Deep inverse Radar Models as Priors for evidential Occupancy Mapping

Tiefe inverse Radarmodelle as A-priori-Information in evidenzbasierten Belegungskarten

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

Daniel Bauer

IMPRESSUM

WIRD VOM INSTITUT GESTALTET

Preface

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

Hier eventuell eigenen Text.

Aachen, im Januar 2033

Max Mustermann

Contents

1 State of the Art 3

1.1 Occupancy Mapping 3

1.2 Geometric Inverse Sensor Models 4

1.2.1 Ray-Casting in geo ISMs 4

1.2.2 Geo ISMs for Lidars 7

1.2.3 Geo ISMs for Cameras 8

1.2.4 Geo ISMs for Radars 10

1.3 Deep Inverse Sensor Models 11

1.3.1 Deep ISM Architecture 11

1.3.2 Deep ISMs for Cameras 13

1.3.3 Deep ISMs for Range Sensors 14

1.3.4 Fusion of Sensor Modalities in deep ISMs 15

1.4 Evidential Combination Rules 16

1.4.1 Combination of independent Evidence 16

1.4.2 Combination of dependent Evidence 17

1.4.3 Combination of Evidence during Training 18

2 Research Approach 21

2.1 Requirements 21

2.1.1 Requirements for deep, evidential ISMs 21

2.1.2 Requirements for Usage of deep, evidential ISMs as Priors in Occupancy Mapping 22

2.2 Research Needs 23

2.2.1 Research Needs for deep, evidential Inverse Sensor Models 23

2.2.2 Research Needs for Usage of deep, evidential ISMs as Priors in Occupancy Mapping 25

2.3 Overview of Methodology 27

2.3.1 Definition of geometric ISMs for Lidars and Radars 28

2.3.2 Methodology to define the Ground-Truth 29

2.3.3 Methodology to define the deep ISM Architecture ... 30
2.3.4 Methodology to account for aleatoric Uncertainties in deep ISMs ... 30
2.3.5 Methodology to use deep Inverse Sensor Model (ISM)s in Occupancy Mapping ... 33
2.3.6 Methodology to use deep ISMs as Priors in Occupancy Mapping ... 35
3 Deep ISM Experiments ... 39
3.1 Choice of Ground-Truth ... 39
3.1.1 Experimental Setup ... 40
3.1.2 Experimental Results ... 41
3.2 Deep ISM Dataset ... 42
3.2.1 Deep ISM Inputs ... 42
3.2.2 Deep ISM Targets ... 43
3.3 Choice of UNet Architecture ... 44
3.3.1 Experimental Setup ... 44
3.3.2 Experimental Results ... 45
3.4 Aleatoric Uncertainties in deep ISMs ... 46
3.4.1 Experimental Results ... 46
3.5 Camera-Radar Fusion in deep ISMs ... 48
3.5.1 Experimental Results ... 48
3.6 Lidar-Radar Fusion deep ISMs ... 49
3.6.1 Experimental Results ... 49
4 Bibliography ... 53
5 Publications ... 65

1 State of the Art

This chapter contains the current state-of-the-art in fields relevant for this thesis.

1.1 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 by [ELF89], the surrounding is stored as a bird's-eye-view (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

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

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

$$\boldsymbol{p} = \begin{bmatrix} p_f & p_o \end{bmatrix} \qquad \text{Eq. 1-3}$$

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

$$2^U = \{\emptyset, F, O, U\}, \qquad U = \{F, O\} \qquad \text{Eq. 1-4}$$

consisting of the empty $\emptyset$, free F, occupied O and unknown set U. Additionally, mass functions $m(A)$ 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

$$m(\emptyset) = 0 \qquad \text{Eq. 1-5}$$

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

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

$$\boldsymbol{m} = \begin{bmatrix} m(F), & m(O), & m(U) \end{bmatrix}^\mathsf{T} = \begin{bmatrix} m_f, & m_o, & m_u \end{bmatrix}^\mathsf{T} \qquad \text{Eq. 1-7}$$

The evidential formulation adds and additional degree of freedom by modeling the unknown class separately. This allows 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.

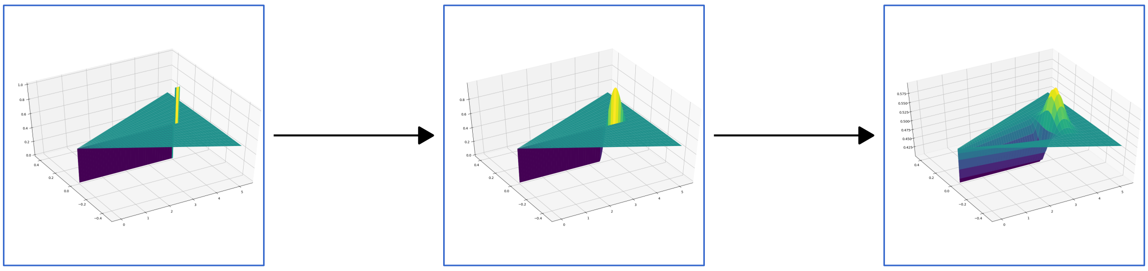
To build such occupancy maps, sensor measurements are being fed into so called ISMs to obtain an estimate of the occupancy state around the vehicle. This estimate is then transformed into map coordinates and fused into the map using evidential combination rules. To obtain unbiased maps in which each 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.

1.2 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. 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 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 [**zhou2014big**] and are universal function approximators [HOR91].

1.2.1 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 who's true mean is defined in polar coordinates as (μ_r, μ_ϕ) while its measurement is defined as $(\tilde{\mu_r}, \tilde{\mu_\phi})$ with the estimated measurement variance $(\tilde{\sigma}_r, \tilde{\sigma}_\phi)$.
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, marking the regions crossed by the ray with the ideal Inverse Detection Model (IDM). This ideal IDM defines everything along the distance r between the true object boundary μ_r and the sensor as free, the position of the object boundary as occupied



a) b) c)

Fig. 1-1: Visualization of a) the ideal IDM 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 ideal IDM.

and everything else as unknown. Given the true object angle μ_ϕ, this can be written as follows

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

Next, since the radial measurement $z = (\tilde{\mu}_r, \tilde{\mu}_\phi)$ is in fact not ideal but contains uncertainty, the ideal IDM is convoluted with a radial and angular Gaussian noise model. Here, 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. 1-1.

In [LOO16], Loop et al. provide a formulation of the IDM's radial component along μ_ϕ using the error function Erf$(x) = \frac{2}{\pi} \int_0^x e^{-t^2} dt$ as follows

$$\begin{aligned} \text{IDM}_{\text{Gauss}}(r, \tilde{\mu}_r, \sigma_r, \phi = \mu_\phi) = P(p_o(r, \phi = \mu_\phi) = 1|\tilde{\mu}_r) =& \frac{1}{2}\text{Erf}\left(\frac{\tilde{\mu}_r - r}{\sqrt{2}\sigma_r}\right) \\ &- \frac{1}{4}\text{Erf}\left(\frac{\tilde{\mu}_r - r - 3\sigma_r}{\sqrt{2}\sigma_r}\right) + \frac{1}{4} \end{aligned} \qquad \text{Eq. 1-9}$$

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 ideal IDM's. This effect is illustrated in the zoomed region left in Fig. 1-2. 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 IDM$_{\text{ideal}}$ while approximately maintaining the shape of the IDM$_{\text{Gauss}}$ (see Fig. 1-2). The quadratic b-spline and the

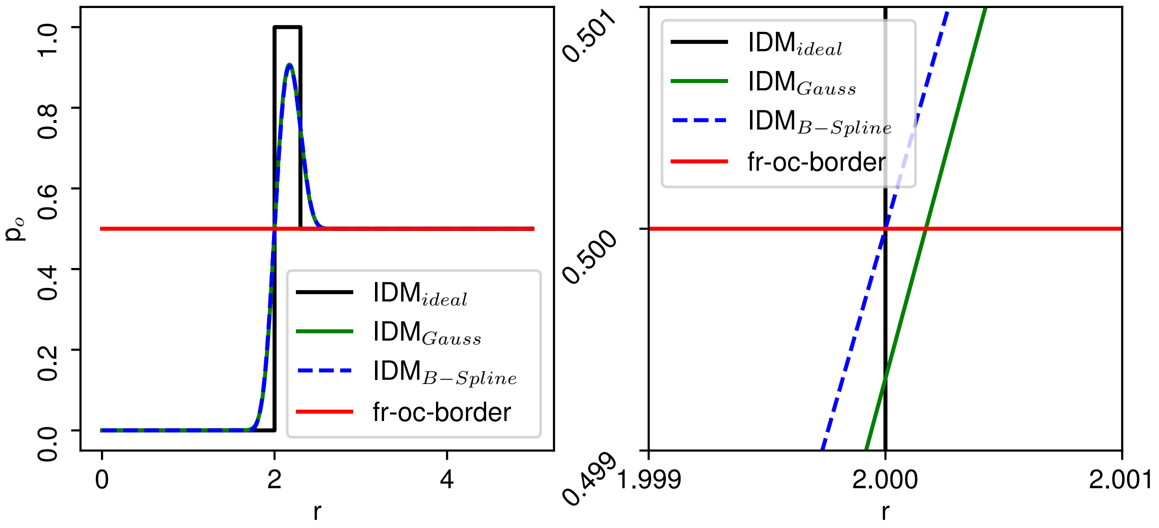


Fig. 1-2: Example of the ideal IDM with $\mu_r = 2$ and an object thickness $\tau = 0.3$, together with the IDM$_{\text{Gauss}}$ with $\sigma_r = 0.1$ and IDM$_{\text{B-Spline}}$ with the same variance. The left side shows a zoom of the right side around the border between free and occupied space.

corresponding radial IDM, after convolving the proposed b-spline with the ideal IDM, can be written as follows

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

$$\text{IDM}_{\text{B-Spline}}(r, \tilde{\mu}_r, \sigma_r, \phi = \mu_\phi) = \text{BSpline}((r - \tilde{\mu}_r)/\sigma_r) - \frac{1}{2}\text{BSpline}((r - \tilde{\mu}_r)/\sigma_r - 3) \qquad \text{Eq. 1-11}$$

The 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 as follows

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

$$\Delta p = \|0.5 - \text{IDM}_{\text{prob}}\| \qquad \text{Eq. 1-13}$$

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 1.4.1.

