answer to RQ6.

4.5 Analysis of Camera, Lidar and Fused Inputs for deep ISMs

The focus of this section is the experimental comparison between deep ISMs trained with inputs from cameras, lidar and a respective fusion of these modalities with radar measurements.

4.5.1 Experimental Setup

To analyze the performance and occurring effects using a deep ICM and ILM, the homography projection into BEV of all camera images C_{RGB}, their semantics C_S, the MonoDepth BEV projection C_D and the lidar BEV projection L are considered as inputs. Details of the chosen inputs are described in Section 3.3.5. For reasons discussed in Section 4.3.3, the ShiftNet configuration is chosen for the experiments. Additionally, the fusion of camera and lidar respectively with radar inputs R_{20} shall be investigated. Here, only the monocular depth projection is used on the camera side for the experiments since it is shown in Section 4.5.2 to provide the best performance among the investigated camera input encodings. Since the C_D, L and R_{20} signals are provided in the same projection, the fusion can be performed by concatenating the inputs channel-wise to form a new one as described in Section 4.3.3. Training, validation and testing is performed on the respective data splits as described in Section 1.1.

4.5.2 Experimental Results for Camera and Camera-Radar Inputs

Based on the false rates, the deep IC_DM slightly outperforms the other purely camera-based counterparts with an accumulated overall false rate of 52.7 as compared to 54.1, both for deep $IC_{RGB}M$ and IC_SM. Looking at the true rates, this distinction becomes even clearer with the deep $IC_{RGB}M$ providing the worst true rates in all categories (true rate sum of 103.1), followed by the deep IC_SM (true rate sum 111.7) and, finally, with the deep IC_DM as the best purely camera-based model (true rate sum 126.1). This overall better performance of the deep IC_DM is also reflected in the overall decreased unknown mass compared to the other purely camera-based ISMs. The improved performance of the deep IC_DM is also reflected in the qualitative results in Fig. 4-15 by the highlighted occupancy regions and, more importantly, the increased correctness of free and occupied space contours compared to the remaining purely camera-based models.

	$\tilde{k}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	
$IC_{RGB}M$	$p(k\|d)$	33.2	23.0	5.3	38.6	28.3	33.0	5.6	33.0	36.3	18.9	4.6	40.2
	$p(\tilde{k}\|f)$	4.0	58.9	3.1	33.9	3.1	70.1	2.7	24.1	6.6	29.8	4.3	59.4
	$p(\tilde{k}\|o)$	4.1	14.6	11.0	70.3	4.3	22.6	12.2	60.8	4.1	12.2	10.6	73.1
	$p(\tilde{k}\|u)$	3.2	4.7	5.0	87.2	-	-	-	-	3.2	4.6	4.9	87.3
IC_SM	$p(k\|d)$	38.1	22.3	4.6	35.0	34.6	30.2	4.8	30.5	40.9	18.6	4.2	36.2
	$p(\tilde{k}\|f)$	4.0	62.4	2.5	31.2	3.2	73.9	1.8	21.1	6.3	32.2	4.2	57.3
	$p(\tilde{k}\|o)$	4.8	15.9	11.2	68.1	5.7	23.7	11.6	58.9	4.6	13.6	11.0	70.8
	$p(\tilde{k}\|u)$	2.6	6.5	5.4	85.5	-	-	-	-	2.6	6.4	5.4	85.6
IC_DM	$p(k\|d)$	40.8	16.8	7.0	35.3	36.1	25.5	6.6	31.7	44.3	12.0	7.1	36.6
	$p(\tilde{k}\|f)$	3.4	69.3	2.4	25.0	2.3	81.4	1.5	14.9	6.4	38.3	4.7	50.5
	$p(\tilde{k}\|o)$	6.9	16.2	16.0	60.9	8.2	25.2	15.5	51.2	6.6	13.5	16.1	63.7
	$p(\tilde{k}\|u)$	3.3	6.5	8.0	82.1	-	-	-	-	3.3	6.4	8.0	82.2
$IC_DR_{20}M$	$p(k\|d)$	42.4	14.3	12.6	30.7	38.0	20.6	13.0	28.5	45.7	11.3	12.3	30.7
	$p(\tilde{k}\|f)$	2.9	69.6	2.5	25.0	2.2	80.0	1.6	16.1	4.9	42.3	4.9	47.9
	$p(\tilde{k}\|o)$	5.0	12.0	28.8	54.1	6.4	17.2	30.0	46.4	4.7	10.5	28.4	56.4
	$p(\tilde{k}\|u)$	2.0	8.1	8.0	81.9	-	-	-	-	1.9	8.0	8.0	82.1
ShiftNet		overall				visible				occluded			

Fig. 4-14: Normed confusion matrix evaluated on the ShiftNet model which was trained on C_{RGB}, C_S, C_D and, finally, the fusion of C_D and R_{20}. For a detailed explanation of the table format kindly refer to Section 3.3.8.

More specifically, the only prominent deficit of the deep IC_DM lies in its increased false rates for $p(d|o)$ and $p(o|d)$. This shows that, compared to the other purely camera-based models, the deep IC_DM is better in estimating whether there is an object present but not in which state (static or dynamic) it is. On the other hand, both homography-based ISMs struggle to distinguish free from dynamic space. But, the incorporation of semantic information leads to a jump in the true positive rate for dynamic objects. This

is to be expected since the semantics provide class information which is directly linked to the possibility of a pixel being dynamic.

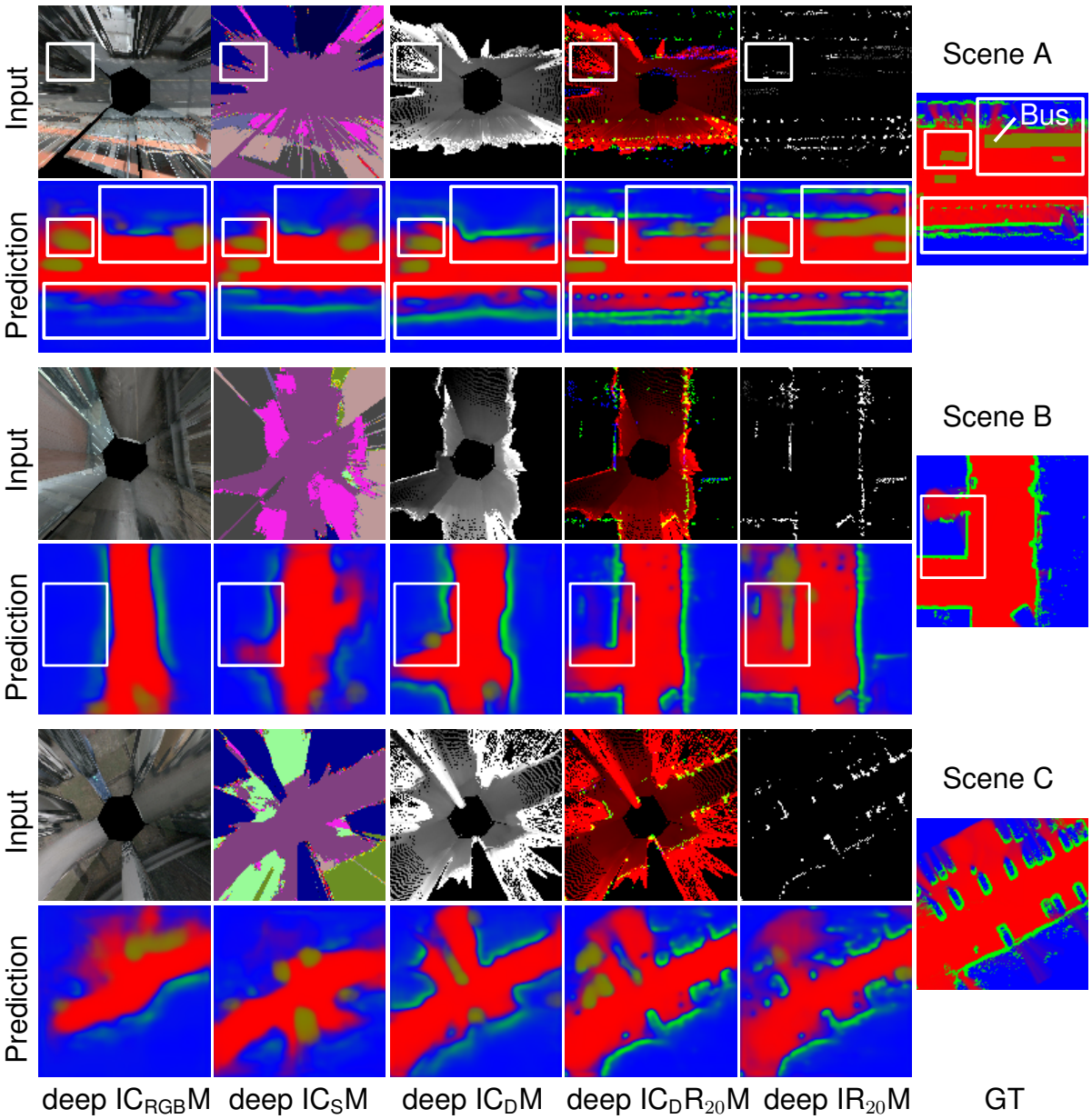


Fig. 4-15: Qualitative comparison of ShiftNets trained on different camera and radar inputs shown for three scenes with the respective GT. For a detailed explanation of the abbreviations and color format kindly refer to Section 3.3.5.

Since the deep IC$_{D}$M shows an on average increased performance over all classes, it will be used to further analyze the effect of fusion with radar. These experiments show that, compared with the deep IC$_{D}$M, the additional radar information leads to less unknown mass and an overall improvement of positive and false rates throughout all classes. While this improvement is only marginal for the free class, the occupied true positive rate is increased by 12.8% while also reducing the false rates. The only

worsening is observed with regards to false occupied predictions in the dynamic class. Here, the false rate is almost doubled compared to the other camera ISMs. Comparing to the deep $IR_{20}M$, the deep $IC_DR_{20}M$ provides better performance in each of the overall scores without exception.

A qualitative example for the improvement of the deep $IC_DR_{20}M$ over the deep $IR_{20}M$ can be seen in the white box in Scene B of Fig. 4-15. Here, the MonoDepth provides the additional information that the wall proceeds around the corner, leading the model to assign the area behind the wall as unknown rather than free. Also, the shape of occupied space in the deep $IC_DR_{20}M$ predictions is highly improved over the purely camera-based ISM predictions.

4.5.3 Experimental Results for Lidar and Lidar-Radar Inputs

When comparing the scores of the deep ILMs in Fig. 4-16 with the ones solely based on camera and radar (see Fig. 4-14 and 4-10) the deep ILM obtains the overall best results. The only exception is with regards to dynamic objects. Here, it obtains the highest false rates of dynamic predictions in the occupied class. Also, the unknown mass is reduced for each category, further illustrating ShiftNet's capability to adapt the unknown mass to account for the higher accuracy in the lidar data. For the geo ILM, the importance of treating the visible and occluded metrics separately becomes more evident than for all other ISMs. Here, the occluded area, as per definition, is mainly assigned to the unknown class. Hence, only the visible area will be considered for the following discussion.

In the visible area, it can be seen in Fig. 4-16 that there are still predictions for dynamic objects remaining. This is due to the fact that not all occupied pixels belonging to the dynamic objects get removed since the bounding box ground-truth in NuScenes often does not fully enclose the objects. The remaining occupied pixels get mixed with the free space rays leading to half occupied half free, and hence, dynamic pixels. This effect can also be seen in Fig. 4-17 in the upper white box in scene A where some contour points of the bus remain as dynamic detections. Regarding the free predictions in the visible area, they provide an almost perfect overlap with the resulting occupancy maps. Thus, the mapping doesn't change the free space much in the visible area. However, looking at the occupied class, the false free rates are among the highest of all models. This is due to the ILM's parameterization allowing free space rays to cut through occupied areas. Here, the ILM is parameterized to correct these errors during mapping resulting in good maps but sub-optimal ILM performance. This effect can be seen qualitatively in Fig. 4-17 looking at the perforated contours of occupied areas.

When comparing the deep with the geo ILM scores in Fig. 4-16, it can be seen that

the deep variant outperforms the geometric one in the dynamic and occupied class by a large margin. This can be verified in the qualitative results by looking at the continuous occupancy boundaries and the well shaped dynamic vehicle prediction in scene A. However, large portions of dynamic objects are estimated as being static. This is to be expected since the deep ILM can only guess the dynamic state based on the position of the vehicles on the road for it is not measured by the lidar sensor. An example of dynamic objects being falsely predicted as static is shown in Fig. 4-17 scene A. On the other hand, the deep ILM does not reach the same amount of correct free space mass. This might be due to the more continuous probabilistic prediction of free space as opposed to the binary one in the geo ILM, which is e.g. shown in the white box of scene B in Fig. 4-17 close to the image boundary. Additionally, estimates for the occluded area are provided which, on average, are correct. Thus, the deep ILM provides a valid alternative over the geometric counterpart.

	$\tilde{k}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{d}$	$\tilde{f}$	$\tilde{o}$	
geo ILM	$p(k\|d)$	0.2	27.6	1.7	70.3	1.8	73.2	3.6	21.0	0.0	0.0	0.0	100.0
	$p(\tilde{k}\|f)$	0.0	72.0	0.2	27.5	0.0	98.7	0.3	0.6	0.0	0.0	0.0	100.0
	$p(\tilde{k}\|o)$	0.1	15.4	7.7	76.7	0.3	60.6	32.6	6.1	0.0	0.0	0.0	100.0
	$p(\tilde{k}\|u)$	0.0	0.7	0.1	99.2	-	-	-	-	0.0	0.0	0.0	100.0
deep ILM	$p(k\|d)$	47.3	10.5	16.3	25.8	42.0	15.1	17.9	25.1	51.1	7.5	16.4	25.0
	$p(\tilde{k}\|f)$	2.4	78.8	1.9	16.9	1.6	89.3	0.9	8.1	4.5	51.1	4.6	39.8
	$p(\tilde{k}\|o)$	8.7	8.4	43.6	39.3	11.0	10.8	46.5	31.7	8.1	7.5	42.9	41.6
	$p(\tilde{k}\|u)$	3.0	8.2	8.7	80.1	-	-	-	-	3.0	8.0	8.7	80.3
deep ILRM	$p(k\|d)$	47.3	9.7	18.3	24.8	43.5	13.6	20.6	22.2	50.6	7.4	17.4	24.6
	$p(\tilde{k}\|f)$	1.8	80.8	1.9	15.5	1.2	89.8	1.1	7.9	3.6	56.8	4.1	35.4
	$p(\tilde{k}\|o)$	6.0	7.6	46.8	39.5	8.1	9.6	52.8	29.4	5.5	6.8	45.2	42.5
	$p(\tilde{k}\|u)$	1.6	7.6	7.4	83.3	-	-	-	-	1.6	7.5	7.3	83.5
ShiftNet		overall				visible				occluded			

Fig. 4-16: Normed confusion matrix evaluated on the geo ILM and the ShiftNet model, trained on the lidar BEV projection and the fusion of the lidar and radar BEV images. For a detailed explanation of the table format kindly refer to Fig. 4-10.

Finally, when looking at the deep $ILR_{20}M$, the overall scores in Fig. 4-16 among all classes are improved as expected. For the occupied and free class, the deep $ILR_{20}M$ even manages to improve the true rate while at the same time decrease the false rates. These improvements can also be qualitatively verified looking at the white box in Fig. 4-17 scene B. Here, the area behind the wall both for the deep ILM and IRM is partially or, in case of the radar, fully assigned as free. This error is corrected using the fused input.