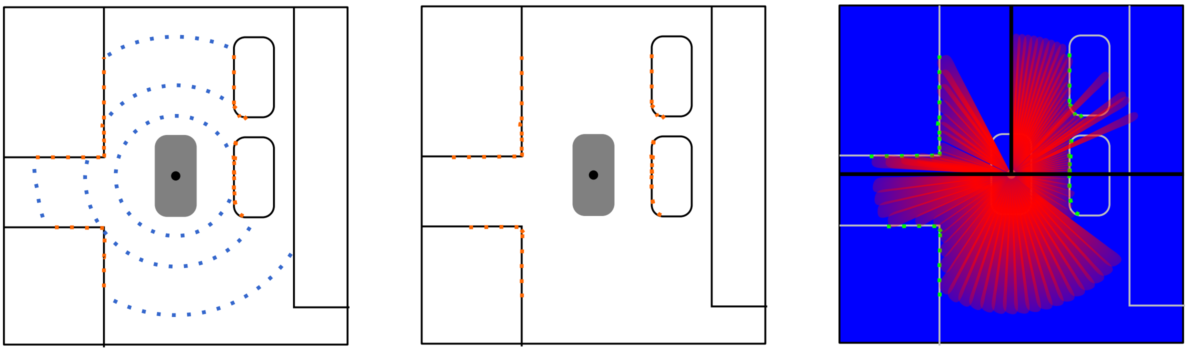
1.2.2 Geo ISMs for Lidars

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 away that they are cutoff when hitting 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 or manual labeling [VEL18]. 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 authors 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



a) ground-plane detection

b) ground-plane removal

c) ray-casting variants

Fig. 1-3: 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) and b) with ground detections in blue and object detections in orange, followed by the application of the IDM. Here, three different variants are illustrated. The upper left sector shows the result of only applying the IDM for angles with object detections. The upper right shows the application of an IDM with a small opening for all 360° angles using small angular increments. Finally, the bottom sector again shows the IDM applied for all angles but this time with a larger opening angle and bigger angular increments.

free space gets lost for areas not affected by the object detection's rays, as illustrated in the upper left sector in Fig. 1-3 c). 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. 1-3 c) upper 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 the bottom sector of Fig. 1-3 c).

1.2.3 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 there 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 ill-posed problem of directly inferring the depth using deep learning. Eigen et al. [EIG14] where on 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 additionally 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.

1.2.4 Geo ISMs for Radars

For automotive radars, the measurements are provided in the form of point clouds in 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 and the ray-casting, as described in subsec. 1.2.1, can be applied [BOU10, DUB14]. However, the accuracy of radars is not constant, as assumed in the standard ISM, but rather depends on the radial distance, reflecting material, detection angle and disturbances like multi-path reflections and atmospheric interferences. Therefore, the IDM's probabilities have to be adapted to account for these influences as proposed in [CLA12]. This model is further extended in [WER15] to account for the fact that radars can detect objects in occluded areas through multi-path reflection and, thus, violate line-of-sight assumption made in subsec. 1.2.1. The model is adapted by ignoring free space of rays which falls together with occupied areas of other rays. In [PRO18], the ego vehicle's velocity is identified as an additional factor to decrease the IDM's probabilities. Moreover, a further adaptation is proposed to tackle the sparseness in free space predicted by the Inverse Radar Model (IRM) caused by the limited amount of detections provided by the radars. They tackle the problem in the following way

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

A different approach to tackle the sparsity of free space is proposed in [SLU19]. Here, they use the standard IDM in presence of detection and an additional so called "negative ISM" in the remaining FoV, which defines the area as partially free. This is used to dampen the effect of false positives from other radars with overlapping FoVs. However, it should be mentioned that imaging radars have been deployed for their experiments which provide way denser point clouds than current standard automotive radars.

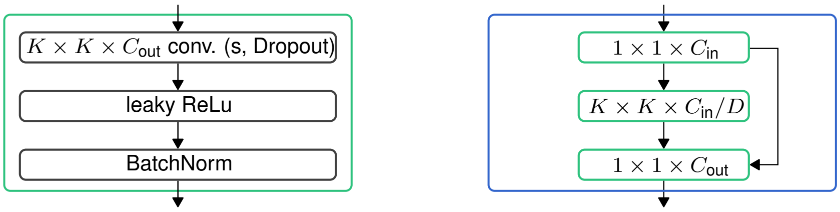
1.3 Deep Inverse Sensor Models

In recent years, the literature solely focuses on the creation of deep ISMs utilizing deep Convolutional Neural Network (CNN)s.

1.3.1 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. 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 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. An example of such a standard UNet architecture is depicted in subsec. **??** in Fig. 2-3.

The convolution layers normally consist of 3×3 convolutions with Dropout, followed by leaky ReLU activations and batch normalization, which is depicted on the left-hand side in Fig. 1-4. 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 out-



$K \times K \times C_{\text{out}}$ conv. (s, Dropout)
leaky ReLu
BatchNorm
$1 \times 1 \times C_{\text{in}}$
$K \times K \times C_{\text{in}}/D$
$1 \times 1 \times C_{\text{out}}$

Fig. 1-4: Structure of convolution (left) and ResNet layer (right) as commonly used in the literature with the kernel size K, 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.

put non-linearity, the input features are added to the outputs. The sum of multiply-add operations (MAC) over the ResNet layer's three convolutions can be written as follows

$$1 \cdot 1 \cdot WHC_{\text{in}}\frac{C_{\text{in}}}{D} + 3 \cdot 3 \cdot WH\frac{C_{\text{in}}^2}{D^2} + 1 \cdot 1 \cdot WH\frac{C_{\text{in}}}{D}C_{\text{out}} \qquad \text{Eq. 1-14}$$

for features of width W, height H, a channel reduction by D and C_{in}, C_{out} input and output channels respectively. For $C_{\text{in}} = C_{\text{out}}$, the ResNet layer needs less MACs compared to the $9 \cdot WHC_{\text{in}}^2$ 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 KL divergence 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 L_1 and 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].

1.3.2 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 are a vast amount of different approaches to tackle the problems in this field of research, Fig. 1-5 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 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 subsec. 1.2.3. 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 resample simulated patches of street layouts. This feat can be achieved by utilizing an adversarial loss as known from the literature of Generative Advrserial 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 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 inputs can work to a certain extend 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, there remains the question of how to handle occluded areas. In [SCH18], this problem is addressed in the preprocessing 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. 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 e.g. 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].

	monodepth + semantics	FCN	spatial trafo. layer	learned trafo. layer
adversarial	[SCH18]	[MAN20]		
maps	[SCH18, PHI20]	[MAN20]	[ROD20]	
sparse supervision		[MAN20, LU19]		
simulation		[REI20]	[REI20]	[PAN20]

Fig. 1-5: Overview of deep ICM literature with row-wise supervision approaches and column-wise solutions for the transformation between camera and bev projection

1.3.3 Deep ISMs for Range Sensors

The origins of approximating ISMs for range sensors using neural networks go back to the 90s [VAN95, THR93]. One this range sensors have in common is the fact that the measurement signals can be directly projected into bev images and fed into neural networks. This makes the training on radar and lidar 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 convoluted with the ideal ray-casting model from subsec. 1.2.1 and afterwards utilized for occupancy mapping.

1.3.4 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. 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 point 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.

1.4 Evidential Combination Rules

1.4.1 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

$$K = m_{f1}m_{o2} + m_{o1}m_{f2} \qquad \text{Eq. 1-15}$$

$$\boldsymbol{m}_1 \oplus_D \boldsymbol{m}_2 = \begin{bmatrix} (m_{f1}m_{f2} + m_{f1}m_{u2} + m_{u1}m_{f2})/(1-K) \\ (m_{o1}m_{o2} + m_{o1}m_{u2} + m_{u1}m_{o2})/(1-K) \\ m_{u1}m_{u2}/(1-K) \end{bmatrix} \qquad \text{Eq. 1-16}$$

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

$$\max(m_{u1}, m_{u2}) \geq m_{u12} \qquad \text{Eq. 1-17}$$

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

$$m_{u1} \geq m_{u12} \underbrace{=}_{\text{with Eq. 1-16}} m_{u1}\frac{m_{u2}}{1-K} \qquad | \div m_{u1} \qquad \text{Eq. 1-18}$$

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

$$1 \geq \frac{m_{u2}}{1-K} \underbrace{=}_{\text{with Eq. 1-5}} \frac{1 - m_{f2} - m_{o2}}{1-K} \qquad | \cdot (1-K) \qquad \text{Eq. 1-19}$$

$$1-K \underbrace{=}_{\text{with Eq. 1-15}} 1 - m_{f1}m_{o2} - m_{o1}m_{f2} \geq 1 - m_{f2} - m_{o2} \qquad | -1 + m_{f1}m_{o2} + m_{o1}m_{f2}$$

$$\text{Eq. 1-20}$$

$$0 \geq m_{f2}\underbrace{(m_{o1}-1)}_{\leq 0} + m_{o2}\underbrace{(m_{f1}-1)}_{\leq 0} \qquad \blacksquare \qquad \text{Eq. 1-21}$$

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

$$\boldsymbol{m}_1 \oplus_Y \boldsymbol{m}_2 = \begin{bmatrix} m_{f1}m_{f2} + m_{f1}m_{u2} + m_{u1}m_{f2} \\ m_{o1}m_{o2} + m_{o1}m_{u2} + m_{u1}m_{o2} \\ m_{u1}m_{u2} + K \end{bmatrix} \qquad \text{Eq. 1-22}$$

with K as defined in Eq. 1-15. 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 allows 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, ZHA20].

1.4.2 Combination of dependent Evidence

In case of dependencies between evidences, direct combination, as proposed is sec. 1.4.1, 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 subsec. 1.4.1 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

$$\gamma \otimes \boldsymbol{m} = [\gamma m_f, \gamma m_o, 1 - \gamma + \gamma m_u]^\top \qquad \text{Eq. 1-23}$$

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 form different sensor sources over time is 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 Sp and entropy H 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

$$Sp(\boldsymbol{m}) = \sum_{A \in 2^U} \frac{m(A)}{card(A)} = \frac{m_f}{1} + \frac{m_o}{1} + \frac{m_u}{2} \in [0.5, 1] \qquad \text{Eq. 1-24}$$

$$H(\boldsymbol{m}) = \sum_{A \in 2^U} -\log_2(Pl(A))m(A) = -\log_2(m_f + m_u)m_f - \log_2(m_o + m_u)m_o \in [0, 1] \qquad \text{Eq. 1-25}$$

$$Pl(A) = \sum_{B|B \cap A \neq \emptyset} m(B) \qquad \text{Eq. 1-26}$$

with $card(A)$ being the cardinality 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

$$\gamma_i = 1 - \frac{H_i}{\sum_{i=1}^{2} H_i} \qquad \text{Eq. 1-27}$$

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

$$R_g(\boldsymbol{m}_1, \boldsymbol{m}_2) = \frac{I(\boldsymbol{m}_1, \boldsymbol{m}_2)}{\sqrt{H(\boldsymbol{m}_1)H(\boldsymbol{m}_2)}} \qquad \text{Eq. 1-28}$$

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

$$\gamma = 1/R_g(\boldsymbol{m}_1, \boldsymbol{m}_2) \qquad \text{Eq. 1-29}$$

1.4.3 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]. 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 cost function over the N training tuples (x_i, y_i).

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} \frac{(\mu(x_i) - y_i)^2}{\sigma(x_i)} + \log(\sigma(x_i)) \qquad \text{Eq. 1-30}$$

For classification, the common approach is to apply Softmax output activations coupled with a cross-entropy loss. This trains the network to predict the weights of a Categorical distribution $\mathrm{Cat}(\tilde{y}_i|x_i)$ [GOO16]. To further estimate the aleatoric uncertainties, the network can be altered to predict a Dirichlet distribution $\mathrm{Dir}(p_i|\boldsymbol{\alpha}(x_i))$ over the weights by estimating its shape parameters $\boldsymbol{\alpha} \in \mathbb{R}_{>1}$. Here, the 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 $\mathrm{Dir}(p_i|\boldsymbol{\alpha}(x_i))$ is treated as a prior to a Multinomial distribution $\mathrm{Mult}(\tilde{y}_i|p_i)$ and has to be marginalized out. Thus, to train the network, the negative marginal log-likelihood

$$\mathcal{L}_i = -\log\left(\int \prod_{k=1}^{K} p_{ik}^{y_{ik}} \frac{1}{\mathrm{Beta}(\boldsymbol{\alpha}_i)} \prod_{k=1}^{K} p_{ik}^{\alpha_{ik}^{-1}} d\boldsymbol{p}_i\right) = \sum_{k=1}^{K} y_{ik}(\log(S_i) - \log(\alpha_{ik})) \qquad \text{Eq. 1-31}$$

can be minimized, as proposed in [WU19, SEN18]. Here, $\mathrm{Beta}(\cdot)$ equals the Beta distribution and $S_i = \sum_{k=1}^{K} \alpha_k$. Sensoy et al. [SEN18] additionally propose to minimize the Bayes risk for the mean squared error (MSE)

$$\mathcal{L}_i = \int \underbrace{\underbrace{\|\boldsymbol{y}_i - \boldsymbol{p}_i\|_2^2}_{\text{amount of error}} \frac{1}{\mathrm{Beta}(\boldsymbol{\alpha}_i)} \prod_{k=1}^{K} p_{ik}^{\alpha_{ik}^{-1}}}_{\text{probability of error}} d\boldsymbol{p}_i = \sum_{k=1}^{K} (y_{ik} - \tilde{p}_{ik})^2 + \frac{\tilde{p}_{ik}(1 - \tilde{p}_{ik})}{S_i + 1} \qquad \text{Eq. 1-32}$$

as an empirically verified more stable variant.

Josang [JOS18] proposed the so called "subjective logic" framework, where he argues that evidential masses can be expressed as probabilistic Dirichlet distributions using evidence $e \in \mathbb{R}_{\geq 0}$ as the connecting element. The evidence can be used as follows to transform one representation to the other

$$\boldsymbol{b} = \frac{\boldsymbol{e}}{S} \text{ and } u = \frac{K}{S} \qquad \text{Eq. 1-33}$$

$$\boldsymbol{p} = \mathbb{E}[\mathrm{Dir}(\boldsymbol{e})] = \frac{\boldsymbol{\alpha}}{S} \qquad \text{Eq. 1-34}$$

$$\alpha_k = e_k + 1 \qquad \text{Eq. 1-35}$$

This allows 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. 1-33 [SEN18].

2 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 geometric 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 learned 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 geometric ISM. The consequent sections elaborate the requirements for each of the three afore mentioned tasks which is followed by an analysis of the research gap.

2.1 Requirements

This section details the requirements for both the trainable ISM and the approach .

2.1.1 Requirements for deep, evidential 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 deep ISMs in literature given point cloud inputs (see sec. 1.3). 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 model 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 model must be capable to learn from big data.*

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

Requirement 1.3 (R1.3): *Minimize the amount of manpower, work hours and equipment needed to create the deep, evidential ISM.*

Also, the data-driven ISM should be able to run in parallel with the already existing geometric ISM on the hardware of production-ready vehicles in real-time. Since the deep ISM's estimates are only used as an enhancement, the real-time constraint can be relaxed to near real-time. The sensor with the highest capture frequency in the NuScenes dataset is the lidar sensor with 20Hz. It is thus proposed to aim for a 10Hz inference time. To emulate these hardware restrictions, the requirement can be formulated as follows

Requirement 1.4 (R1.4): *The deep, evidential ISM shall be executable with 100Hz 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 $40m \times 40m$. This is an acceptable range for low speed scenarios like parking. Additionally, the resolution of $31,25cm$ per cell is satisfactory for the deep ISM, as it is mainly used to enhance the geometric ISM. Thus, leading to the following requirement

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

Finally, on the theoretical side, the model shall estimate the evidential classes as defined in 1.1, 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.*

2.1.2 Requirements for Usage of deep, evidential ISMs as Priors in Occupancy Mapping

As mentioned in R1.4, the deep ISM only operates at near real-time. Thus, fusion will be realized asynchronously by fusing the geometric and deep ISM estimates directly into the map whenever predictions are available. Therefore, the deep ISM estimates have to suffice the additional specification for ISMs used in occupancy mapping, as defined in sec. 1.1, namely

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

Additionally, deep ISMs do also provide predictions in regions further away from sensor measurements through means of data-driven interpolation. This poses the potential to overwrite high certain predictions close to data through many low certain predictions accumulated over time. Therefore, the accumulation of deep 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 geometric ISM is considered to be a verified, production-ready model which shall solely be enhanced by the deep ISM estimates to increase convergence speed and spatial coverage of the occupancy maps. Thus, given enough measurements, the geometric ISM should be trusted over the deep ISM resulting in the following requirement
Requirement 2.3 (R2.3): *The occupancy map shall be initialized with the deep ISM's estimates up to the point when a definable amount of measurements are collected.*
Requirement 2.4 (R2.4): *The occupancy map shall converge to the geometric ISM with increasing amount of measurements.*

2.2 Research Needs

In this section, the state-of-the-art from chapter 1 is analyzed with respect to the requirements defined in sec 2.1.