For the dynamic class, the false and true rates remain about the same which shows the lack of distinguishability for dynamic objects provided by the radar encoding. This can again be qualitatively verified looking at the upper white box in scene A.

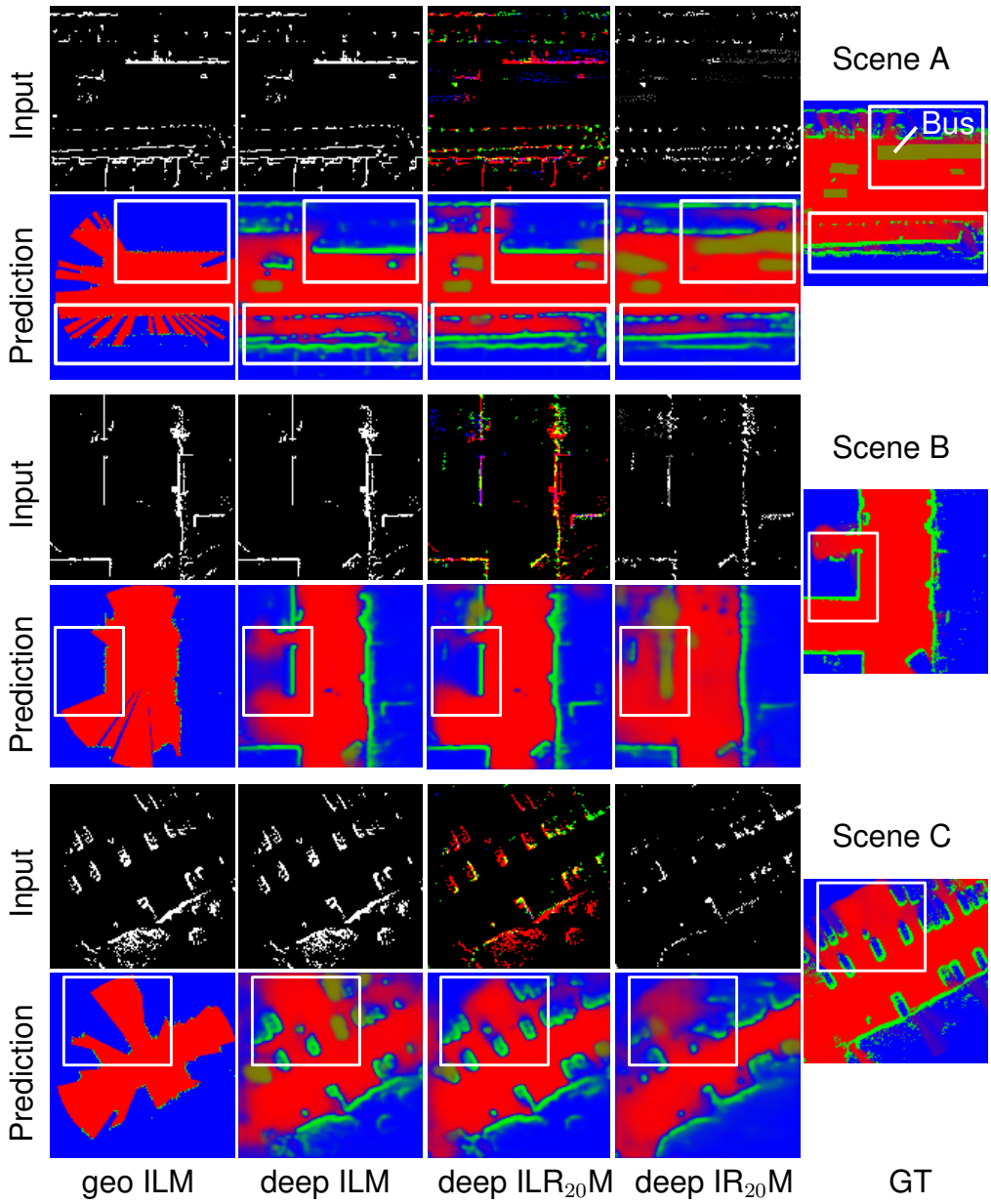


Fig. 4-17: Qualitative comparison of ShiftNets trained on different lidar and radar inputs shown for three scenes with the respective GT to the right. For a detailed explanation of the abbreviations and color format kindly refer to Section 3.3.5.

The deep IRM formerly fully recognized the bus as being dynamic while the deep ILM interpreted it as occupied. In the fused result, the bus is only partially predicted as dynamic which, thus, worsened the prediction. However, the static car in the deep ILM is assigned dynamic in the fusion and the dynamic cyclist's contour becomes more accurate in the fusion.

4.5.4 Discussion

Based on the observations in Section 4.5.2, first, the deep ICMs shall be discussed. Here, using the C_{RGB} input encoding resulted in the worst performance. This might be another indicator to showcase the flaws in using homography over MonoDepth or the other deep learning based methods mentioned in the state-of-the-art review in Section 2.3.2 to obtain environment perception from camera images in the BEV projection. However, it needs to be mentioned that the decreased performance might be due to the fact that the network's capacity has been adjusted in Section 4.2 for the R_1 input. Thus, it might be possible to eliminate some of the deficits using a bigger model and more data.

When comparing the C_{RGB} with the C_S input encoding, it can be seen that the semantic information leads to an overall better performance. This, again, might be due to the fact that the network capacity is not big enough to fully learn the mapping based on the camera image input. The biggest performance gap, however, is between the true positive rates of dynamic predictions which shows that including semantic information helps in identifying dynamic objects. Another prominent observation is the increased false rate of $p(\tilde{f}|d)$ showing the confusion of homography-based ISMs to distinguish free from dynamic space. A possible explanation for this characteristic might be the shape distortion of non-flat objects like cars during the homography projection leading to free space in areas where the camera captures space beneath the car. A qualitative example showing the distortion is depicted in the upper left white box in Scene A of Fig. 4-15. Eventually, it needs to be mentioned that, up to a certain degree, the decreased performance of the deep IC_SM can be explained by the errors of the used semantic segmentation model. Since the model has not been retrained, its performance is limited for some classes like the only partially correctly classified bus depicted in the upper right white box in Scene A of Fig. 4-15.

For the case of the deep IC_DM, objects can be detected way more clearly as compared to the homography variants. However, the model is confused whether the object is indeed static or dynamic. Here, the implicit assumption to only utilize the height information in C_D might not suffice to provide this information. Here, enhancing C_D with semantic information, e.g. using an additional channel, at least for potentially dynamic classes like e.g. pedestrians or vehicles might lead to a better distinction.

Eventually, the results of the geo and deep ILMs as shown in Section 4.5.3 will be discussed. Here, the geo ILM's free predictions provide almost perfect overlap with the ground-truth maps', showing that the mapping process does not alter the free space much. However, the occupied space shows lots of free space ray breaches which can also be verified qualitatively. This either shows deficits of the chosen parameterization

(e.g. opening angle of cone $\varphi_{\sphericalangle}$ might be too small) or hint to potential improvements of the geo ILM's algorithm. With regards to the deep ILM, the occupied contours are more continuous compared to the geo ILM's. Also, the capability to initialize areas occluded for the geo ILM can be quantitatively as well as qualitatively shown.This shows the potential to provide better ground-truth occupancy maps when applying the deep over the geo ILM. Nevertheless, the dynamic objects can only be estimated partially correct and the scores indicate lots of cases where dynamic objects are predicted as static. This, though, is to be expected since the lidar does not provide measured information about the motion status, leading the deep ILM to estimate the motion state purely based on their position on the road. One example is shown in the white box in scene C of Fig. 4-17. Here, the opening in the parking lot can also be interpreted as a street with the parked vehicle moving along it.

The above examination of the deep ISM properties provides the remaining answer to RQ4 with regards to the proposed deep ISM's properties given camera and lidar inputs respectively. Eventually, to answer the question about the proposed deep ISM's properties for fused inputs, as formulated in RQ5, the findings in Section 4.5.2 and 4.5.3 regarding the fusion with radar shall be examined.

Starting with the fusion of C_D with R_{20} images, the quantitative results show that the radar information improves the overall performance in all classes. For the occupied space, the true rate rises by over 12$\%$ which can be verified by the improved edge quality of occupied space in the qualitative results. For free space, the amount of false dynamic predictions is decreased further showing the improvement of edge information delivered by the radar. In case of dynamic areas, however, an increase in 5$\%$ of false occupied rate can be seen. This problem has been examined already in Section 4.3.2 and seems to originate from either the proposed encoding of dynamic detections or the overall quality of the provided motion status flag of the detections provided in NuScenes.Thus, the radar helps in identifying that there is indeed an object but shows confusion whether it is indeed dynamic or not.

Looking at the fusion of L and R_{20} in deep ISMs, as discussed in Section 4.5.3 similar behavior can be observed. Again, the addition of radar detections overall improves all scores while it also leads to an increase in false occupied rate in the dynamic class. Qualitatively, it can be seen in Fig. 4-17 that the radar indeed resolves some of the false dynamic predictions (see white box in scene C) while other areas are falsified (see dynamic area above white box in scene B that is static in deep ILM and becomes dynamic in the fusion).

Finally, with regards to RQ10, the experiments again show that the unknown mass predicted by the ShiftNet is correlated with the information content in the inputs by de-

creasing starting with the camera input, over camera-radar, to lidar and being overall smallest for lidar-radar inputs. This qualifies the unknown mass estimated by ShiftNet to be used as a measure for information in the subsequent occupancy mapping experiments, as proposed in Section 3.3.9.

5 Deep, evidential ISMs as Priors for Occupancy Mapping Experiments

This chapter contains the investigation of temporally accumulating the afore defined ISMs. This investigation is started with the analysis of the evidential combination rules mentioned in the SotA against the ones proposed by the author in Section 3.3.9 and 3.3.10. Afterwards, the hypothesis of redundancy accumulation for occupancy maps created with deep ISMs H2 is investigated together with the effectiveness of the solution proposed in Section 3.3.9. Given these results, occupancy maps are created using the best performing geo as well as deep ISMs for each sensor modality as identified in Chapter 4. Eventually, the fusion approach proposed in Section 3.3.10 is analyzed with regards to violations of the proposed bounds and improvements over the baseline fusion. Besides the comparison of combination rules in the first part of the chapter, which is based on a simulated input signal, all experiments are based on the NuScenes dataset (see Section 3.2.4) using the reduced train-val-test split (see Section 1.1).

5.1 Experiment to verify Choices of Combination Rules

In this section, a simulated verification of the combination rules' properties derived in Section 3.3.10 is provided.

5.1.1 Setup of the Verification of Combination Rule Choices

To demonstrate the properties, an evidential occupancy signal (first row Fig. 5-1) is fused over time with a mass initialized with $m_u = 1$ using different combination rules. This simulates the occupancy state of a single cell in the occupancy map over time. The signal starts with a step-wise ramp-up of the occupied mass, followed by a completely contradictory signal, a signal with partial conflict, two consecutive contradicting signals and, eventually, a signal with full conflict.

5.1.2 Results of the Verification of Combination Rule Choices

Beginning with Dempster's rule, it can be seen that the fused mass always fully converges to the class of the signal with the highest portion (see Dempster graph e.g. in the time interval $[0, 20]$). Given temporal independent information and no conflict, this behavior would be desired since each time step provides new evidence of the state. However, in presence of conflict, the fused mass should converge to a state representing the portion of conflict to represent cells being dynamic. Since Dempster's rule lacks the ability to do so, it is disqualified for usage in this work.

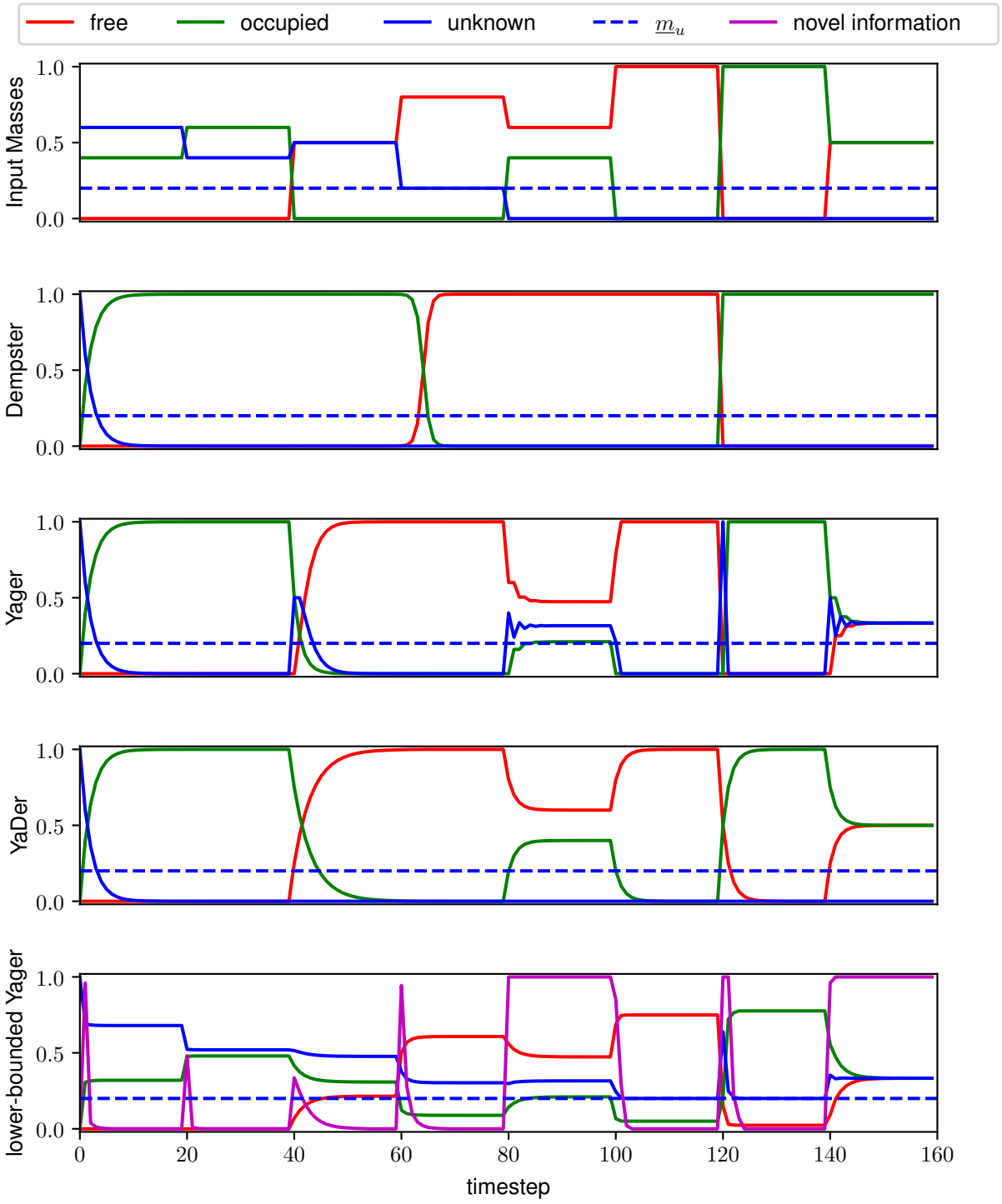


Fig. 5-1: This figure shows the qualitative evaluation of the different evidential combination rules given input signals for free, occupied and unknown mass (first subplot) over time with an exemplary lower unknown threshold $\underline{m}_u$ overlayed.