2.2.1 Research Needs for deep, evidential Inverse Sensor Models

The current state-of-the-art approaches all tackle the creation of deep ISM 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 recent breakthroughs like Dropout for regularization, BatchNorm for normalization and skip connections for conservation of information, current CNN models can be designed with increase 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.
With regards to resource efficient generation of labels, as posed in R1.3, the state-of-

the-art in all cases relies on automatic label generation. For the case of bev occupancy target, as considered in this work, geometric ILMs from 360° spinning lidars are generated as labels. 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 geometric ILM, no additional manual labor is required. Therefore, the automatic generation of bev occupancy target via geometric ILMs suffices R1.3.
Nevertheless, the radar sensors used in this work have reduced perception capabilities when compared to those 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 the 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 NuScenes setup 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 can be formulated as follows

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 detections: threshold-based or semantic segmentation-based ground plane removal?*

Finally, with regards to resource efficiency during inference, the 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 2 (RQ2): *How should the amount of filters of a UNet architecture be chosen to maximize performance while keeping the run time at about 10Hz?*

Additionally, the majority of deep ISM 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 authors knowledge, non of the published methods model dynamic objects by distributing mass equally to the free and occupied class. Training on dynamic objects 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. Additionally, since no prior work has been done in this area, the capabilities of deep ISMs to model dynamic objects with regards to R 1.8 given different sensor modalities shall investigated to answer the following questions

Research Question 3 (RQ3): *To which extend are deep ISMs capable to estimate the position of dynamic objects given radar, camera and lidar data respectively?*

Research Question 4 (RQ4): *To which extend does the capability of deep ISMs to estimate the position of dynamic objects given camera and lidar inputs respectively change, when radar information is added?*

Additionally, 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 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 questions.

Research Question 5 (RQ5): *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.*

2.2.2 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 authors knowledge, the only instances of occupancy mapping with deep ISMs use the standard Bayes filtering approach. Thus, the first step consists in analyzing the characteristics and identifying short comings when applying deep ISMs for occupancy mapping.

First, in contrast to geometric 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. 2-1. 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

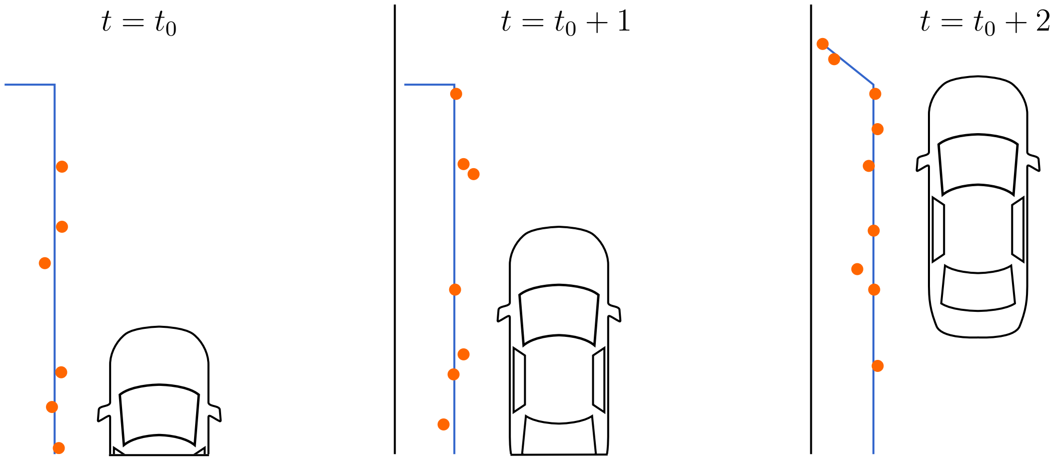


Fig. 2-1: Illustration of informational dependence between deep ISM predictions over time on the example of dataset bias. Here, the ego vehicle (black) drives along a wall and obtains radar measurements (orange) over three time steps. In each time step, the contour of the wall is estimated (blue).

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 former 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 area with predictions close to data with later occurring extrapolations. This thought experiment leads to the following hypothesis.

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, the literature in sec. 1.4.2 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. 1-23 since the Yager and Dempster combination rule, as defined in subsec. 1.4.1, 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 depends 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. 1-

28. However, no general procedure is proposed to measure the mutual information in signals. Thus, the following questions emerges
Research Question 6 (RQ6): *How can the mutual information be measured to asses the informational redundancy in evidential occupancy mapping?*
Research Question 7 (RQ7): *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. 1-24. 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 questions
Research Question 8 (RQ8): *How can the information content in evidential occupancy mapping 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 following questions
Research Question 9 (RQ9): *How can the influence of the deep ISM be disabled in case a definable amount of data has been collected, as defined in R2.3 and 2.4?*

2.3 Overview of Methodology

In this section, a framework is presented to address the research questions defined in Sec. 2.2.2. The framework extends the standard evidential occupancy mapping pipeline [PAG96] to incorporate estimates of a data-driven ISM, as shown in Fig. 2-2. Here, the choices of how to obtain the data and how to define the architecture are detailed in Sec. 2.3.3. For the learned ISM to suffice R1.7 and 1.8, a method to investigate H1 is proposed in Sec. 2.3.4.
The incorporation of the deep ISMs 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. The specific choice of the fusion methods for both geometric and deep ISMs are detailed in Sec. Todo.

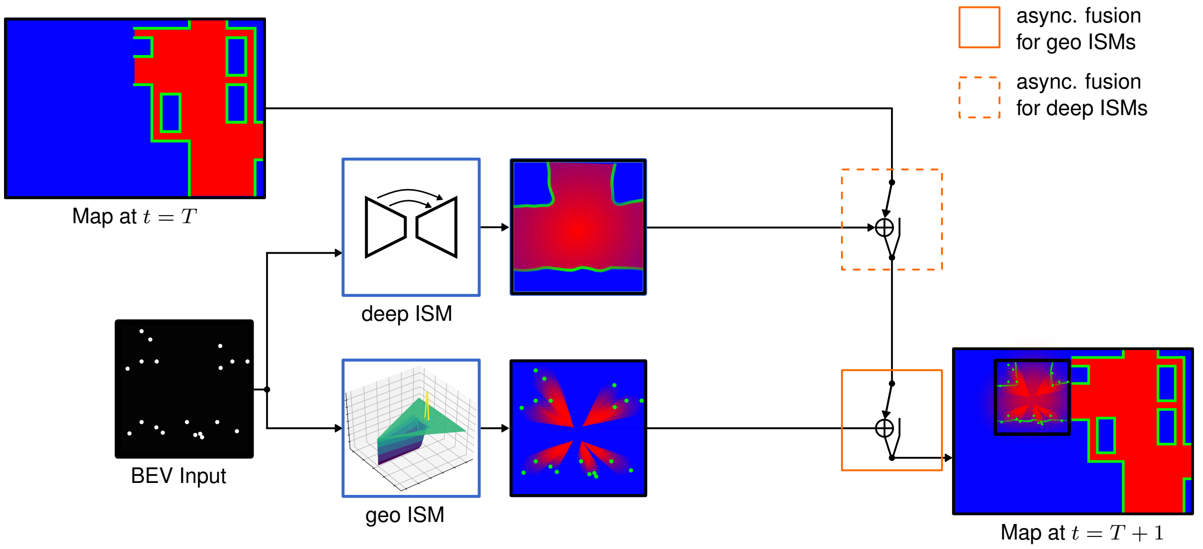


Fig. 2-2: Structural overview of the proposed framework, showing how the bev input is transformed both by the deep and geometric ISM into occupancy estimates. After removing the temporal redundancy, both ISM's estimates are fused into the occupancy map.

2.3.1 Definition of geometric ISMs for Lidars and Radars

Both lidar and radar sensors provide range measurements and, thus, the ray casting ISM as explained in Sec. 1.2.1 can be applied. However, for reasons detailed in Sec. 1.2.2 and 1.2.4, additional steps have to be taken.

In case of the **ILM**, first the non-ground detections have to be removed. Then, for each angle around the ego vehicle without a detection in line of sight, a free space ray is cast setting all cells between the lidar sensor up to a max range along the angle to $\boldsymbol{m} = [pF, 0, 1 - pF]^\top$ with pF being a parameter of the ILM. Next, for each detection in line of sight, a ray is cast as defined in [PAG96] with a parameterizable opening angle, free and occupied probability pF, pO. In case of labeling with lidar, the detections are enhanced with the motion state. For dynamic detections, the model in Eq. 1-12 is adapted as follows An algorithmic summary of the geometric ILM is shown in Alg. 1.

For the **IRM**, the main problem when applying the ray casting model from Sec. 1.2.1 is the assumption that only objects in line of sight are measured. 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 static objects by ISM rays cast towards detections in occluded areas. In this work, the occlusion problem is addressed similar to [WER15] with the alteration that free space rays are not only ignored in occupied regions of other rays, but end at the collision range. 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 define end points of ISM rays. Another problem connected with the sparsity is the low coverage of free space. In this work,

Algorithm 1: geometric Inverse Lidar Model

```
remove non-obstacle detections
initialize all cells in the ILM image to $m_u = 1$
for angle in discretizedAnlges360 do
    if (no detection along current angle) then
        castFreeSpaceRay(angle, maxRange, openingAngle, pF)
    else
        if (detection is static) then
            // see Eq.1-12
            castStatRay(angle, rangeToDetection, openingAngle, pF, pO)
        else
            // see Eq.??
            castDynRay(angle, rangeToDetection, openingAngle, pF, pD)
```

the method proposed in [PRO18] is altered by instead casting the original rays again but with a much wider opening angle. This assumes that the area close to the sensor is to a large degree free up to the closest detection. The remaining effects listed in Sec. 1.2.4 only aim at scaling the ISM to account for different sensor effects and are, thus, comparatively of minor importance. An algorithmic summary of the geometric IRM used in this work is shown in Alg. 2.