Similar to Dempster's rule, Yager's rule, in absence of conflict, converges fully to the dominating class (see Yager graph e.g. in the time interval $[0, 20]$). In presence of conflicting mass, Yager's rule recuperates unknown mass (see e.g. time interval $[35, 45]$) and allows to directly switch the state, as opposed to Dempster's rule (compare Dempster and Yager graph in time interval $[35, 45]$). The recuperation capability is useful in the initialization phase to correct for state changes without falling below $\underline{m}_u$. Also, while Yager's rule is capable of representing the conflict state of a signal (see Yager graph in the time interval $[80, 100]$ and $[140, 160]$) it never reaches the original input conflict due to the recuperated unknown mass. It shall be argued that the property of recuperating unknown mass outweighs the deficit of biased conflict representation for the purpose of initializing the occupancy state.

To solve the biased conflict state in the convergence phase, the YaDer rule has been proposed. It can be seen that by equally distributing the conflict into the conflicting classes instead of shifting it to the unknown class, the fused mass converges to the input's conflict state (see YaDer graph in the time interval $[80, 100]$ and $[140, 160]$). Also, in case of absent conflict, YaDer rule, like Yager and Dempster, fully converges to the dominating class (see YaDer graph e.g. in the time interval $[0, 20]$). Thus, the YaDer rule combines both the Yager rule's property to represent conflict and Dempster rule's property to strictly reduce the unknown mass, giving it its name. These properties makes YaDer rule an ideal candidate to be used in the convergence phase.

Finally, the lower-bounded (lb) Yager rule's property to only converge to the input signal until the same unknown mass is reached is shown in the last graph (e.g. in time interval $[0, 30]$). It can be seen that there is a discrepancy between the converged fused state and the input mass. This is due to the fact that the input is assumed to originate from a deep ISM with limited certainty. Therefore, the input is first rescaled into the interval $m_u \in [\underline{m}_u, 1]$ before fed into the lb Yager rule. Moreover, it can be seen when comparing the time intervals $[35, 45]$ and $[45, 55]$ that a state switch between free and occupied can only be realized if the conflicting signal has high enough certainty relative to the current state. This is desired in order not to overwrite a highly certain estimate, potentially predicted close to data, with lower certainty ones, potentially based on extrapolations. Problematic, however, is that the state change is not fully completed letting the fused mass converge to a conflicting state. This is due to increasingly discounting the input signal when reducing the difference in unknown mass between input and fused signal. This property ensures that once $\underline{m}_u$ is reached, the deep ISM can not further influence the fused mass (see lower-bounded Yager graph in the time interval $[100, 160]$). Here, it shall be argued that the benefit of reducing the influence of outliers and the property of deactivating the deep ISM's influence outweigh the shortcoming of incomplete state change for the initialization phase.

5.1.3 Discussion of the Verification of Combination Rule Choices

The above results demonstrate the claimed properties of all combination rules and, thus, verify the choices in Section 3.3.10. With regards to RQ11 to disable the influence of the deep ISM, the results for the lower-bounded Yager rule clearly show the restricting properties of the fusion to suffice a lower bound on the unknown mass. However, to fully verify the robustness of the fusion approaches and investigate additional cases not covered in the above simulations, the proposed approach is applied on real data in the following sections.

5.2 Analysis of redundant Information in deep ISMs

This section focuses on the experimental verification of H2 which concerns the accumulation of temporal redundancy in deep ISM mapping.

5.2.1 Setup of Redundancy Analysis in deep ISMs

To verify H2, occupancy maps are created in five ways. The baseline is the direct accumulation using Yager's rule, abbreviated as "Yager". This is evaluated for the deep as well as the geo ISM. Second, a naïve solution is evaluated which moves all free and occupied mass to unknown for predictions with $m_u >= 0.9$ to hinder small predictions to accumulate over time ("Yager + cutoff"). Third, the method as proposed in Section 3.3.9 is used to reduce the redundancy of the deep ISM predictions before accumulation with Yager's rule. The so reduced predictions are abbreviated with "deep, red.". Eventually, the fusion using the deep ISM with and without redundancy reduction respectively with the geo ISMs by accumulating the predictions using Yager's rule is evaluated. Here, the fusion is investigated to analyze whether the deep ISM indeed overwrites most of the geo ISM's predictions as suggested in Section 3.3.10.

The evaluation is performed using only the accumulated radar detections of the recent 20 timesteps (R_{20}) since the focus of this work lies on radar occupancy mapping. Additionally, it is assumed that the effects are similar using other sensor modalities. The geo ISM method is used as described in Section 3.3.3 and ShiftNet, as described Section 3.3.7, is used as a deep ISM.

The metrics used are the normed confusion matrix (see Section 3.3.8) and the mIoU. Here, the metrics are evaluated separately in all of the area in a 20 m vicinity around the ego vehicle trajectory ("whole mapped area") and in an area of 15 pixels around occupied ground truth pixels in the reference occupancy maps. Here, the mIoU is only evaluated around the occupied borders to quantify the cleanliness. The reference maps are created by accumulating the geo ILM using Yager's rule. Two examples of

the evaluation areas can be found in 5-3. The scenes mapped are solely taken from the test set (see Fig. 1-4) which was not used during training.

5.2.2 Results of Redundancy Analysis in deep ISMs

The findings in Fig. 5-2 show that, even though, the geo $IR_{20}M$'s occupancy map has the least true positive rates overall, it also has by far the least false rates. More specifically, the deep $IR_{20}Ms$ without redundancy reduction produce about four times and the one with the reduction about twice the amount of false occupied predictions in the whole mapped area. Around the boundaries, this ratio even goes up slightly. For free false rates, the ratio of geo $IR_{20}M$ maps is about half of the other variants. On the other hand, the geo $IR_{20}M$ maps only reach about half the true occupied and about two third of the free positive rates. Also, the sparse nature of the geo $IR_{20}M$ can be seen for the unknown class where the accumulation of the deep $IR_{20}M$ results in ten times the false rates both for free and occupied. This is also reflected by the "cleanliness" of boundaries as measured by the mIoU where this variant reaches the best score in the occupied and unknown classes and is clearly shown in Fig. 5-3 2nd column for both scenes.

Next, scores of the accumulated deep $IR_{20}M$ reach the highest true and false rates for occupied predictions especially in the unknown regions which can be seen in all of the marked boxes in Fig. 5-3 column three to six. This also leads to the lowest mIoU scores in the boundary region. When fusing the predictions with the geo $IR_{20}Ms$, the scores do only minimally change hinting to a domination of the deep $IR_{20}M$ over the geometric one. The qualitative results verify this domination by barely showing any signs of the geo $IR_{20}Ms$ influence. By looking closely in the upper white box in Fig. 5-3 scene B 4th column some occupied detections of the geo $IR_{20}M$ can be found.

The naïve solution to cutoff low probable predictions $m_u > 0.9$ and set them to unknown slightly reduces the true and false rates of occupied predictions. However, in unknown areas, the amount of occupied predictions is more drastically reduced. Interestingly, the false and true rates of free space slightly increase through the cutoff. These findings can also be verified in Fig. 5-3 column four and five by comparing the accumulated deep $IR_{20}M$ with and without cutoff. Here, the most significant change can be seen by a reduction in occupied predictions in the unknown areas. In some cases (white box in scene A), free space replaces the former occupied areas.

Applying redundancy reduction before accumulation has a similar effect compared to cutoff. The occupied rates decline while the free rates are increased. However, in the unknown area, both the free and occupied rates are reduced in contrast to cutoff where the free rate is increased.

	$\tilde{k}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$
geo	$p(\tilde{k}\|f)$	57.9	2.3	39.8	44.3	5.9	49.8
$IR_{20}M$	$p(\tilde{k}\|o)$	16.1	36.3	47.6	15.9	36.0	48.1
Yager	$p(\tilde{k}\|u)$	2.8	5.7	91.5	4.0	9.7	86.3
	mIoU	-	-	-	16.8	24.6	14.2
deep	$p(\tilde{k}\|f)$	86.6	11.7	1.7	68.2	28.4	3.4
$IR_{20}M$	$p(\tilde{k}\|o)$	27.2	69.6	3.2	27.3	69.4	3.3
Yager	$p(\tilde{k}\|u)$	28.9	57.5	13.6	27.2	64.5	8.3
	mIoU	-	-	-	15.3	12.4	2.8
deep & geo	$p(\tilde{k}\|f)$	86.7	11.6	1.7	68.4	28.3	3.3
$IR_{20}M$	$p(\tilde{k}\|o)$	27.3	69.5	3.2	27.4	69.3	3.3
Yager	$p(\tilde{k}\|u)$	29.0	57.4	13.6	27.3	64.4	8.3
	mIoU	-	-	-	15.3	12.4	2.7
deep	$p(\tilde{k}\|f)$	87.4	10.5	2.1	69.4	26.7	3.9
$IR_{20}M$	$p(\tilde{k}\|o)$	28.3	67.0	4.7	28.4	66.8	4.8
Yager + cutoff	$p(\tilde{k}\|u)$	33.7	34.7	31.6	30.4	51.9	17.7
	mIoU	-	-	-	15.1	16.5	5.2
deep, red.	$p(\tilde{k}\|f)$	88.5	5.9	5.6	71.6	16.7	11.7
$IR_{20}M$	$p(\tilde{k}\|o)$	29.0	55.7	15.3	29.0	55.4	15.6
Yager	$p(\tilde{k}\|u)$	26.3	12.9	60.8	26.3	23.4	50.3
	mIoU	-	-	-	16.8	23.4	11.8
deep, red. & geo	$p(\tilde{k}\|f)$	88.9	5.8	5.3	72.9	16.4	10.7
$IR_{20}M$	$p(\tilde{k}\|o)$	29.7	56.6	13.7	29.7	56.3	14.0
Yager	$p(\tilde{k}\|u)$	26.8	14.3	58.9	26.9	25.2	47.9
	mIoU	69.1	24.0	27.0	17.0	24.1	11.4
		whole mapped area			boundary area		

Fig. 5-2: Normed confusion matrix for the three mapping variants using deep ISMs based on ShiftNet applied on different sensor modalities. See 5.2.1 for further information on the methods and abbreviations used.

Another difference is the ratio of changes which, compared to cutoff, is more in favor of true rates. The better ratio together with the improved performance in unknown areas, eventually, lead to a strictly better edge preservation that almost reaches the level of the geo $IR_{20}M$'s mIoU. These findings can also be verified in the white box in scene A of Fig. 5-3 comparing column six with the others. Here, the accumulation of free space in the occluded area is reduced compared to the cutoff variant. Also, in the white boxes in scene B, the occupied edges are less spread. At the same time, there is also less weight put into occupied areas.

Finally, the fusion of the reduced deep IRM with the geometric further shows the ben-

efit of the proposed method. Here, all but the free false rate improve while the free false rate only slightly worsens. In the boundary area, the overall improvement leads to reaching the best score for free space and a close performance in the other classes compared to the geo IRM map. These findings are also qualitatively verified in Fig. 5-9 in Section 5.4.2.

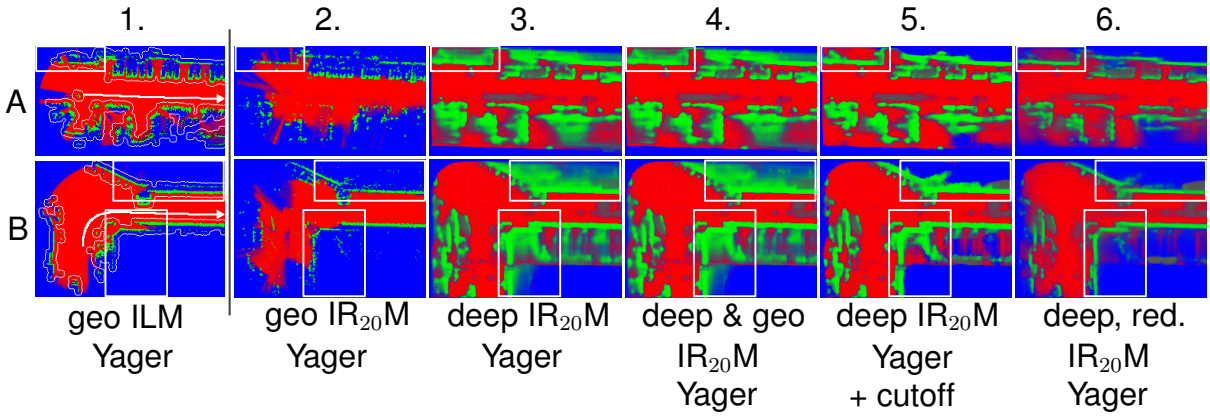


Fig. 5-3: Illustration of qualitative results for the different occupancy mapping variants for two scenes (A & B). The vehicle's trajectory is overlayed in white over the geo ILM map.

To further verify the effect of the redundancy reduction update, three steps of the mapping procedure are visualized in Fig. 5-4. It can be seen that in the first step, when all map cells are initialized to unknown, the complete prediction of the deep IRM is used to update the map. Afterwards, the majority of new information comes from areas that just entered the deep IRM's FoV. In the depicted case, the ego vehicle is moving from left to right and, thus, the majority of new information is at the right border (see e.g. white box on the right for $t = 10$). Besides from areas entering the ISM's FoV, formerly occluded areas that become visible are also identified to provide new information (see white box for $t = 12$). Finally, the redundancy reduction is formulated to treat regions of conflict as rich in new information. This is done to allow the map to react to changes in the environment e.g. a parked vehicles starting to move. Thus, dynamic objects are also treated as new information and remain in the update (see left white box for $t = 10$).

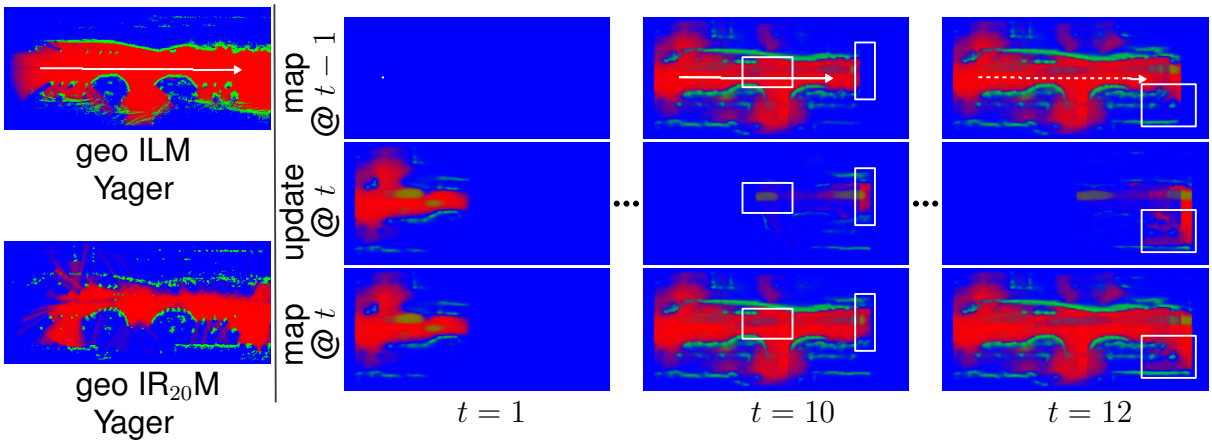


Fig. 5-4: Mapping process given redundancy reduced deep IRM updates. Left, the geo ILM and IRM map are shown for reference. Right, the first row shows the map state before the update, the second the redundancy reduced deep IRM update based on the first row's state and, finally, the updated map in the last row. Additionally, the full trajectory is overlayed in white on the geo ILM map. For the map progress, the trajectory is shown as solid for the time between visualizations and dashed for previous timesteps.

5.2.3 Discussion of Redundancy Analysis in deep ISMs

The experiment in Section 5.2.2 shows the problem of uncontrolled accumulation of masses in occluded areas. This property leads to fully overwriting the geo IR_{20}Ms predictions during mapping. This makes the use of the original deep IR_{20}M in fusion with the standard Yager rule prohibitive.

Applying cutoff reduces the distribution of occupied mass in unknown areas more compared to visible. This proves H2 by showing that small biases are being accumulated. Here, the cutoff might have been chosen too small since, instead of occupied, now free mass is being accumulated in occluded areas (see Fig. 5-3 white box in scene A and free rate in unknown area in Fig. 5-2). However, by further increasing the cutoff threshold, only high certain estimates can be conserved leading to too much loss in information of the deep ISM predictions.

Using redundancy reduction, the low certain estimates can still be included in the mapping. But, they can only contribute up to their certainty and, thus, not overwrite areas by sheer number. Additionally, no hyperparameters are introduced. Among the deep IR_{20}M based maps, this leads to the best edge preservation even coming close to the geo IR_{20}M map with regards to mIoU (see Fig. 5-2). It is also qualitatively verified in Fig. 5-4 that only conflicting estimates (e.g. dynamic objects or falsely predicted regions) and areas with lower unknown mass (e.g. formerly completely unknown areas) are updated. This verifies the properties claimed in Section 3.3.10 of

the lower-bounded Yager rule. The benefits of the redundancy reduction are further verified by the results of its fusion with the geo IRM. The improvements of the scores clearly shows the influence of the geo IRM is not negated but instead utilized. The qualitative and quantitative verification of the redundancy removal approach described in Section 3.3.9 clearly prove the problem of temporal redundancy accumulation, as claimed in H2. Furthermore, the benefits of the proposed countermeasure prove the choice of mutual information and, based on it, the discount factor to be valid. Thus, verifying the proposed answers of Section 3.3.9 to RQ8 and 9.

Since the problem indeed seems to originate from the accumulation of false predictions in occluded areas, another solution might be to directly train the deep ISM to predict the geo ILM and not the mapped geo ILM patches. This solution has not been considered from the start since the work's aim is to initialize as much space around the vehicle as possible to increase the convergence speed. Also, it might be possible to train the deep ISM to, based on a given state of a map, predict either the best next occupancy state or the next update. This, however, is highly non-trivial since the network has to be trained to cope with the maps created by a number of sensor models including itself.

Even though the redundancy reduced $IR_{20}M$ map comes close to the edge performance, it still lags behind the geo $IR_{20}M$ map's mIoU. Thus, in Section 5.4, it is analyzed how the inter- & extrapolation properties of a deep ISM can be combined with the edge accuracy of a geo ISM during mapping.

5.3 Comparison of deep ISM Occupancy Maps given different Sensor Modalities

In this section, RQ12 shall be tackled concerning the comparison of occupancy mapping results given the proposed deep ISM with different inputs.

5.3.1 Setup of Deep ISM Maps Comparison

The setup for comparing occupancy maps given different sensor modalities is the same as described in 5.2.1. The only difference is that for this experiment the fusion method is chosen to be the redundancy reduced accumulation with Yager's rule for all variants (abbrev. as "deep, red."). The compared ISMs are all trained ShiftNets while the inputs are MonoDepth (C_D), lidar (L) and their respective combinations with the accumulated radar point cloud of the last 20 steps (R_{20}).

5.3.2 Results of deep ISM Maps Comparison

The quantitative results show a similar line of improvement as shown for the ISM comparison in Section 4.5 and are overall consistent in the whole as well as the boundary area. More specifically, the MonoDepth input lags behind the lidar input in all classes. While, the false occupied rate is doubled for the lidar resulting in a 1% increase, the true occupied rate is more than three times higher providing an increase of almost 50%. This improvement can be clearly verified qualitatively comparing the sharpness of occupied boundaries between MonoDepth and lidar maps. Moreover, adding radar information provides a further improvement in true rates. This is especially true for the MonoDepth case, showing that providing measured depth over estimated depth improves the occupancy predictions. For the false rates, radar provides consistent improvement for the free class while slightly decreasing the performance for the occupied case.

	$\tilde{k}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$
deep, red.	$p(\tilde{k}\|f)$	89.1	1.1	9.8	72.5	3.3	24.2
IC_DM	$p(\tilde{k}\|o)$	34.3	19.1	46.6	34.1	18.9	47.0
Yager	$p(\tilde{k}\|u)$	11.4	4.4	84.2	13.9	8.8	77.3
	mIoU	77.8	13.5	34.6	19.3	13.4	15.7
deep, red.& R_{20}	$p(\tilde{k}\|f)$	88.4	2.3	9.3	69.2	7.2	23.6
IC_DM	$p(\tilde{k}\|o)$	20.3	45.7	34.0	20.2	45.3	34.5
Yager	$p(\tilde{k}\|u)$	12.8	7.0	80.2	12.9	13.8	73.3
	mIoU	78.0	28.3	33.4	18.6	28.2	15.2
deep, red.	$p(\tilde{k}\|f)$	93.4	2.0	4.6	81.5	6.4	12.1
ILM	$p(\tilde{k}\|o)$	14.2	68.8	17.0	14.1	68.3	17.6
Yager	$p(\tilde{k}\|u)$	14.5	11.4	74.1	14.7	22.3	63.0
	mIoU	82.2	38.0	32.6	21.6	37.9	14.0
deep, red.	$p(\tilde{k}\|f)$	93.9	2.2	3.9	82.0	7.2	10.8
$ILR_{20}M$	$p(\tilde{k}\|o)$	12.7	71.0	16.3	12.7	70.6	16.7
Yager	$p(\tilde{k}\|u)$	12.2	9.9	77.9	11.9	19.5	68.6
	mIoU	83.3	40.7	34.8	21.9	40.7	15.2
		whole mapped area			boundary area		

Fig. 5-5: Normed confusion matrix for deep ISMs based on ShiftNet applied on different sensor modalities. See 5.3.1 for further information on the methods and abbreviations used.

The findings of Fig. 5-5 are further qualitatively verified in Fig. 5-6. It can again be seen that the overall quality improves from radar (column three), over MonoDepth (column four) to lidar (column six) and again when fusing the sensor inputs (columns five and seven). More specifically, looking at the upper white box in scene A, the falsely

extrapolated free space of the deep IRM map behind the row of parked cars can be corrected in the fusion with MonoDepth. Additionally, the gaps between the two cars parked in the middle of the upper and lower four is barely visible in the MonoDepth map while it is well defined in the fusion with radar. This shows that inaccuracies of the radar as well as the MonoDepth map can be corrected through the fusion. Furthermore, the lidar map shows the gaps and contours with more precision which is even improved providing additional radar information. This again shows that the deep ISM is capable of utilizing the improved accuracy of the sensor and further translate this improvement into the mapping process. These effects can again be seen in the left white box in scene B where both the radar and MonoDepth maps fail to properly capture the opening of the alley. However, the fused map can capture it properly.

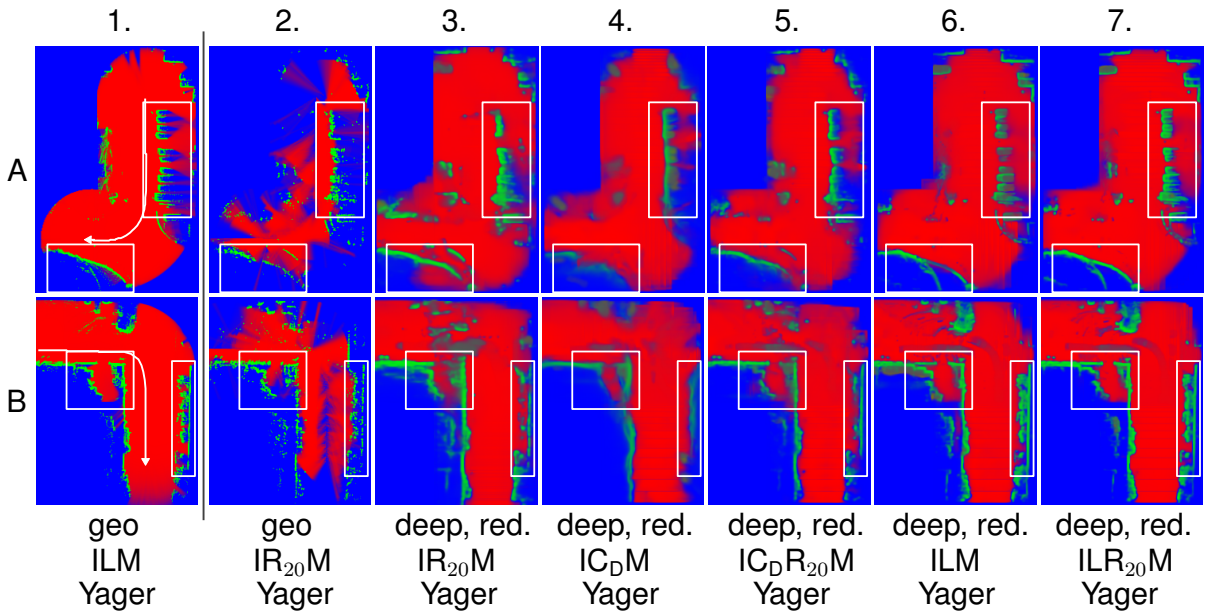


Fig. 5-6: Qualitative results of the occupancy maps created using different ISMs. The abbreviations are explained in Section 5.3.1. The vehicle's trajectory is overlayed in white over the geo ILM map.

The analysis in this section provides a quantitative as well as qualitative comparison of the occupancy maps created by the proposed deep ISM using different sensor input, thus, answering RQ12.

5.4 Analysis of deep ISM Priors for Occupancy Mapping

In this section, it shall be analyzed how the high accuracy of geo ISM around edges can be combined with the inter- & extrapolation properties of deep ISMs by initializing the map using the deep and finetuning it with the geo ISM.

5.4.1 Setup of deep ISM Priors Analysis

The experiments are performed using the same sensor inputs, test sets and ground truth as described in Section 5.2.1 for the same reasons. The methods compared create maps by accumulating

- the geo IRM using Yager's rule,
- the redundancy reduced deep and geo IRM combined with Yager's rule and $\underline{m_u} = 0$ for the deep IRM and
- the fusion of the reduced deep IRM with $\underline{m_u} = 0.3$ using Yager while the geo IRM is fused using Yager for all cells with $\underline{m_u} < 0.3$ and YaDer elsewhere.

The considerations and definitions for this approach are detailed in Section 3.3.10.

The metrics for the comparison are normed confusion matrices as described in Section 3.3.8. This metric is separately evaluated in the total area influenced by the geo IRM (geo IRM $(m_u) < 1$) and for all areas where the geo IRM was false or correct respectively. This distinction is chosen to quantify the overall improvement of the fused maps over the geo IRM and to specifically quantify the falsification and correction properties. For lower bounded deep IRM fusion, only the areas with $m_u > \underline{m_u}$ are evaluated since the method lacks the capability to provide this distinction. The remaining areas stay in the initialization phase and, therefore, are of lesser interest. Additionally, the percentage and absolute number of violations of the lower bound m_u in areas untouched by the geo IRM are investigated to verify the properties described in Section 3.3.10.

5.4.2 Experiment of deep ISM Priors Analysis

First, the redundancy reduction's property to suffice a lower bound $\underline{m_u}$ during fusion, which is proven on simulated data in Section 5.1, shall be additionally verified using the real world sensor inputs. Fig. 5-7 shows the violations of the lower bound in percentage and absolute number. It can be seen that, given a lower bound of $\underline{m_u} = 0.3$, a small percentage of cells violate the lower bound. Provided a grace interval of 0.002, these violations, however, vanish.

violations of $\underline{m_u} = 0.3$	percentage	total amount
fused Map(m_u < 0.300) | geo IRM Map(m_u = 1)	0.00023%	2476
fused Map(m_u < 0.298) | geo IRM Map(m_u = 1)	0.0%	0

Fig. 5-7: Analysis of the amount of lower bound violations. A violation is given when the maps unknown mass m_u falls below the lower bound $\underline{m_u} = 0.3$ for pixels untouched by the geo IRM (geo IRM map's unknown mass equals one). The violations are computed for the strict lower bound and given some slack of 0.002 on the lower bound.

Next, the falsification and correction properties of the fusion approaches are analyzed. Starting with deep & geo IRM map with $\underline{m_u} = 0$, the unknown mass is the lowest of all variants showing the dense properties of the deep IRM. However, this also causes an increase in false rates. Here, the lower bounded fusion overall decreases the rates. But, the decrease in false rates is higher compared to the decrease in true rates resulting in an overall better ratio of false to true classification.

For the areas, where the geo IRM map is correct, it can be seen that the lower bounded deep IRM leads to some areas becoming unknown while the majority of the already correct predictions remain correct after the fusion. Looking at the fused maps without lower bound, the deep IRM influence causes more falsification as seen at the higher false rates and lower true rates.

In the areas, where the geo IRM map is wrong, the lower bounded deep IRM leads to worsening the false rates which, however, is still increased given no lower bound. Nevertheless, the lower bounded deep ISM also leads to a great portion of improvement transforming unknown mass to correct assignments. For the unrestricted fusion, however, the true rates exceed the ones of the restricted fusion.

	$\tilde{k}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$	$\tilde{f}$	$\tilde{o}$	$\tilde{u}$
geo	$p(\tilde{k}\|f)$	59.9	8.1	32.0	100.0	0.0	0.0	0.0	20.9	79.1
$IR_{20}M$	$p(\tilde{k}\|o)$	21.7	50.1	28.2	0.0	100.0	0.0	42.7	0.0	57.3
Yager	$p(\tilde{k}\|u)$	11.1	33.7	55.2	0.0	0.0	100.0	25.6	74.4	0.0
deep, red. & geo	$p(\tilde{k}\|f)$	78.3	16.7	5.0	88.1	9.3	2.6	63.1	28.4	8.5
$IR_{20}M$	$p(\tilde{k}\|o)$	29.2	65.1	5.7	8.3	89.8	1.9	47.9	42.4	9.7
YaDer($\underline{m_u} = 0$)	$p(\tilde{k}\|u)$	33.4	48.9	17.7	41.0	29.8	29.2	23.6	73.2	3.2
deep, red. & geo	$p(\tilde{k}\|f)$	77.8	13.0	9.2	91.4	6.0	2.6	57.0	24.1	18.9
$IR_{20}M$	$p(\tilde{k}\|o)$	25.2	64.5	10.3	5.2	92.5	2.3	46.8	34.2	19.0
YaDer($\underline{m_u} = 0.3$)	$p(\tilde{k}\|u)$	27.8	43.0	29.2	31.5	18.8	49.7	23.2	73.2	3.6
		geo $IR_{20}M(m_u) < 1$			geo $IR_{20}M(m_u) < 1$ & correct			geo $IR_{20}M(m_u) < 1$ & false		

Fig. 5-8: Normed confusion matrices evaluated for the geo IRM map, the fusion of the reduced deep & geo IRM and the fusion of the lower bounded, reduced deep & geo IRM. The confusion matrices are evaluated in the area influenced by the geo IRM (left), the correctly influenced area (middle) and the falsely influenced area (right). Additionally, the lower bounded method is only evaluated for cells in the convergence phase ($m_u < \underline{m_u}$). Examples of areas in the convergence phase are depicted in right Fig. 5-9.