Algorithm 2: geometric Inverse Radar Model

```
initialize all cells in the IRM image to $m_u = 1$
for angle in anlgesWithCurrentDetections do
    // range at which the free space of the ray is stopped
    cutOffRange = $\infty$
    if (current or previous detection hit by current ray) then
        cutOffRange = rangeFirstHitDetection
    if (detection is static) then
        // see Eq.1-12
        castStatRay(angle, rangeToDetection, openingAngle, pF, pO,
         cutOffRange)
    else
        // see Eq.??
        castDynRay(angle, rangeToDetection, openingAngle, pF, pD,
         cutOffRange)
```

2.3.2 Methodology to define the Ground-Truth

To obtain deep ISMs, first, the generation of labels according to RQ1 has to be addressed. Here, the NuScenes dataset provides semantic labels for the lidar detections which can be utilized 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 Sec. 3.1.

To account for the fact that radars can utilize multi-path reflections to obtain detections hidden for the lidar, the geometric ISMs (see Sec. 2.3.1) of both sensors will be used. Additionally, to account for the sparseness in radar data, the ISMs are mapped over subsequent time steps and afterwards compared. This not only tackles the sparseness but also softens the requirement for lidar and radar measurements to exactly align by discretizing them into the bev grid map's cells.

To obtain a quantitative comparison, the mIoU score is computed between the evidential occupancy map classes. This score is a standard to measure the overlap between semantic classes in computer vision and is capable to handle the class imbalance present in this experiment. Nevertheless, to soften the class imbalance, the score shall only be computed in areas which can be potentially affected by the respective ISMs.

2.3.3 Methodology to define the deep ISM Architecture

Next, the network architecture has to be analyzed to suffice RQ2. To restrict the search space to a feasible subset, the architecture as illustrated in Fig. 2-3 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 subsec. 1.3.1 and Fig. 1-4). Whenever the grid's height and width remain constant, ResNet layers, as shown in Fig. 1-4, are deployed to increase efficiency. The last three layers consist of two additional convolution layer 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 Sec. 3.3.

2.3.4 Methodology to account for aleatoric Uncertainties in deep ISMs

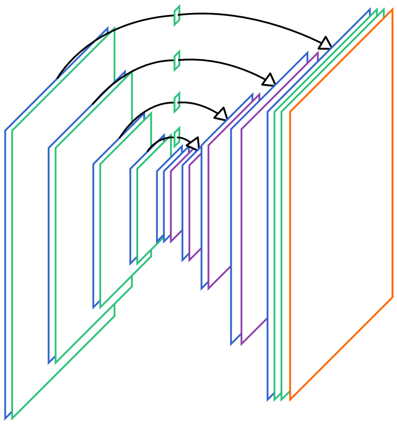
As explained in Sec. 2.2.1, the baseline deep ISM considered in this work shall model the evidential occupancy estimation using a Softmax output activation (see Eq. 2-1) and the MSE, as the standard regression loss. This configuration is from here on referred to as SoftNet.

$$\sigma(\boldsymbol{z})_i = \frac{e^{z_i}}{\sum_{k=1}^{K} e^{z_k}} \qquad \text{Eq. 2-1}$$

$$\mathcal{L}_{\text{MSE}} = \sum_{k=1}^{K} (\hat{y}_k - \tilde{y}_k)^2 \qquad \text{Eq. 2-2}$$

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

Layers



UNet Architecture

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

Encoder Architecture **Decoder Architecture**

Fig. 2-3: 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. 1-4. The skip connection in the ResNet layer is realized by adding the input to the ResNet layers output before applying the non-linearity.

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 Sec. ToDo will be trained in the SoftNet configuration using the data as specified in Sec. ToDo. Next, the magic fancy score will be computed for the test data predictions, which is an expansion of the commonly used IoU and thus provides more detailed insights. Additionally, two alternatives to SoftNet shall be investigated.

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. 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 evidential occupancy labels $\hat{\boldsymbol{m}} = [b_f, b_o, u]^\top$ to the extended representation $\hat{\boldsymbol{m}}' = [b'_d, b'_f, b'_o, u']^\top$, sufficing R 1.6, 1.7 and 1.8, can be defined as follows

$$\boldsymbol{m}' \looparrowleft \boldsymbol{m} \qquad \text{Eq. 2-3}$$

$$b'_d = 2 \cdot \min(b_f, b_o) \qquad \text{Eq. 2-4}$$

$$b'_{f/o} = b_{f/o} - \min(b_f, b_o) \qquad \text{Eq. 2-5}$$

$$u' = u \qquad \text{Eq. 2-6}$$

Here, $\min(b_f, b_o)$ 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 $\tilde{\boldsymbol{m}}'$ in the evidential occupancy mapping pipeline, a compression operation can be defined as follows

$$\boldsymbol{m} \looparrowleft \boldsymbol{m}' \qquad \text{Eq. 2-7}$$

$$b_{f/o} = b'_{f/o} - \min(b'_f, b'_o) + \frac{b'_d}{2} \qquad \text{Eq. 2-8}$$

$$u = u' + 2 \cdot \min(b'_f, b'_o) \qquad \text{Eq. 2-9}$$

This operation quantifies the learned aleatoric uncertainty between the free and occupied class as $\min(b'_f, b'_o)$, extracts it from their respective classes and shifts it 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 $\hat{\boldsymbol{m}}'$ and capable of producing evidential estimates $\tilde{\boldsymbol{m}}$ by applying the shift operation defined in Eq. 2-7 is from here on referred to as ShiftNet.

The second variation is heavily based on the method proposed by Sensoy et al. [SEN18] who train a Dirichlet network on the Bayes risk of the MSE (see Eq. 1-32) to model aleatoric uncertainty and use subjective logic (see Eq. 1-33 and 1-34) to transform the Dirichlet PDF to evidential masses. This will be referred to as DirNet. 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

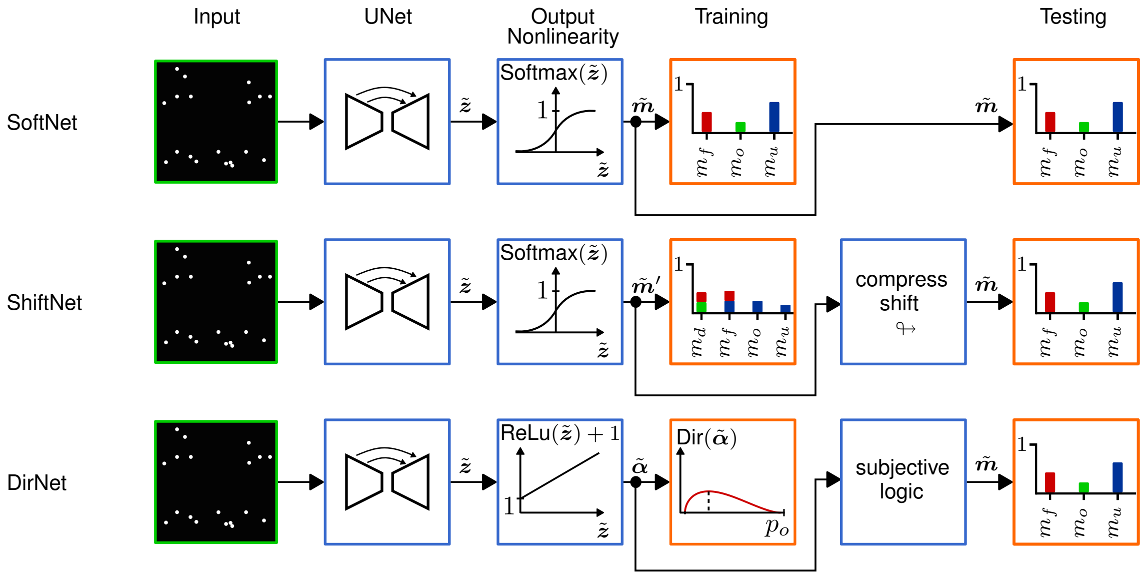


Fig. 2-4: Illustration of the three deep ISM configurations investigated in this work.

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. 1-32 shall be altered as follows

$$\mathcal{L}_i = (\hat{u}_i - \tilde{u}_i)^2 + \sum_{k\in[f,o]} (b_{ik} - \tilde{p}_{ik})^2 + \frac{\tilde{p}_{ik}(1-\tilde{p}_{ik})}{S_i + 1} \tag{Eq. 2-10}$$

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.

2.3.5 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. 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 RQ8. As stated in Sec. 2.2.2, 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 can be directly utilized to quantify the information content instead.

Given the unknown mass as a measure for information, the problem of how to quantify the temporal redundancy shall be tackled (RQ6). In this work, the redundancy shall be assessed through mutual information, for reasons explained in Sec. 2.2.2. Alternatives will be discussed in Sec. ToDo disscussion section. 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 defined as follows

$$I_{T|0:T} = I_{0:T} - I_{0:T-1} = \underbrace{m_{u,0:T-1} - m_{u,0:T}}_{\equiv \Delta m_u} \qquad \text{Eq. 2-11}$$

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. This, however, would remove all of the geometric ISM's influences. Therefore, it is rather proposed to discount the deep ISM estimate according to the amount of non-redundant information Δm_u and combine it with the current map state.