Looking at the qualitative results in Fig. 5-9, the property of the deep IRM to inter- & extrapolate can be seen between radar detections and in areas lacking information. One notable example of extrapolation is marked in the lower white box of scene A where the area is assigned/initialized by the deep IRM variants (columns three to five) while the geo IRM (column two) lacks to provide this information. However, in some

cases (e.g. upper white box in scene A column three), the extrapolation might be false leading to mass that has to be reassigned later. Additionally, wrong interpolations (e.g. white box on the upper right in scene B column three) or even direct errors in presence of detections (e.g. white box on the upper left of scene B column three) can occur in the deep IRM. In these cases, using the deep IRM purely for initialization leads the areas to remain in a distinguishable state in contrast to the verified cells in the convergence phase (column five). Also, the influence of a dynamic object following the ego vehicle (e.g. bottom white box in scene B) can be dampened and distinguished using the deep IRM as a prior.

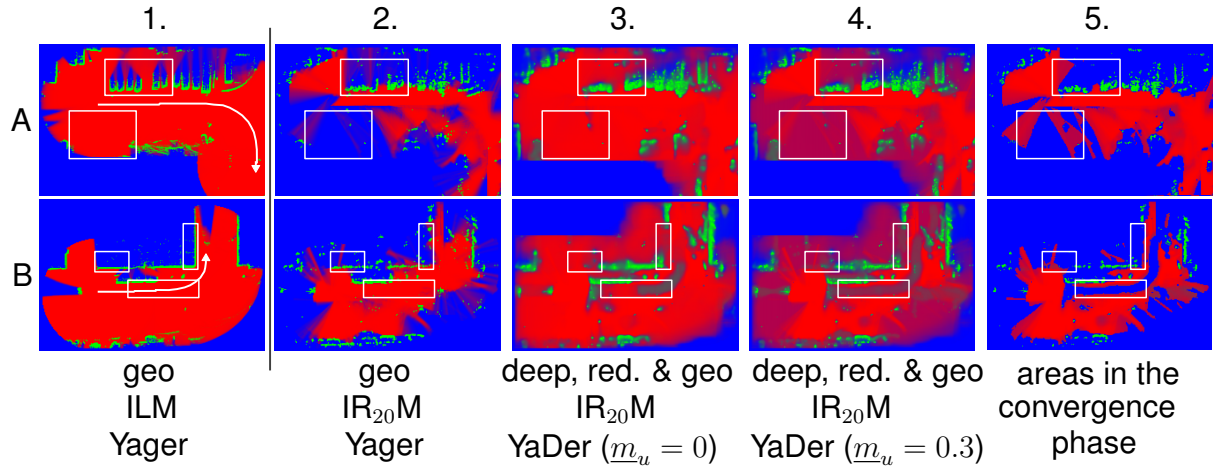


Fig. 5-9: Qualitative results of three mapping approaches (column 1-4) for two scenes with the geo ILM ground-truth map in the first column. Additionally, the 4th column shows the result of filtering out all non-converged pixels of the 3rd column's map. This is done by setting all pixels with $m_u > \underline{m}_u = 0.3$ to $m_u = 1$. The vehicle's trajectory is overlayed in white over the geo ILM map.

5.4.3 Discussion of deep ISM Priors Analysis

The above shown experimental results on real world sensor data together with the verification based on simulation in Section 5.1 show that the method devised in Section 3.3.10 is indeed capable to restrict the fusion of an ISM to a lower bound given a grace interval to allow for numerical instability, which provides an answer to RQ11. It has also been shown that the majority of the geo IRM map's correct predictions remain intact. The found falsification is to be expected since it is still a fusion approach. If both models result in similar errors, e.g. due to incorrect modeling, missing detections, false dynamic state of a detection etc., these errors are amplified resulting in a falsification. It is, however, shown that the falsification is reduced using the deep IRM as a prior, again demonstrating the restrictive influence of the deep ISM purely for initialization.

6 Summary and Outlook

In this chapter, the experimental findings from Chapter 4 and 5 are summarized and compared with the research questions and hypothesis defined in Section 3.1 and 3.2. Based on this summary, open questions are derived, based on which future research directions are being proposed.

6.1 Summary and Outlook of geo ILM and IRM

To create the targets for the free, occupied and unknown class to train the deep ISMs, the usage of occupancy map patches created by accumulating a geo ILM over time is proposed. This geo ILM needs to only be manually defined and tuned once. Thus, it suffices R1.3 to minimize manual labor. It is further tuned for the dimensions defined in R1.5 so that the map patches as well as all models trained on them meet the requirement. However, since the lidar sensor does not measure motion information, the manually annotated labels for dynamic objects are overlayed. This can be further optimized e.g. by clustering the lidar points and determining the cluster's motion state through radar measurements or by using a Doppler lidar [MA19].

Using the preliminarily tuned geo ILM maps as reference, three variants of geo IRMs are compared to analyze how to enhance the free space prediction coverage without worsening the other classes, as formulated in RQ2. Here, both adding accumulating detections over time and casting additional free space cones with bigger opening angle lead to improvements consistent over all classes. However, there is still no free space coverage in regions without detections (see Fig. 5-9 lower white box of scene A). Hence, further improvements might contain casting free space rays in regions whenever no detections are provided within a certain opening angle.

Using the best performing geo IRM variant as reference, the initially tuned geo ILM is analyzed with regards to the best ground-plane-removal technique to provide maximal overlap with the geo IRM, as formulated in RQ1. Here, removing all street, sidewalk and terrain detections of the lidar sensor provides about the same overlap as a threshold-based removal with a cutoff at 0.5 m. To further suffice R1.3 to use minimal manual labor, the threshold-based removal is being favored.

6.2 Summary and Outlook of deep, evidential ISMs

Given the ground-truth occupancy maps and a reference geo IRM, the trained ISM models can be analyzed. Here, deep CNN-based models are favored over other machine learning approaches since they have been proven in literature and in this work to be capable of utilizing large amounts of data, sufficing R1.2 use big data. Also, CNNs

are designed to utilize the spatial coherence in data, which can be seen e.g. in the white box of scene A in Fig. 4-13 where the shape of moving vehicles is being estimated solely on a single detection point and the street layout. Eventually, deep CNNs are capable to be trained on multi-class classification tasks, sufficing R1.6, thus meeting all the preliminary requirements to chose the model.

Given the usage of deep CNNs to model the ISM, an architecture search similar as proposed in [RAD20] is performed to answer RQ3 considering the best architecture for the given requirements for input & ouput dimension R1.5 and the requirement for inference speed R1.4. Even though the effects described in [RAD20] are verified in the experiments leading to a good trade-off between accuracy and speed, the search is only conducted on a small subset of the proposed parameter range in [RAD20]. The reason for the restriction of the search to the small parameter subset is that a network architecture has been identified within a few search steps that already suffices the requirements. Also, the search is only conducted for radar inputs R_1 since this work focuses on radar ISMs and it is the most challenging radar input encoding. However, these restrictions might cause limited performance for other sensor modalities or radar encodings. Thus, it might make sense to revisit the architecture search when specifying the deep ISMs based on other sensors or given other input/outputs dimensions.

Next in Section 4.3, the optimized architecture is used to train three variants of deep ISMs both on R_1 and R_{20} (see definitions in Section 3.3.5) to analyze the influence of added information. The first, referred to as SoftNet, is treated as the baseline using a standard softmax activation and cross entropy loss. The results verify the suggested property of SoftNet to model uncertainty between classes by distributing mass to equal portions, as formulated in H1. In case of uncertainty between free and occupied class, this leads to assigning conflicting mass violating both R1.7 and R1.8. As a solution, DirNet and ShiftNet are analyzed both using a specific mechanism to identify aleatoric uncertainty and moving it to the unknown class. The results show that indeed both methods are capable of solving the problem to a certain extent, better sufficing the requirement of the unknown mass as inverse measure for information (R1.7) and the conflicting mass being an indicator for dynamic objects (R1.8). Here, the ShitNet variant is chosen for further considerations since it provided overall better results around the boundaries of occupied areas and higher free space scores.

Afterwards in Section 4.4, to fully analyze the effects of different radar input encodings, as formulated in RQ6, the ShiftNet results for R_1 and R_{20} inputs from the uncertainty analysis are further compared with the results of a ShiftNet trained on $R_{20|1}$ inputs (see definitions in Section 3.3.5). It can be seen that the scores overall improve in all but the dynamic class by accumulating static detections over time for $R_{20|1}$ as compared to R_1. These scores can additionally be improved by accumulating dynamic detec-

tions over time with a decay on their intensify, as for R_{20}. However, also for R_{20} the quantitative scores suggest an increase of false occupied predictions around a factor of two when comparing R_{20} and $R_{20|1}$ with R_1. One potential cause might be the overall increased amount of occupied mass being estimated for R_{20} and $R_{20|1}$ leading to doubling the amount of true occupied mass and therefore also amplifying the mistakes in other classes. Nevertheless, the qualitative results show a clear improvement of overall quality when using the R_{20} encoding which is why it is chosen for further comparisons.

To further fully answer RQ4 regarding the comparison in performance between deep ISMs trained on different sensor modalities, ShiftNets trained on different camera encodings and the lidar BEV projection are compared in Section 4.5 with the former results for a ShiftNet trained on R_{20}. Regarding the camera encodings, consistent improvement can be seen moving from the homographic RGB BEV projection (C_{RGB}) over the semantic one (C_S) to the projection using MonoDepth and height intensities (C_D) (see further details in Section 3.3.5 and visualizations in Fig. 4-15). Therefore, the C_D encoding is considered for further comparisons. The comparison of the camera encodings might, however, not be fair since some of the transformation for the semantic and MonoDepth inputs is relocated into upstream networks thereby indirectly increasing their respective deep ISM's computational capacity compared to directly using the C_{RGB} as an input. Thus, the deep $IC_{RGB}M$ might be further improved using a bigger network and more data.

Moreover, the MonoDepth-based ShiftNet consistently outperforms the one trained on R_{20} for dynamic and free predictions. However, it fails to capture the occupied areas similarly, showing the improvement the measured depth estimates of the radar provide over the estimated ones of the MonoDepth model. Looking at the deep ILM, these scores are even further improved. Here, the unknown mass also follows the improvements being consistently overall decreased for better performing models. This shows that the ShiftNet model is indeed capable of utilizing improved sensor information to provide better estimates and further verifies that the unknown mass in ShiftNet can be used as an inverse measure for information, as requested in R1.7.

Eventually, Section 4.5 also provides an analysis of the deep ISMs capability to utilize information of fused inputs. Here, the fusion of lidar and camera with the radar inputs respectively is being analyzed by concatenating the inputs channel-wise to analyze RQ5. Again, a consistent improvement can be seen when adding the radar information. Qualitatively, it can be seen that the deep ISM is capable to pick the best predictions from the respective inputs. An example can be seen in Fig. 4-15 scene A, where the fused ISM uses the better vehicle shape predicted with MonoDepth in the left white box while using the radars improved accuracy at the wall in the lower white box. The only exception to the improvement is the increase of false occupied predictions in

the dynamic class. One potential reason might be that in many cases static objects are falsely detected as dynamic (see e.g. orange box for R_{20} of scene B in Fig. 4-11), leading the network to predict an object more often as occupied than dynamic. Here, a further analysis and improvement of the dynamic flag in the radar measurements might improve the predictions for the dynamic class. Furthermore, other means of fusing the inputs by e.g. using completely or partially separate encoders for each sensor modality might lead to further improving the predictions.

Since the deep ISMs analyzed in this work are data driven, they suffer the standard problems of this type of models. One such problem is the generalizability to unseen scenarios. Here, the integration of uncertainty estimates as investigated in Section 4.3 is an attempt to increase the prediction robustness. However, an in-depth discussion on e.g. night or bad weather scenes or measurements of the same modality from sensors not trained on is still pending for future work as they are not well enough covered in the NuScenes dataset to provide statistically relevant evidence.

6.3 Summary and Outlook of deep, evidential ISMs in Occupancy Mapping

For occupancy mapping, it is suggested in H2 that the estimates of deep ISMs contain temporal redundancy which would violate R2.1. Using the proposed Yager's rule of combination, this might lead to an accumulation of low certain estimates up to the point of full convergence. This phenomenon is investigated in Section 5.2 and can be most clearly seen in the occluded areas in Fig. 5-3 for the "deep $IR_{20}M$ accumulated" results. Also, the fusion with the geo IRM shows that its influence is completely overwritten by directly fusing it with the deep IRM, thus, further demonstrating the importance of a solution. To further analyze the cause of the problem, a threshold-based cutoff of low probable estimates is analyzed in the same section. It shows that the accumulation of occupied mass is dampened, proving that the unchecked influence of low certain estimates is indeed one part of the problem.

The solution proposed in Section 3.3.9 uses the difference in unknown mass between the map and the ShiftNet estimates as a measure of mutual information, answering RQ8. This choice is reasonable since the correlation between information and unknown mass is already established in the deep ISM analysis. Moreover, the conflict between the map and the estimates shall be added to the difference in information to react to changes in the static world assumption. The sum of the unknown mass difference and the conflicting mass, thus, provides an answer to RQ10 on how to define the information content for occupancy mapping. Finally, a discount factor is proposed that maps the sum of unknown mass difference and conflict into a suitable interval in a linear way, providing an answer to RQ9. It shall be mentioned that a further analysis

could be beneficial in order to see the influence of varying discount factor definitions to answer RQ9 more thoroughly. Also, alternatives to the proposed approach might be to directly train a deep ISM to only estimate the additional information given a map state or even estimate the next map state directly given different ISM predictions (e.g. network to predict occupancy state with geo and deep ISM estimates as input).

In order to verify the solution proposed in Section 3.3.9, the maps created in Section 5.2 are further compared with maps created using the informational discount operation. Here, the property to limit the deep ISM's influence can be qualitatively as well as quantitatively seen by the unknown areas being less influential. Also, the qualitative verification in Fig. 5-4 shows that the deep ISM fusion only affects regions in which unknown mass can be decreased or conflicting mass occurs. However, by restricting the predictions to maximally accumulate to their own certainty, the overall occupancy estimates become less certain also in correct regions. Here, further work has to be done by e.g. adding an additional term into the ShiftNet optimization that increases the mass estimated or using a transformation that dampens the low probability predictions and amplifies the high probable ones.

Using the redundancy reduced fusion method, occupancy maps are created and compared with deep ISMs based on the best performing sensor encodings as analyzed in Chapter 4 to answer RQ12. The trend of improvement from radar over camera to lidar are also reflected in the mapping results which is to be expected. Again, as for the deep ISMs, it can be seen that the maps with fused sensor modalities correct errors from both sensors respectively. An example is shown in scene A of Fig. 5-6 where the improved shape of the parked cars remains from the camera while the clear separation between the vehicles remains from the radar.