To do so, RQ7 has to be answered to obtain a discount factor for the discount operation defined in Eq. 1-23. The requirements for the discount factor γ can be formulated as follows

$$\gamma = \begin{cases} 0, \Delta m_u < 0 \\ 0, \lim\limits_{\Delta m_u \to 0} \\ 1, \lim\limits_{\Delta m_u \to 1} \end{cases} \qquad \text{Eq. 2-12}$$

It should be noted that, ideally, $\Delta m_u < 0$ never occurs since the next time step's estimate is based on at least the same amount of information as the previous one. Since Δm_u lies within the interval $[-1, 1]$, the behavior defined in Eq. 2-12 can e.g. be achieved in a linear way as follows

$$\gamma = \mathrm{ReLU}(\Delta m_u) \qquad \text{Eq. 2-13}$$

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. 2-13. Finally, the decision of which combination rule to chose to combine the deep ISM estimate after discounting the redundancy will be postponed to Sec. 2.3.6.

2.3.6 Methodology to use deep ISMs as Priors in Occupancy Mapping

In this section, the requirements R2.3 and 2.4 shall be tackled. To do so, it shall be first discussed how the prior information is integrated into the map. Normally, as explained in Sec. 1.1, 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. 2-5.

Initialization Phase
Afterwards, in the initialization phase, both the geometric 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 Sec. 2.3.5). 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 leading to the possibility that the state change cannot be fully successful. To achieve this, the following procedure is proposed.
First, restrict the certainty of the deep ISM estimates to $\underline{m}_u$ using the discounting operation as follows

$$\tilde{\boldsymbol{m}}' = (1 - \underline{m}_u) \otimes \tilde{\boldsymbol{m}} \qquad \text{Eq. 2-14}$$

This restriction of certainty, however, does not influence the amount of redundancy in the deep ISM estimates. Thus, the next step consists of removing the redundancy as proposed in sec. 2.3.5 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

$$\boldsymbol{m}^{0:T} = \boldsymbol{m}^{0:T-1} \oplus (\gamma \otimes \tilde{\boldsymbol{m}}^{0:T}) \qquad \text{Eq. 2-15}$$

$$\underline{m}_u \leq m_u^{0:T} \qquad \text{Eq. 2-16}$$

Here, the adaption of γ 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 sec. 1.4.1). Given Yager's combination rule, γ shall be chosen as follows

$$\underline{m}_u \leq m_u^{0:T} \underbrace{=}_{\text{Eq. 1-22}} m_u^{0:T-1}\tilde{m}_u^{T} + K \qquad \text{Eq. 2-17}$$

$$\underline{m}_u \underbrace{\leq}_{\text{Eq. 1-23 \& Eq. 1-15}} m_u^{0:T-1}(1-\gamma+\gamma\tilde{m}_u^{0:T}) + m_o^{0:T-1}\gamma\tilde{m}_f^{0:T} + m_f^{0:T-1}\gamma\tilde{m}_o^{0:T} \qquad \text{Eq. 2-18}$$

$$\underline{m}_u \leq m_u^{0:T-1} + \gamma\underbrace{(m_u^{0:T-1}\tilde{m}_u^{0:T} - m_u^{0:T-1} + m_o^{0:T-1}\tilde{m}_f^{0:T} + m_f^{0:T-1}\tilde{m}_o^{0:T})}_{=:\xi} \quad | - m_u^{0:T-1} | \div \xi \qquad \text{Eq. 2-19}$$

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 $[\underline{m}_u, 1]$. Thus, given $K = 1$ and $\underline{m}_u = 0$, no solution for γ can be found to suffice the requirement above. This setting, however, would be pointless since by setting $\underline{m}_u = 0$, the function of the lower bound $\underline{m}_u$ as a distinction between the initialization and convergence phase would be disabled. Thus, given $\underline{m}_u > 0$, $|\xi| \in (0, 1]$ and the two following solutions can be found

$$\text{case: } \xi > 0 \rightarrow \frac{\underline{m}_u - m_u^{0:T-1}}{\xi} \leq \gamma \qquad \text{Eq. 2-20}$$

$$\text{case: } \xi < 0 \rightarrow \frac{\underline{m}_u - m_u^{0:T-1}}{\xi} \geq \gamma \qquad \text{Eq. 2-21}$$

Here, the case of $\xi > 0$, 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 to both reduce the informational redundancy in the deep ISM estimate and, at the same time, respects a lower bound on the unknown mass $\underline{m}_u$ can be written as

$$\text{case: } \xi > 0 \rightarrow \tilde{\gamma} = \gamma \qquad \text{Eq. 2-22}$$

$$\text{case: } \xi < 0 \rightarrow \tilde{\gamma} = \min\left(\gamma, \frac{\underline{m}_u - m_u^{0:T-1}}{\xi}\right) \qquad \text{Eq. 2-23}$$

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. 2-5, 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.

Convergence Phase

Once the unknown mass has fallen below the threshold $\underline{m}_u$, the convergence phase starts. Here, the geometric ISM should still be integrated into the map while the influence of the deep ISM should be disabled, guaranteeing a convergence towards the geometric ISM.

Here, 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 adaption seeks to disable the capability to recuperate unknown mass to ensure that the unknown mass always remain 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

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

To underline the modesty of the author, this rule shall be unknown as Bauer's combination rule.

The combination approaches for each phase together with a comparison against the two baseline combination rules is provided in sec. todo. Furthermore, sec. todo shows the experimental results of applying this procedure for occupancy mapping.

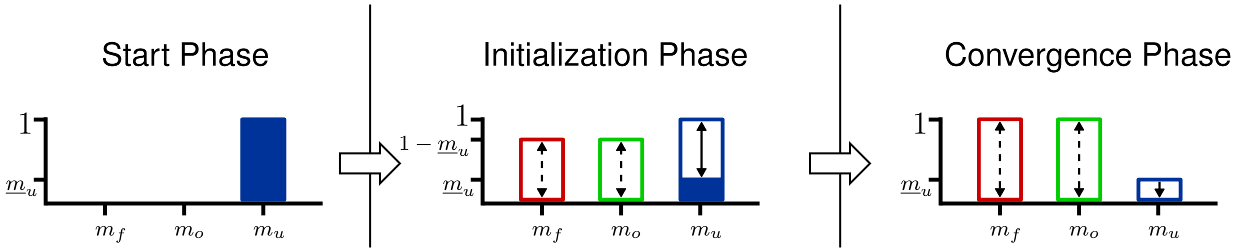


Fig. 2-5: Illustration of the three phases of a grid cell's occupancy state. Beginning with the start phase where the evidential mass is set to be all unknown. Afterwards, in the initialization phase, the unknown mass can be shifted into the free and occupied class and be recuperated using the deep and geometric ISM's predictions. Finally, in the convergence phase, the geometric ISM's estimates are used to strictly reduce the unknown mass while the deep ISM is being disabled.

3 Deep ISM Experiments

- give sensor specification

The dataset chosen in this work is the publicly available NuScenes dataset for it contains measurements of all sensor modalities investigated in the experiments (lidar with semantic information, camera, odometry information and radar), is openly available which makes the results comparable and reproducible and is among the datasets containing the biggest amount of data (see Sec. ToDo). It 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.

For this work, the train-val-test split as proposed by the creators is used, but some of the scenes have been removed. Specifically, 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 20m are considered.

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 geometric 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.

To obtain the baseline geometric IRM and ILM bev images, the ISMs as detailed in Sec. 2.3.1 and 3.1 are utilized. In the following, the creation of the investigated inputs and targets for the deep ISM are detailed.

3.1 Choice of Ground-Truth

This section details the comparison of different approaches to adapt lidar to radar bev projections. 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.

3.1.1 Experimental Setup

In the following, evidential occupancy maps of scenes as defined in the NuScenes dataset are created based on the geometric IRM and ILM as described in Sec. 2.3.1. To find the best overlap between the two types of maps, the first step in the ILM, namely the removal of ground detections will be altered. The two filters under investigation are a purely geometric height threshold-based and a semantic filter. Here, height thresholds are compare 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 geometric ISMs, a three stepped removal of the semantics is proposed as follows $[$no street, no street or sidewalk, no street, sidewalk or terrain$]$.