Finally, a method is proposed in Section 3.3.10 to use the deep ISM purely for initialization in the mapping process. Here, the deep ISM's unknown mass is restricted to a lower bound which, together with the redundancy removal of Section 3.3.9, disables the deep ISM's influence once the lower bound has been reached, answering RQ11. Thus, the map is only initially altered by the deep ISM, which meets R2.3, and, afterwards, can only be altered by the geo ISM which suffices R2.4. To not only disable the deep ISM after reaching the lower bound but, additionally, prohibiting it from reducing the map's unknown mass below the lower bound, the discount factor proposed in Section 3.3.9 is altered.

These claims are first verified by simulating the fusion of mass predictions over time for a single occupancy map cell in Section 5.1 qualitatively showing that the lower bound is met and the information content follows the difference in unknown mass and the conflict. Afterwards, the results are verified qualitatively and quantitatively on real-world

data in Section 5.4. It can be seen that the lower bound is met, given a grace interval to account for numerical errors. Also, it can be seen that the proposed approach leads to less falsification of the geo IRM compared to directly fusing it with the redundancy removed deep IRM while it is capable to improve the true rates in the formerly unknown areas (see Fig. 5-8). Example cases where errors of the deep IRM remain in the initialized state and, as such, become distinguishable from the converged areas, verified by the geo IRM, are shown in Fig. 5-9. These cases further demonstrate the motivation and improvements of the proposed approach.

7 Glossary

7.1 List of Symbols

A	enumerator over evidential sets
K	set containing evidence indices $[f, o, u]$ need for occupancy mapping
α_f	positive shape parameter for free class in a Dirichlet distribution
α_k	placeholder either for free or occupied Dirichlet shape parameter
α_o	positive shape parameter for occupied class in a Dirichlet distribution
Beta	Beta distribution
card	cardinality
$[\boldsymbol{e}_x, \boldsymbol{e}_y]$	unit vectors of 2D Cartesian coordinate system
x	distance in $\boldsymbol{e}_x$ direction
y	distance in $\boldsymbol{e}_y$ direction
φ_{center}	center angle of space ray in IDM
D	channel reduction factor of a convolution operation in CNNs
C	amount of channels of a convolution operation in CNNs
d_{det}	dynamic flag of a radar's detection
$\oplus_D$	evidential combination using Dempster's rule
Dir	Dirichlet distribution
$\underline{\gamma}$	evidential discount factor that ensure a lower bound on unknown mass
γ	evidential discount factor
$\otimes$	evidential discount operation
S_e	entropy
$\oplus$	arbitrary evidential combination rule
K	evidential conflict
H	height of a feature in a CNN
W	width of a feature in a CNN
G_a	range-dependant scaling factor of IRM [CLA12]
G_{Clarke}	overall scaling factor of IRM proposed by Clark et al. [CLA12]
G_φ	angle-dependant scaling factor of IRM [CLA12]
G_p	power-dependant scaling factor of IRM [CLA12]
R_g	generalized correlation coefficient
M_D	dynamic weight in the IDM
M_F	free weight in the IDM
M_O	occupied weight in the IDM
I	measure of information
κ	kernel size of a convolution operation in CNNs
$\mathcal{L}_1$	loss function using the absolute error
$\mathcal{L}_2$	loss function using the squared absolute error
$\boldsymbol{z}$	features in CNN before the final output activation

$\mathcal{L}$	loss function
$m(A)$	evidential mass function $A \in 2^U \rightarrow [0, 1]$
$\mathring{m}_d$	dynamic mass function in extended evidential framework
m_d	dynamic evidence mass function
$\mathring{m}_f$	free mass function in extended evidential framework
m_f	evidential mass function for free set
$\mathring{m}_k$	placeholder for dynamic, free, occupied or unknown mass in extended evidential frame
m_k	placeholder either for free, occupied or unknown evidential mass
$\mathring{m}_o$	occupied mass function in extended evidential framework
m_o	evidential mass function for occupied set
m_s	static evidence mass function
$\mathring{m}_u$	unknown mass function in extended evidential framework
$\underline{m_u}$	lower bound on evidential mass function of unknown set
m_u	evidential mass function of unknown set
μ_φ	mean object boundary in angular direction (polar coordinates)
μ_r	mean object boundary in radial direction (polar coordinates)
μ	mean of a Gaussian distribution
$\varphi_\sphericalangle$	opening angle of free space ray in IDM
p_f	free probability
p_k	placeholder either for free or occupied probability
p_o	occupancy probability
φ_{det}	radial distance of a range sensor detection
Pl	plausability
$[\boldsymbol{e}_r, \boldsymbol{e}_\varphi]$	unit vectors of 2D polar coordinate system
φ	angle in $\boldsymbol{e}_\varphi$ direction
r	radial distance in $\boldsymbol{e}_r$ direction
r_{det}	radial distance of a range sensor detection
r_{max}	maximal range of free space ray in IDM
$\mathrm{RecepField}$	receptive field in CNNs
F	evidential free set
O	evidential occupied set
U	evidential unknown set
σ_φ	variance of object boundary in angular direction (polar coordinates)
σ_r	variance of object boundary in radial direction (polar coordinates)
σ	variance of a Gaussian distribution
S_p	specificity
s	stride of a convolution operation in CNNs
Σ_α	sum of positive shape parameters of a Dirichlet distribution
ξ	target in a loss function
T	temporal accumulation horizon for detections in ISMs
$\tilde{}$	estimated quantity

u	upsampling factor of a convolution layer with binlinear upsampling
$\boldsymbol{\alpha}$	vector containing positive shape parameters of a Dirichlet distribution
$\boldsymbol{e}$	evidence vector in SL
$\mathring{\boldsymbol{m}}$	vector of mass functions in extended evidential framework
$\underline{\boldsymbol{m}}$	evidential mass vector with lower-bound on unknown mass
$\boldsymbol{m}$	vector containing the free, occupied and unknown mass functions
$\boldsymbol{p}$	vector containing the free and occupancy probability
$\oplus_{YD}$	evidential combination using the proposed YaDer rule
$\oplus_{Y}$	evidential combination using Yager's rule

7.2 List of Abbreviations

$IDM_{B\text{-}Spline}$	ideal IDM convolved with B-Spline noise model
IDM_{Gauss}	ideal IDM convolved with Gaussian noise model
IDM_{ev}	evidential IDM
IDM_{ideal}	ideal IDM
IDM_{prob}	arbitrary probabilistic IDM
ReLU	Rectified Linear Unit
ADAS	Automated Driver Assistance System
ADF	Automated Driving Function
BatchNorm	batch normalization
BEV	Bird's-Eye-View
C_D	BEV projection of MonoDepth prediction for each camera
C_{RGB}	homography BEV projection of the RGB values for each camera
C_S	homography BEV projection of the semantic segmentation for each camera
CNN	Convolutional Neural Network
conv	convolution
FCN	Fully Convolutional Network
FMCW	Frequency-Modulated Continuous-Wave
FoV	Field of View
GAN	Generative Adversarial Networks
GPS	Global Positioning System
GT	ground-truth occupancy map patch used as labels to train the deep inverse sensor models
IC_DM	Inverse Camera Depth Model
$IC_DR_{20}M$	Inverse Camera Depth and Radar (20) Model
$IC_{RGB}M$	Inverse Camera RGB Model
IC_SM	Inverse Camera Semantic Model

ICM	Inverse Camera Model
IDM	Inverse Detection Model
ILM	Inverse Lidar Model
$ILR_{20}M$	Inverse Lidar and Radar (20) Model
IMU	Inertial Measurement Unit
IoU	Intersection over Union
IRM	Inverse Radar Model
ISM	Inverse Sensor Model
L	BEV projection of lidar detections without ground-plane
MAC	Multiply-Accumulate Operations
MLP	Multilayer Perceptron
MonoDepth	Monocular Depth Estimation
MSE	Mean Squared Error
NN	Neural Networks
PDF	Probability Density Function
R	BEV projection of all radars' detections of the most recent timestep
R_T	BEV projection of all radars' dynamic and static detections of the recent T timesteps
$R_{T\|1}$	same as R_T but only projecting the most recent dynamic detection
RF	Receptive Field
SAE	Society of Automotive Engineers
SemSeg	Semantic Segmentation
SGD	Stochastic Gradient Descent
SL	Subjective Logic
SotA	State-of-the-Art
VAE	Variational Autoencoder

8 Bibliography

8.1 List of References

[ABA16] ABADI, M., BARHAM, P., CHEN, J., CHEN, Z., DAVIS, A., DEAN, J., DEVIN, M., GHEMAWAT, S., IRVING, G., ISARD, M., et al.
Tensorflow: A system for large-scale machine learning
12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16), 2016, pp. 265–283

[APT19] APTIV, AUDI, BAIDU, BMW, CONTINENTAL, DAIMLER, FIAT CHRYSLER AUTOMOB., HERE,INFINEON, INTEL, VOLKSWAGEN
Safety First for Automated Driving
White Paper (2019)

[ARK98] ARKIN, R. C., ARKIN, R. C., et al.
Behavior-based robotics
MIT press, 1998

[ASV15] ASVADI, A., PEIXOTO, P., NUNES, U.
Detection and tracking of moving objects using 2.5 d motion grids
2015 IEEE 18th International Conference on Intelligent Transportation Systems, IEEE, 2015, pp. 788–793

[BAZ18] BAZAREVSKY, V., TKACHENKA, A.
Mobile Real-time Video Segmentation
Google AI Blog (2018)

[BEN09] BENESTY, J., CHEN, J., HUANG, Y., COHEN, I.
Pearson correlation coefficient
Noise reduction in speech processing, Springer, 2009, pp. 1–4

[BER18] BERMAN, M., RANNEN TRIKI, A., BLASCHKO, M. B.
The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks
Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4413–4421

[BOS22] BOSCH MOBILITY SOLUTIONS
Ultrasonic Sensor
2022, URL: https://www.bosch-mobility-solutions.com/en/solutions/sensors/ultrasonic-sensor/

[CAE20] CAESAR, H., BANKITI, V., LANG, A. H., VORA, S., LIONG, V. E., XU, Q., KRISHNAN, A., PAN, Y., BALDAN, G., BEIJBOM, O.
nuscenes: A multimodal dataset for autonomous driving
Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11621–11631

[CAO17] CAO, Y., WU, Z., SHEN, C.
Estimating depth from monocular images as classification using deep fully convolutional residual networks
IEEE Transactions on Circuits and Systems for Video Technology 28.11 (2017), pp. 3174–3182

[CAR15] CARRILLO, H., DAMES, P., KUMAR, V., CASTELLANOS, J. A.
Autonomous robotic exploration using occupancy grid maps and graph slam based on shannon and rényi entropy
2015 IEEE international conference on robotics and automation (ICRA), IEEE, 2015, pp. 487–494

[CAT11] CATTANEO, M. E.
Belief functions combination without the assumption of independence of the information sources
International Journal of Approximate Reasoning 52.3 (2011), pp. 299–315

[CHA19] CHANG, M.-F., LAMBERT, J., SANGKLOY, P., SINGH, J., BAK, S., HARTNETT, A., WANG, D., CARR, P., LUCEY, S., RAMANAN, D., et al.
Argoverse: 3d tracking and forecasting with rich maps
Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 8748–8757

[CHE18] CHEN, L.-C., ZHU, Y., PAPANDREOU, G., SCHROFF, F., ADAM, H.
Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation
ECCV, 2018

[CHO17] CHOLLET, F.
Xception: Deep learning with depthwise separable convolutions
Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1251–1258

[CLA12] CLARKE, B., WORRALL, S., BROOKER, G., NEBOT, E.
Sensor modelling for radar-based occupancy mapping
2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2012, pp. 3047–3054

[CNE18] CNET
Velodyne just made self-driving cars a bit less expensive
2018, URL: https://www.cnet.com/roadshow/news/velodyne-just-made-self-driving-cars-a-bit-less-expensive-hopefully/

[COR16] CORDTS, M., OMRAN, M., RAMOS, S., REHFELD, T., ENZWEILER, M., BENENSON, R., FRANKE, U., ROTH, S., SCHIELE, B.
The cityscapes dataset for semantic urban scene understanding
Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3213–3223

[DAV07] DAVISON, A. J., REID, I. D., MOLTON, N. D., STASSE, O.
MonoSLAM: Real-time single camera SLAM
IEEE transactions on pattern analysis and machine intelligence 29.6 (2007), pp. 1052–1067

[DEM68] DEMPSTER, A. P.
A generalization of Bayesian inference
Journal of the Royal Statistical Society: Series B (Methodological) 30.2 (1968), pp. 205–232

[DIN02] DING, J., WANG, W.-s., ZHAO, Y.-l.
General correlation coefficient between variables based on mutual information
JOURNAL-SICHUAN UNIVERSITY ENGINEERING SCIENCE EDITION 34.3 (2002), pp. 1–5

[EIG14] EIGEN, D., PUHRSCH, C., FERGUS, R.
Depth map prediction from a single image using a multi-scale deep network
Advances in neural information processing systems 27 (2014), pp. 2366–2374

[ELF89] ELFES, A.
Using occupancy grids for mobile robot perception and navigation
Computer 22.6 (1989), pp. 46–57

[FEN19] FENG, D., ROSENBAUM, L., TIMM, F., DIETMAYER, K.
Leveraging heteroscedastic aleatoric uncertainties for robust real-time lidar 3d object detection
2019 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2019, pp. 1280–1287

[FIS81] FISCHLER, M. A., BOLLES, R. C.
Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography
Communications of the ACM 24.6 (1981), pp. 381–395

[FLE07] FLEURET, F., BERCLAZ, J., LENGAGNE, R., FUA, P.
Multicamera people tracking with a probabilistic occupancy map
IEEE transactions on pattern analysis and machine intelligence 30.2 (2007), pp. 267–282

[FOR20] FORD MEDIA CENTER
What Volkswagens Investment in Argo AI Means for Fords Self-Driving Vehicle Business
2020, URL: https://media.ford.com/content/fordmedia/fna/us/en/news/2020/06/02/volkswagen-investment-argo-ai-ford-self-driving.html

[FU18] FU, H., GONG, M., WANG, C., BATMANGHELICH, K., TAO, D.
Deep ordinal regression network for monocular depth estimation

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2002–2011

[GAR16] GARG, R., BG, V. K., CARNEIRO, G., REID, I.
Unsupervised cnn for single view depth estimation: Geometry to the rescue
European conference on computer vision, Springer, 2016, pp. 740–756

[GEI13] GEIGER, A., LENZ, P., STILLER, C., URTASUN, R.
Vision meets robotics: The kitti dataset
The International Journal of Robotics Research 32.11 (2013), pp. 1231–1237

[GOD17] GODARD, C., MAC AODHA, O., BROSTOW, G. J.
Unsupervised monocular depth estimation with left-right consistency
Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 270–279

[GOD19] GODARD, C., MAC AODHA, O., FIRMAN, M., BROSTOW, G. J.
Digging into self-supervised monocular depth estimation
Proceedings of the IEEE international conference on computer vision, 2019, pp. 3828–3838

[GOO16] GOODFELLOW, I., BENGIO, Y., COURVILLE, A., BENGIO, Y.
Deep learning
Vol. 1, 2, MIT press Cambridge, 2016

[GOO20] GOODFELLOW, I., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A., BENGIO, Y.
Generative adversarial networks
Communications of the ACM 63.11 (2020), pp. 139–144

[GUI20] GUIZILINI, V., LI, J., AMBRUS, R., PILLAI, S., GAIDON, A.
Robust Semi-Supervised Monocular Depth Estimation With Reprojected Distances
Conference on Robot Learning, PMLR, 2020, pp. 503–512