3.1.2 Experimental Results

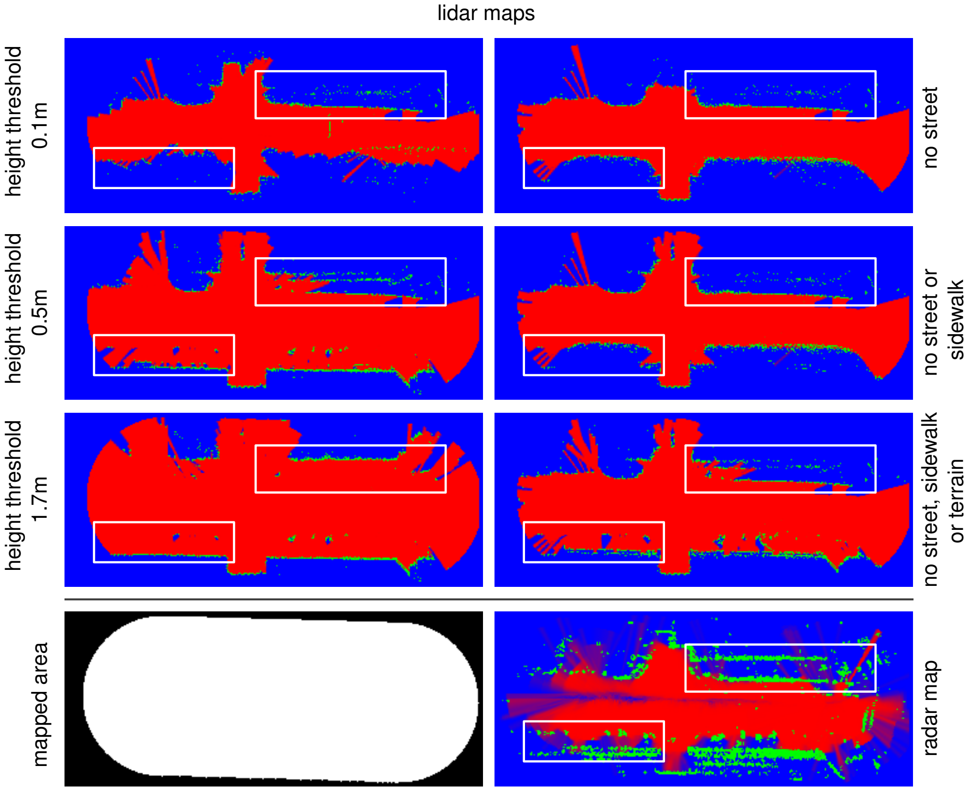


Fig. 3-1: Example of lidar maps created by successively removing ground-plane semantics and three threshold-based filters, together with the mapped area (white) and the radar map. The white box in the bottom left corner shows that for lower height threshold and up to semantics of sidewalks, many structures detected in the radar map are occluded. On the other hand, setting the height threshold too high (here demonstrated for threshold $1.7m$) removes too many points at the forefront, as shown in the white box in the upper right corner. Height threshold of about $0.5m$ or the removal of semantics up to the terrain level show a good compromise.

The quantitative comparison in Fig. 3-2 shows that the successive removal of semantics up to the terrain level leads to increasingly better overlap up to the best reached mIoU score of 11.27%. On the other hand, the height threshold-based filters show improved performance up to a height threshold of $0.5m$ with a score of 10.78% 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. 3-1. 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. 3-1. Thus, a height threshold of about $0.5m$ or the removal of semantics 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 $0.5m$ 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.

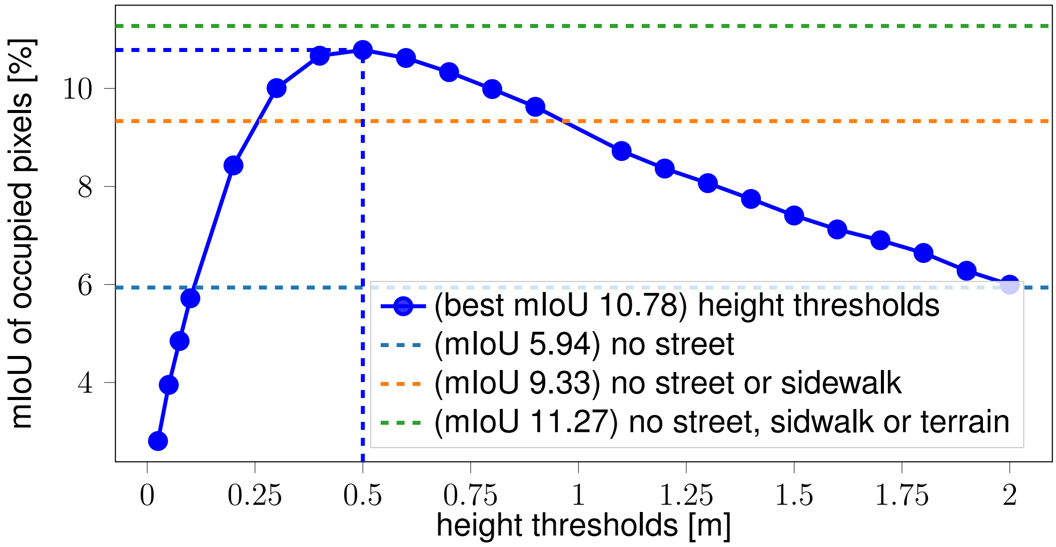


Fig. 3-2: Results of mIoU between different evidential occupancy maps create with variations of geometric ILMs and the geometric IRM computed in the mapped area.

3.2 Deep ISM Dataset

3.2.1 Deep ISM Inputs

To generate lidar bev detection images, first, the ground-plane removal as described in Sec. 3.1 is applied. 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 (see Fig.).

For the radar images, the sweeps information of all corner and the front radar are used. All of the static detections are then marked as 1.0 as opposed to the default pixel value 0.0. The dynamic detections, distinguishable through a flag provided in the NuScenes dataset, are marked as 0.5 since they indicate the transition between free and occupied. ToDo how are dynamic detections handled

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 5cm away from the ideal flat ground-plane. These lidar detections are then transformed both to the bev and the camera image. Afterwards, 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.

Additionally, a variant is considered where the semantic annotations are transformed using the homography transformation. Here, the semantic labels are being obtained by applying DeepLab V3+ [CHE18] using the Xceptin network [CHO17] as a backbone trained on the Cityscapes dataset without further finetuning.

For the monodepth projection, the semi-supervised model as proposed by [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 $[0, 1]$.

3.2.2 Deep ISM Targets

Finally, the occupancy map patches used as targets to train the deep ISMs are generated as follows. First the geometric ILM as defined in Sec. 2.3.1 and 3.1 is used to create an occupancy map of the considered scene. To reduce the effect of artifacts due to dynamic objects in the targets, the lidar sweeps are enhanced with dynamic object information. Detections tagged as "dynamic" are only used to produce free space.

To obtain the dynamic information for lidar sweeps, the sample's bounding boxes are interpolated for intermediate sweeps and marking all intersecting detections 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. For each found pair, the bounding box poses are interpolated in the temporally first bounding box's coordinate frame by a third degree polynomial in a way that the first and last position intersect and their derivatives are zero. Here, the coordinate transformation prevents the occurrence of singularities in the interpolation in case of holonomic, short distance motions. The interpolated orientation is given by the arc-tangent of the polynomial's derivative. To finally obtain the interpolated bounding box pose, the way along the interpolated trajectory is 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 interpolation procedure and the overlay of the resulting interpolated bounding boxes over a lidar sweep's detection image are illustrated in Fig. 3-3. provide algorithm for interpolation method

After the creation of the occupancy map using the above described, enhanced geometric ILM, 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.

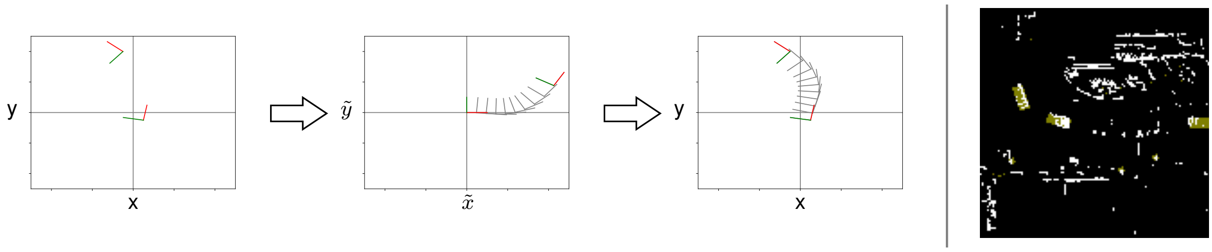


Fig. 3-3: Illustration of the three steps to obtain interpolations of the 2d bounding box poses on the left hand side together with an example of resulting interpolated bounding boxes (yellow) overlayed on a lidar sweep's bev detection image on the right. The three interpolation stages from left to right 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 Choice of UNet Architecture

3.3.1 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 Sec. **??**). More specifically, radar bev images based on one sweep's information 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 Sec. 2.3.3) is trained in SoftNet configuration (see Sec. 2.3.4), since it is the baseline configuration as proposed by the literature.

The hyperparameters searched with the following procedure are the downsample factor 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 filters, 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 filters is adapted in this work (see Sec. 1.3.1). Thus, reducing the filter search to the initial amount of filters C_0 and the maximum amount of filter $C_{\max}$. 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 Sec. 1.3.1). 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 until convergence, as indicated by the validation set, to remove the bias due to the random initialization. The experiments are conducted on two NVIDIA Tesla V100 (32GB) GPUs and evaluated on a single core of an Intel Core Processor i7-10750H CPU.

3.3.2 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 3-4 while the results are shown in Fig. 3-5 marked in orange. The experiments show an exponential decrease in inference time to obtain improve the mIoU. With regards to the time till convergence of training, the time remains about the same up to $C_0 = 24$. Afterwards, or $C_0 = 32$, it more than triples and doubles again for $C_0 = 40$. The behavior of the training and inference time are as expected since todo: show that the complexity of a resnet layer is squared with filter size.
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. todo: show some qualitative results of $C_0 = 32$ that demonstrate superiority.

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. 3-4: Hyperparameter setting for all network tuning experiments

To fine tune the capacity of the network, C_0 is fixed to 32 while an upper limited 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. 3-5. The parameter settings for the two experiments labeled 7 and 8 are shown in the lower part of 3-4. 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. At the same time, it more than halves the training time.