[GUO11] GUO, C., SATO, W., HAN, L., MITA, S., MCALLESTER, D.
Graph-based 2D road representation of 3D point clouds for intelligent vehicles
2011 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2011, pp. 715–721

[GUR06] GURALNIK, V., MYLARASWAMY, D., VOGES, H.
On handling dependent evidence and multiple faults in knowledge fusion for engine health management
2006 IEEE aerospace conference, IEEE, 2006, 9–pp

[HAK08] HAKLAY, M., WEBER, P.
Openstreetmap: User-generated street maps
IEEE Pervasive Computing 7.4 (2008), pp. 12–18

[HAN08] HAN, D., HAN, C., YANG, Y.
A modified evidence combination approach based on ambiguity measure

2008 11th International Conference on Information Fusion, IEEE, 2008, pp. 1–6

[21] HDL-32R
Velodyne Lidar, 2021, URL: https://velodynelidar.com/products/hdl-32e/

[HE15] HE, K., ZHANG, X., REN, S., SUN, J.
Delving deep into rectifiers: Surpassing human-level performance on imagenet classification
Proceedings of the IEEE international conference on computer vision, 2015, pp. 1026–1034

[HE16] HE, K., ZHANG, X., REN, S., SUN, J.
Deep residual learning for image recognition
Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778

[HEN20] HENDY, N., SLOAN, C., TIAN, F., DUAN, P., CHARCHUT, N., XIE, Y., WANG, C., PHILBIN, J.
FISHING Net: Future Inference of Semantic Heatmaps In Grids
arXiv preprint arXiv:2006.09917 (2020)

[HOR91] HORNIK, K.
Approximation capabilities of multilayer feedforward networks
Neural networks 4.2 (1991), pp. 251–257

[HOU62] HOUGH, P. V.
Method and means for recognizing complex patterns
US Patent 3,069,654, Dec. 1962

[IOF15] IOFFE, S., SZEGEDY, C.
Batch normalization: Accelerating deep network training by reducing internal covariate shift
International conference on machine learning, PMLR, 2015, pp. 448–456

[JAD15] JADERBERG, M., SIMONYAN, K., ZISSERMAN, A., et al.
Spatial transformer networks
Advances in neural information processing systems 28 (2015), pp. 2017–2025

[JIA09] JIANG, H.-N., XU, X.-B., WEN, C.-L.
The combination method for dependent evidence and its application for simultaneous faults diagnosis
2009 International Conference on Wavelet Analysis and Pattern Recognition, IEEE, 2009, pp. 496–501

[JIN19] JING, Y., YANG, Y., FENG, Z., YE, J., YU, Y., SONG, M.
Neural style transfer: A review
IEEE transactions on visualization and computer graphics (2019)

[JOS18] JOSANG, A.
Subjective Logic: A formalism for reasoning under uncertainty
Springer, 2018

[KEN17] KENDALL, A., GAL, Y.
What uncertainties do we need in bayesian deep learning for computer vision?
Advances in neural information processing systems, 2017, pp. 5574–5584

[KIN14] KINGMA, D. P., BA, J.
Adam: A method for stochastic optimization
arXiv preprint arXiv:1412.6980 (2014)

[KIN13] KINGMA, D. P., WELLING, M.
Auto-encoding variational bayes
arXiv preprint arXiv:1312.6114 (2013)

[KUL51] KULLBACK, S., LEIBLER, R. A.
On information and sufficiency
The annals of mathematical statistics 22.1 (1951), pp. 79–86

[KUR12] KURDEJ, M., MORAS, J., CHERFAOUI, V., BONNIFAIT, P.
Map-aided fusion using evidential grids for mobile perception in urban environment
Belief Functions: Theory and Applications, Springer, 2012, pp. 343–350

[KUZ17] KUZNIETSOV, Y., STUCKLER, J., LEIBE, B.
Semi-supervised deep learning for monocular depth map prediction
Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 6647–6655

[LAI16] LAINA, I., RUPPRECHT, C., BELAGIANNIS, V., TOMBARI, F., NAVAB, N.
Deeper depth prediction with fully convolutional residual networks
2016 Fourth international conference on 3D vision (3DV), IEEE, 2016, pp. 239–248

[LI15] LI, B., SHEN, C., DAI, Y., VAN DEN HENGEL, A., HE, M.
Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs
Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1119–1127

[LI14] LI, Q., ZHANG, L., MAO, Q., ZOU, Q., ZHANG, P., FENG, S., OCHIENG, W.
Motion field estimation for a dynamic scene using a 3D LiDAR
Sensors 14.9 (2014), pp. 16672–16691

[LIA18] LIANG, M., YANG, B., WANG, S., URTASUN, R.
Deep continuous fusion for multi-sensor 3d object detection
Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 641–656

[LIU15] LIU, F., SHEN, C., LIN, G.
Deep convolutional neural fields for depth estimation from a single image
Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 5162–5170

[LOM17] LOMBACHER, J., LAUDT, K., HAHN, M., DICKMANN, J., WÖHLER, C.
Semantic radar grids
2017 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2017, pp. 1170–1175

[LON81] LONGUET-HIGGINS, H. C.
A computer algorithm for reconstructing a scene from two projections
Nature 293.5828 (1981), pp. 133–135

[LOO16] LOOP, C., CAI, Q., ORTS-ESCOLANO, S., CHOU, P. A.
A closed-form Bayesian fusion equation using occupancy probabilities
2016 Fourth International Conference on 3D Vision (3DV), IEEE, 2016, pp. 380–388

[LU19] LU, C., MOLENGRAFT, M. J. G. van de, DUBBELMAN, G.
Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks
IEEE Robotics and Automation Letters 4.2 (2019), pp. 445–452

[MA20] MA, J., JIANG, X., FAN, A., JIANG, J., YAN, J.
Image matching from handcrafted to deep features: A survey
International Journal of Computer Vision (2020), pp. 1–57

[MA19] MA, Y., ANDERSON, J., CROUCH, S., SHAN, J.
Moving object detection and tracking with doppler LiDAR
Remote Sensing 11.10 (2019), p. 1154

[MAA13] MAAS, A. L., HANNUN, A. Y., NG, A. Y., et al.
Rectifier nonlinearities improve neural network acoustic models
Proc. icml, vol. 30, 1, Citeseer, 2013, p. 3

[MAL91] MALLOT, H. A., BÜLTHOFF, H. H., LITTLE, J., BOHRER, S.
Inverse perspective mapping simplifies optical flow computation and obstacle detection
Biological cybernetics 64.3 (1991), pp. 177–185

[MAN20] MANI, K., DAGA, S., GARG, S., NARASIMHAN, S. S., KRISHNA, M., JATAVALLABHULA, K. M.
MonoLayout: Amodal scene layout from a single image
The IEEE Winter Conference on Applications of Computer Vision, 2020, pp. 1689–1697

[MOR11] MORAS, J., CHERFAOUI, V., BONNIFAIT, P.
Moving objects detection by conflict analysis in evidential grids
2011 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2011, pp. 1122–1127

[MOU17] MOUHAGIR, H., CHERFAOUI, V., TALJ, R., AIOUN, F., GUILLEMARD, F.
Using evidential occupancy grid for vehicle trajectory planning under uncertainty with tentacles
2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2017, pp. 1–7

[NN17] N.N.
Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles
On-Road Automated Vehicle Standards Committee (2017)

[NAR18] NARKSRI, P., TAKEUCHI, E., NINOMIYA, Y., MORALES, Y., AKAI, N., KAWAGUCHI, N.
A slope-robust cascaded ground segmentation in 3D point cloud for autonomous vehicles
2018 21st International Conference on intelligent transportation systems (ITSC), IEEE, 2018, pp. 497–504

[ODE16] ODENA, A., DUMOULIN, V., OLAH, C.
Deconvolution and checkerboard artifacts
Distill 1.10 (2016), e3

[OLI16] OLIVEIRA, M., SANTOS, V., SAPPA, A. D., DIAS, P.
Scene representations for autonomous driving: an approach based on polygonal primitives
Robot 2015: Second Iberian Robotics Conference, Springer, 2016, pp. 503–515

[ORG18] ORGANIZATION, W. H. et al.
Global status report on road safety 2018: summary
Tech. rep., World Health Organization, 2018

[PAG96] PAGAC, D., NEBOT, E. M., DURRANT-WHYTE, H.
An evidential approach to probabilistic map-building
Proceedings of IEEE International Conference on Robotics and Automation, vol. 1, IEEE, 1996, pp. 745–750

[PAN20] PAN, B., SUN, J., LEUNG, H. Y. T., ANDONIAN, A., ZHOU, B.
Cross-view semantic segmentation for sensing surroundings
IEEE Robotics and Automation Letters 5.3 (2020), pp. 4867–4873

[PHI20] PHILION, J., FIDLER, S.
Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d
European Conference on Computer Vision, Springer, 2020, pp. 194–210

[PRO19] PROPHET, R., LI, G., STURM, C., VOSSIEK, M.
Semantic Segmentation on Automotive Radar Maps
2019 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2019, pp. 756–763

[PRO18] PROPHET, R., STARK, H., HOFFMANN, M., STURM, C., VOSSIEK, M.
Adaptions for automotive radar based occupancy gridmaps
2018 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM), IEEE, 2018, pp. 1–4

[RAD20] RADOSAVOVIC, I., KOSARAJU, R. P., GIRSHICK, R., HE, K., DOLLÁR, P.
Designing network design spaces
Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10428–10436

[REI20] REIHER, L., LAMPE, B., ECKSTEIN, L.
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
arXiv preprint arXiv:2005.04078 (2020)

[REI13] REINEKING, T., CLEMENS, J.
Evidential FastSLAM for grid mapping
Proceedings of the 16th International Conference on Information Fusion, IEEE, 2013, pp. 789–796

[ROD20] RODDICK, T., CIPOLLA, R.
Predicting Semantic Map Representations from Images using Pyramid Occupancy Networks
Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11138–11147

[RON15] RONNEBERGER, O., FISCHER, P., BROX, T.
U-net: Convolutional networks for biomedical image segmentation
International Conference on Medical image computing and computer-assisted intervention, Springer, 2015, pp. 234–241

[RUM17] RUMMELHARD, L., PAIGWAR, A., NÈGRE, A., LAUGIER, C.
Ground estimation and point cloud segmentation using SpatioTemporal Conditional Random Field
2017 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2017, pp. 1105–1110

[SAP18] SAPUTRA, M. R. U., MARKHAM, A., TRIGONI, N.
Visual SLAM and structure from motion in dynamic environments: A survey
ACM Computing Surveys (CSUR) 51.2 (2018), pp. 1–36

[SCH18] SCHULTER, S., ZHAI, M., JACOBS, N., CHANDRAKER, M.
Learning to look around objects for top-view representations of outdoor scenes
Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 787–802

[SEN18] SENSOY, M., KAPLAN, L., KANDEMIR, M.
Evidential deep learning to quantify classification uncertainty
arXiv preprint arXiv:1806.01768 (2018)

[SEV21] SEVERINO, A., CURTO, S., BARBERI, S., ARENA, F., PAU, G.
Autonomous Vehicles: An Analysis Both on Their Distinctiveness and the Potential Impact on Urban Transport Systems
Applied Sciences 11.8 (2021), p. 3604

[SHA76] SHAFER, G.
A mathematical theory of evidence
Vol. 42, Princeton university press, 1976

[SHI17] SHI, F., SU, X., QIAN, H., YANG, N., HAN, W.
Research on the fusion of dependent evidence based on rank correlation coefficient
Sensors 17.10 (2017), p. 2362

[SIM14] SIMONYAN, K., ZISSERMAN, A.
Very deep convolutional networks for large-scale image recognition
arXiv preprint arXiv:1409.1556 (2014)

[SLE19] SLESS, L., EL SHLOMO, B., COHEN, G., ORON, S.
Road Scene Understanding by Occupancy Grid Learning from Sparse Radar Clusters using Semantic Segmentation
Proceedings of the IEEE International Conference on Computer Vision Workshops, 2019, pp. 0–0

[SLU19] SLUTSKY, M., DOBKIN, D.
Dual inverse sensor model for radar occupancy grids
2019 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2019, pp. 1760–1767

[SRI14] SRIVASTAVA, N., HINTON, G., KRIZHEVSKY, A., SUTSKEVER, I., SALAKHUTDINOV, R.
Dropout: a simple way to prevent neural networks from overfitting
The journal of machine learning research 15.1 (2014), pp. 1929–1958

[SU18] SU, X., LI, L., SHI, F., QIAN, H.
Research on the fusion of dependent evidence based on mutual information
IEEE Access 6 (2018), pp. 71839–71845

[SU15] SU, X., MAHADEVAN, S., XU, P., DENG, Y.
Handling of dependence in Dempster–Shafer theory
International Journal of Intelligent Systems 30.4 (2015), pp. 441–467

[SUN20] SUN, P., KRETZSCHMAR, H., DOTIWALLA, X., CHOUARD, A., PATNAIK, V., TSUI, P., GUO, J., ZHOU, Y., CHAI, Y., CAINE, B., et al.
Scalability in perception for autonomous driving: Waymo open dataset

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 2446–2454

[TAN19] TAN, M., LE, Q.
Efficientnet: Rethinking model scaling for convolutional neural networks
International Conference on Machine Learning, PMLR, 2019, pp. 6105–6114

[THR06] THRUN, S., MONTEMERLO, M., DAHLKAMP, H., STAVENS, D., ARON, A., DIEBEL, J., FONG, P., GALE, J., HALPENNY, M., HOFFMANN, G., et al.
Stanley: The robot that won the DARPA Grand Challenge
Journal of field Robotics 23.9 (2006), pp. 661–692

[THR93] THRUN, S. B.
Exploration and model building in mobile robot domains
IEEE international conference on neural networks, IEEE, 1993, pp. 175–180

[TIA20] TIAN, Y., SONG, W., CHEN, L., SUNG, Y., KWAK, J., SUN, S.
Fast planar detection system using a GPU-based 3D Hough transform for LiDAR point clouds
Applied Sciences 10.5 (2020), p. 1744

[UMM17] UMMENHOFER, B., ZHOU, H., UHRIG, J., MAYER, N., ILG, E., DOSOVITSKIY, A., BROX, T.
Demon: Depth and motion network for learning monocular stereo
Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5038–5047

[URM07] URMSON, C., ANHALT, J., BAGNELL, J. A. (, BAKER, C. R., BITTNER, R. E., DOLAN, J. M., DUGGINS, D., FERGUSON, D., GALATALI, T., GEYER, H., GITTLEMAN, M., HARBAUGH, S., HEBERT, M., HOWARD, T., KELLY, A., KOHANBASH, D., LIKHACHEV, M., MILLER, N., PETERSON, K., RAJKUMAR, R., RYBSKI, P., SALESKY, B., SCHERER, S., SEO, Y.-W., SIMMONS, R., SINGH, S., SNIDER, J. M., STENTZ, A. (, WHITTAKER, W. (L., ZIGLAR, J.
Tartan Racing: A Multi-Modal Approach to the DARPA Urban Challenge
Tech. rep., Pittsburgh, PA: Carnegie Mellon University, Apr. 2007

[VAN95] VAN DAM, J. W., KRÖSE, B. J., GROEN, F. C.
Neural network applications in sensor fusion for an autonomous mobile robot
International Workshop on Reasoning with Uncertainty in Robotics, Springer, 1995, pp. 263–278

[VAS17] VASWANI, A., SHAZEER, N., PARMAR, N., USZKOREIT, J., JONES, L., GOMEZ, A. N., KAISER, ., POLOSUKHIN, I.

Attention is all you need
Advances in neural information processing systems, 2017, pp. 5998–6008

[VEL18a] VELAS, M., SPANEL, M., HRADIS, M., HEROUT, A.
Cnn for very fast ground segmentation in velodyne lidar data
2018 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), IEEE, 2018, pp. 97–103

[VEL18b] VELODYNE LIDAR
Velodyne Slashes the Price in Half of Its Most Popular LiDAR Sensor
2018, URL: https://velodynelidar.com/press-release/velodyne-slashes-the-price-in-half-of-its-most-popular-lidar-sensor/

[VER19] VERDOJA, F., LUNDELL, J., KYRKI, V.
Deep Network Uncertainty Maps for Indoor Navigation
2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids), IEEE, 2019, pp. 112–119

[WER15] WERBER, K., RAPP, M., KLAPPSTEIN, J., HAHN, M., DICKMANN, J., DIETMAYER, K., WALDSCHMIDT, C.
Automotive radar gridmap representations
2015 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM), IEEE, 2015, pp. 1–4

[WES19] WESTON, R., CEN, S., NEWMAN, P., POSNER, I.
Probably unknown: Deep inverse sensor modelling radar
2019 International Conference on Robotics and Automation (ICRA), IEEE, 2019, pp. 5446–5452

[WIR18] WIRGES, S., STILLER, C., HARTENBACH, F.
Evidential occupancy grid map augmentation using deep learning
2018 IEEE intelligent vehicles symposium (IV), IEEE, 2018, pp. 668–673

[WU19] WU, Q., LI, H., LI, L., YU, Z.
Quantifying intrinsic uncertainty in classification via deep dirichlet mixture networks
arXiv preprint arXiv:1906.04450 (2019)

[WUL18] WULFF, F., SCHÄUFELE, B., SAWADE, O., BECKER, D., HENKE, B., RADUSCH, I.
Early fusion of camera and lidar for robust road detection based on U-Net FCN
2018 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2018, pp. 1426–1431

[XU17] XU, H., DENG, Y.
Dependent evidence combination based on shearman coefficient and pearson coefficient
IEEE Access 6 (2017), pp. 11634–11640

[YAG87] YAGER, R. R.
On the Dempster-Shafer framework and new combination rules
Information sciences 41.2 (1987), pp. 93–137

[YAG09] YAGER, R. R.
On the fusion of non-independent belief structures
International journal of general systems 38.5 (2009), pp. 505–531

[YAN13] YANG, J.-B., XU, D.-L.
Evidential reasoning rule for evidence combination
Artificial Intelligence 205 (2013), pp. 1–29

[YAN20] YANG, N., STUMBERG, L. v., WANG, R., CREMERS, D.
D3VO: Deep Depth, Deep Pose and Deep Uncertainty for Monocular Visual Odometry
Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 1281–1292

[YU15] YU, C., CHERFAOUI, V., BONNIFAIT, P.
Evidential occupancy grid mapping with stereo-vision
2015 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2015, pp. 712–717

[ZAD79] ZADEH, L.
On the validity of Dempsters rule of combination, Memo M 79/24
Univ. of California, Berkeley 74 (1979)

[ZHA14] ZHANG, J., SINGH, S.
LOAM: Lidar Odometry and Mapping in Real-time.
Robotics: Science and Systems, vol. 2, 9, 2014

[ZHA20a] ZHANG, M., CHAI, H., SONG, J., JALLER, M., RODIER, C.
The Impacts of Automated Vehicles on Center City Parking Demand
Tech. rep., University of California, Davis, 2020

[ZHA20b] ZHANG, P., TIAN, Y., KANG, B.
A new synthesis combination rule based on evidential correlation coefficient
IEEE Access 8 (2020), pp. 39898–39906

[ZHO17] ZHOU, T., BROWN, M., SNAVELY, N., LOWE, D. G.
Unsupervised learning of depth and ego-motion from video
Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1851–1858

[ZHO14] ZHOU, Z.-H., CHAWLA, N. V., JIN, Y., WILLIAMS, G. J.
Big data opportunities and challenges: Discussions from data analytics perspectives [discussion forum]
IEEE Computational intelligence magazine 9.4 (2014), pp. 62–74

8.2 List of Publications in Relation to this Thesis

[BAU19a] BAUER, D., KUHNERT, L., ECKSTEIN, L.
Deep, spatially coherent inverse sensor models with uncertainty incorporation using the evidential framework
2019 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2019, pp. 2490–2495

[BAU19b] BAUER, D., KUHNERT, L., ECKSTEIN, L.
Deep, spatially coherent Occupancy Maps based on Radar Measurements
AmE 2019-Automotive meets Electronics; 10th GMM-Symposium, VDE, 2019, pp. 1–6

[BAU20] BAUER, D., KUHNERT, L., ECKSTEIN, L.
Deep inverse sensor models as priors for evidential occupancy mapping
2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2020, pp. 6032–3067

1 Appendix

1.1 NuScenes Data Split

This chapter shall provide an overview of the real-world data used during training, validation and testing. As described in Section 3.2.4, the NuScenes dataset is chosen to conduct the experiments in this work with the restriction that not all scenes are reasonable in this work's setting. The scenes which sufficed the requirements can be found in the following figures.

Train Scenes Part 1 (from "NuScenes Detection")
0001, 0002, 0041, 0042, 0043, 0044, 0045, 0046, 0047, 0048, 0049, 0050, 0051, 0052, 0053, 0054, 0055, 0056, 0057, 0058, 0059, 0060, 0061, 0062, 0063, 0064, 0065, 0066, 0067, 0068, 0069, 0070, 0071, 0072, 0073, 0074, 0075, 0076, 0161, 0162, 0163, 0164, 0165, 0166, 0167, 0168, 0170, 0171, 0172, 0173, 0174, 0175, 0176, 0190, 0191, 0192, 0193, 0194, 0195, 0196, 0199, 0200, 0202, 0203, 0204, 0206, 0207, 0208, 0209, 0210, 0211, 0212, 0213, 0214, 0254, 0255, 0256, 0257, 0258, 0259, 0260, 0261, 0262, 0263, 0264, 0283, 0284, 0285, 0286, 0287, 0288, 0289, 0290, 0291, 0292, 0293, 0294, 0295, 0296, 0297, 0298, 0299, 0300, 0301, 0302, 0303, 0304, 0305, 0306, 0315, 0316, 0317, 0318, 0321, 0323, 0324, 0347, 0348, 0349, 0350, 0351, 0352, 0353, 0354, 0355, 0356, 0357, 0358, 0359, 0360, 0361, 0362, 0363, 0364, 0365, 0366, 0367, 0368, 0369, 0370, 0371, 0372, 0373, 0374, 0375, 0382, 0420, 0421, 0422, 0423, 0424, 0425, 0426, 0427, 0428, 0429, 0430, 0431, 0432, 0433, 0434, 0435, 0436, 0437, 0438, 0439, 0457, 0458, 0459, 0461, 0462, 0463, 0464, 0465, 0467, 0468, 0469, 0471, 0472, 0474, 0475, 0476, 0477, 0478, 0479, 0480, 0566, 0568, 0570, 0571, 0572, 0573, 0574, 0575, 0576, 0577, 0578, 0580, 0582, 0583, 0665, 0666, 0667, 0668, 0669, 0670, 0671, 0672, 0673, 0674, 0675, 0676, 0677, 0678, 0679, 0681, 0683, 0684, 0685, 0686, 0687, 0688, 0689, 0739, 0740, 0741, 0744, 0746, 0747, 0749, 0750, 0751, 0752, 0757, 0758, 0759, 0760, 0761, 0762, 0763, 0764, 0765, 0767, 0768, 0769, 0868, 0869, 0870, 0871, 0872, 0873, 0875, 0876, 0877, 0878, 0880, 0882, 0883, 0884, 0885, 0886, 0887, 0888, 0889, 0890, 0891, 0892, 0893, 0894, 0895, 0896, 0897, 0898, 0899, 0900, 0901, 0902, 0903, 0945, 0947, 0949, 0952, 0953, 0955, 0956, 0957, 0958, 0959, 0960, 0961, 0975, 0976, 0977, 0978, 0979, 0980, 0981, 0982, 0983, 0984, 0988, 0989, 0990, 0991, 1011, 1012, 1013, 1014, 1015, 1016, 1017, 1018, 1019, 1020, 1021, 1022, 1023, 1024, 1025, 1074, 1075, 1076, 1077, 1078, 1079, 1080, 1081, 1082, 1083, 1084, 1085, 1086, 1087, 1088, 1089, 1090, 1091, 1092, 1093, 1094, 1095, 1096, 1097, 1098, 1099, 1100, 1101, 1102, 1104, 1105

Fig. 1-1: First part of the scenes used for training from the from the NuScenes detection dataset.

Train Scenes Part 2 (from "NuScenes Tracking")
0004, 0005, 0006, 0007, 0008, 0009, 0010, 0011, 0019, 0020, 0021, 0022, 0023, 0024, 0025, 0026, 0027, 0028, 0029, 0030, 0031, 0032, 0033, 0034, 0120, 0121, 0122, 0123, 0124, 0125, 0126, 0127, 0128, 0129, 0130, 0131, 0132, 0133, 0134, 0135, 0138, 0139, 0149, 0150, 0151, 0152, 0154, 0155, 0157, 0158, 0159, 0160, 0177, 0178, 0179, 0180, 0181, 0182, 0183, 0184, 0185, 0187, 0188, 0218, 0219, 0220, 0222, 0224, 0225, 0226, 0227, 0228, 0229, 0230, 0231, 0232, 0233, 0234, 0235, 0236, 0237, 0238, 0239, 0240, 0241, 0242, 0243, 0244, 0245, 0246, 0247, 0248, 0249, 0250, 0251, 0252, 0253, 0328, 0376, 0377, 0378, 0379, 0380, 0381, 0383, 0384, 0385, 0386, 0388, 0389, 0390, 0391, 0392, 0393, 0394, 0395, 0396, 0397, 0398, 0399, 0400, 0401, 0402, 0403, 0405, 0406, 0407, 0408, 0410, 0411, 0412, 0413, 0414, 0415, 0416, 0417, 0418, 0419, 0440, 0441, 0442, 0443, 0444, 0445, 0446, 0447, 0448, 0449, 0450, 0451, 0452, 0453, 0454, 0455, 0456, 0499, 0500, 0501, 0502, 0504, 0505, 0506, 0507, 0508, 0509, 0510, 0511, 0512, 0513, 0514, 0515, 0517, 0518, 0525, 0526, 0527, 0528, 0529, 0530, 0531, 0532, 0533, 0534, 0535, 0536, 0537, 0538, 0539, 0541, 0542, 0543, 0544, 0545, 0546, 0584, 0585, 0586, 0587, 0588, 0589, 0590, 0591, 0592, 0593, 0594, 0595, 0596, 0597, 0598, 0599, 0600, 0639, 0640, 0641, 0642, 0643, 0644, 0645, 0646, 0647, 0648, 0649, 0650, 0651, 0652, 0653, 0654, 0655, 0656, 0657, 0658, 0659, 0660, 0661, 0662, 0663, 0664, 0695, 0696, 0697, 0698, 0700, 0701, 0703, 0704, 0705, 0706, 0707, 0708, 0709, 0710, 0711, 0712, 0713, 0714, 0715, 0716, 0717, 0718, 0719, 0726, 0727, 0728, 0730, 0731, 0733, 0734, 0735, 0736, 0737, 0738, 0786, 0787, 0789, 0790, 0791, 0792, 0803, 0804, 0805, 0806, 0808, 0809, 0810, 0811, 0812, 0813, 0815, 0816, 0817, 0819, 0820, 0821, 0822, 0847, 0848, 0849, 0850, 0851, 0852, 0853, 0854, 0855, 0856, 0858, 0860, 0861, 0862, 0863, 0864, 0865, 0866, 0992, 0994, 0995, 0996, 0997, 0998, 0999, 1000, 1001, 1002, 1003, 1004, 1005, 1006, 1007, 1008, 1009, 1010, 1044, 1045, 1046, 1047, 1048, 1049, 1050, 1051, 1052, 1053, 1054, 1055, 1056, 1057, 1058, 1106, 1107, 1108, 1109, 1110

Fig. 1-2: Second part of the scenes used for training from the from the NuScenes tracking dataset.

Validation Scenes
0003, 0012, 0013, 0014, 0015, 0016, 0017, 0018, 0035, 0036, 0038, 0039, 0092, 0093, 0094, 0095, 0096, 0097, 0098, 0099, 0100, 0101, 0102, 0103, 0104, 0105, 0106, 0107, 0108, 0109, 0110, 0221, 0268, 0269, 0270, 0271, 0272, 0273, 0274, 0275, 0276, 0277, 0278, 0329, 0330, 0331, 0332, 0344, 0345, 0346, 0519, 0520, 0521, 0522, 0523, 0524, 0552, 0553, 0554, 0555, 0556, 0557, 0558, 0559, 0560, 0561, 0562, 0563, 0564, 0565, 0625, 0626, 0627, 0629, 0630, 0632, 0633, 0634, 0635, 0636, 0637, 0638, 0770, 0771, 0775, 0777, 0778, 0780, 0781, 0782, 0783, 0784, 0794, 0795, 0796, 0797, 0798, 0799, 0800, 0802, 0904, 0905, 0906, 0907, 0908, 0909, 0910, 0911, 0912, 0913, 0914, 0915, 0916, 0917, 0919, 0920, 0921, 0922, 0923, 0924, 0925, 0926, 0927, 0928, 0929, 0930, 0931, 0962, 0963, 0966, 0967, 0968, 0969, 0971, 0972, 1059, 1060, 1061, 1062, 1063, 1064, 1065, 1066, 1067, 1068, 1069, 1070, 1071, 1072, 1073

Fig. 1-3: The scenes from the NuScenes dataset used for validation.

Test Scenes
0077, 0078, 0079, 0080, 0081, 0082, 0083, 0084, 0085, 0086, 0087, 0088, 0089, 0090, 0091, 0111, 0112, 0113, 0114, 0115, 0116, 0117, 0118, 0119, 0140, 0142, 0143, 0144, 0145, 0146, 0147, 0148, 0265, 0266, 0279, 0280, 0281, 0282, 0307, 0308, 0309, 0310, 0311, 0312, 0313, 0314, 0333, 0334, 0335, 0336, 0337, 0338, 0339, 0340, 0341, 0342, 0343, 0481, 0482, 0483, 0484, 0485, 0486, 0487, 0488, 0489, 0490, 0491, 0492, 0493, 0494, 0495, 0496, 0497, 0498, 0547, 0548, 0549, 0550, 0551, 0601, 0602, 0603, 0604, 0606, 0607, 0608, 0609, 0610, 0611, 0612, 0613, 0614, 0615, 0616, 0617, 0618, 0619, 0620, 0621, 0622, 0623, 0624, 0827, 0828, 0829, 0830, 0831, 0833, 0834, 0835, 0836, 0837, 0838, 0839, 0840, 0841, 0842, 0844, 0845, 0846, 0932, 0933, 0935, 0936, 0937, 0938, 0939, 0940, 0941, 0942, 0943, 1026, 1027, 1028, 1029, 1030, 1031, 1032, 1033, 1034, 1035, 1036, 1037, 1038, 1039, 1040, 1041, 1042, 1043

Fig. 1-4: The scenes from the NuScenes dataset used for testing.