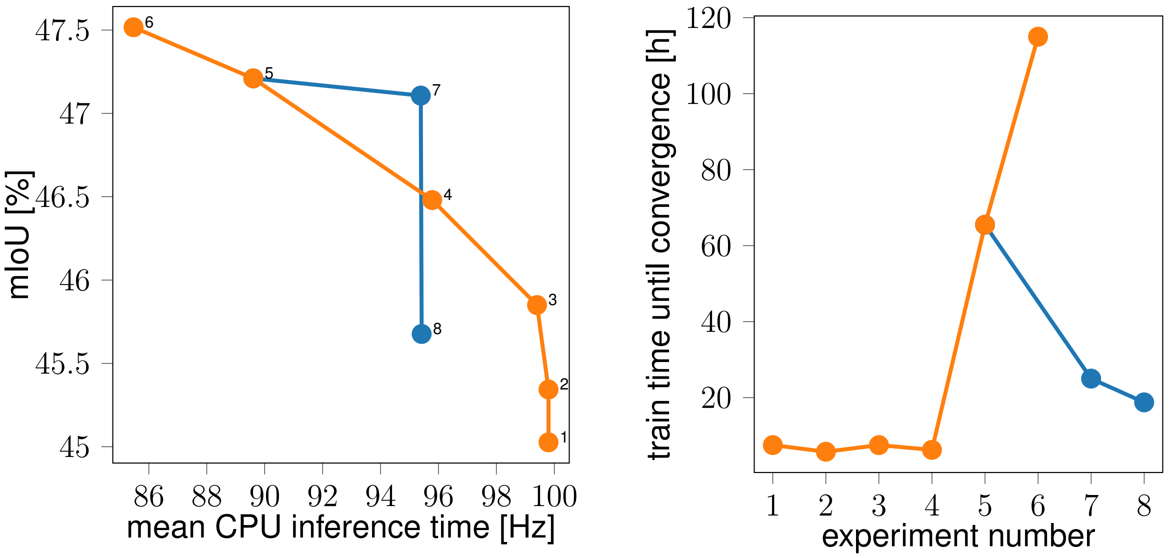


Fig. 3-5: Experimental results of the network tuning experiments. On the left-hand-side, the mIoU over the CPU inference time is shown while on the right-hand-side, the training time is plotted for each experiment. Here, orange marks the first part of the experiments in which the downsamlping rate D 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.

3.4 Aleatoric Uncertainties in deep ISMs

3.4.1 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, as stated in H 1, the SoftNet configuration treats the aleatoric uncertainty by equally distributing mass into the two classes between which the uncertainty occurs. 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. 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 in for free and occupied cells. This is 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. However, when it comes to predicting occupied space, it shows the worst performance both in positive rates and false free predictions.

For the R_{20} scores, an overall improvement of all models can be observed as expected. 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. Additionally, similar to the R_1 scores, an increase in unknown mass can be observed in occluded as compared to visible areas.

3.5 Camera-Radar Fusion in deep ISMs

3.5.1 Experimental Results

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

- homography rgb has worst overall performance, followed by homography of semantic segmentation. Monodepth bev projection surpasses both the rgb and semantic inputs in all categories. While it should be noted that it has slightly increased false rates in the occupancy category. Thus, monopdeth inputs are chosen for further experiments

- monodepth fusion with radar R_{20} images shows that the radar information mainly benefits in distinguishing dynamic and occupied cells. Free overall performance remains as is.

3.6 Lidar-Radar Fusion deep ISMs

3.6.1 Experimental Results

	k	d	f	o	u	d	f	o	u	d	f	o	u
Lidar	$p(k\|d)$	47.3	10.5	16.3	25.8	42.0	15.1	17.9	25.1	51.1	7.5	16.4	25.0
	$p(k\|f)$	2.4	78.8	1.9	16.9	1.6	89.3	0.9	8.1	4.5	51.1	4.6	39.8
	$p(k\|o)$	8.7	8.4	43.6	39.3	11.0	10.8	46.5	31.7	8.1	7.5	42.9	41.6
	$p(k\|u)$	3.0	8.2	8.7	80.1	-	-	-	-	3.0	8.0	8.7	80.3
Lidar & R$_{20}$	$p(k\|d)$	47.3	9.7	18.3	24.8	43.5	13.6	20.6	22.2	50.6	7.4	17.4	24.6
	$p(k\|f)$	1.8	80.8	1.9	15.5	1.2	89.8	1.1	7.9	3.6	56.8	4.1	35.4
	$p(k\|o)$	6.0	7.6	46.8	39.5	8.1	9.6	52.8	29.4	5.5	6.8	45.2	42.5
	$p(k\|u)$	1.6	7.6	7.4	83.3	-	-	-	-	1.6	7.5	7.3	83.5
ShiftNet		overall				visible				occluded			

- as expected, lidar has best performance for free and occupied classification for all ShiftNet inputs.
- more surprisingly, lidar is capable to distinguish between dynamic and occupied objects from mere context and achieves better performance than all other inputs for ShiftNet.
- when fused with R$_{20}$, the overall performance further increases over all classes both in positive and false rates.

	k	d	f	o	u	d	f	o	u	d	f	o	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
	$p(k\|f)$	0.0	28.9	0.0	70.9	0.0	35.9	0.0	63.9	0.0	9.6	0.0	90.2
	$p(k\|o)$	0.2	18.4	0.8	80.5	0.3	30.6	0.9	68.1	0.2	14.5	0.8	84.4
	$p(k\|u)$	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
	$p(k\|f)$	35.8	37.0	3.3	24.0	36.2	42.6	3.1	18.2	35.5	21.5	4.0	39.1
	$p(k\|o)$	37.8	9.6	17.2	35.5	43.7	11.5	20.0	24.8	35.9	9.0	16.2	38.9
	$p(k\|u)$	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
	$p(k\|f)$	13.6	47.5	5.6	33.3	15.5	56.8	4.6	23.2	9.0	22.3	8.4	60.3
	$p(k\|o)$	13.5	9.7	20.7	56.1	22.4	14.3	22.9	40.3	10.8	8.4	19.9	60.9
	$p(k\|u)$	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
	$p(k\|f)$	7.4	48.6	2.0	42.0	7.5	56.0	1.6	34.9	7.6	28.1	3.2	61.1
	$p(k\|o)$	8.4	15.1	13.7	62.8	10.0	19.2	14.9	55.9	7.9	13.9	13.4	64.8
	$p(k\|u)$	4.0	9.4	4.7	81.9	-	-	-	-	4.0	9.3	4.7	82.0
R$_1$ Scores		overall				visible				occluded			

	k	d	f	o	u	d	f	o	u	d	f	o	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
	$p(k\|f)$	0.4	47.0	0.7	51.7	0.4	57.8	0.5	41.1	0.6	17.5	1.2	80.6
	$p(k\|o)$	2.6	16.8	13.5	67.0	2.7	31.0	15.4	50.7	2.5	12.3	13.0	72.1
	$p(k\|u)$	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
	$p(k\|f)$	21.6	64.1	2.0	12.2	19.5	72.0	1.6	6.9	28.1	42.6	3.3	26.0
	$p(k\|o)$	36.2	12.9	22.8	28.1	43.2	15.3	25.8	15.6	33.9	12.2	21.9	32.0
	$p(k\|u)$	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
	$p(k\|f)$	10.9	61.5	5.1	22.4	11.3	71.4	3.9	13.5	10.6	35.4	8.3	45.7
	$p(k\|o)$	15.2	10.5	31.8	42.6	25.0	15.1	35.2	24.6	12.0	9.1	30.7	48.2
	$p(k\|u)$	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
	$p(k\|f)$	4.7	62.9	3.2	29.2	4.3	71.1	2.4	22.1	6.1	40.8	5.3	47.8
	$p(k\|o)$	6.0	13.6	28.5	51.8	7.4	17.6	30.9	44.1	5.7	12.4	27.8	54.1
	$p(k\|u)$	3.0	9.0	7.8	80.2	-	-	-	-	3.0	8.9	7.8	80.3
R$_{20}$ Scores		overall				visible				occluded			

Fig. 3-6: Results of deep ISM variants trained on R$_1$ and R$_{20}$ images for visible, occluded and overall areas. Here, the visible area is defined as the area which is not unknown in the geometric ILM used to create the ground-truth maps. Thus, only a stochastically insignificant portion of unknown labels remains in the visible area, which is why these scores are excluded from consideration. The scores are given in the form of a confusion matrix for each model where each row shows the probabilities in percentage of estimates given the true class. Values in green correspond to true positives, red shows the percentages of false predictions. The unknown class predictions (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.

4 Bibliography

[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

[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

[BOU10] BOUZOURAA, M. E., HOFMANN, U.
Fusion of occupancy grid mapping and model based object tracking for driver assistance systems using laser and radar sensors
2010 IEEE Intelligent Vehicles Symposium, IEEE, 2010, pp. 294–300

[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

[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

[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

[DUB14] DUBÉ, R., HAHN, M., SCHÜTZ, M., DICKMANN, J., GINGRAS, D.
Detection of parked vehicles from a radar based occupancy grid
2014 IEEE Intelligent Vehicles Symposium Proceedings, IEEE, 2014, pp. 1415–1420

[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

[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

[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)

[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

[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

[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

[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

[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)

[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

[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

[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

[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

[VEL18] 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

[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)

[ZHA20] 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

5 Publications

Bringen wir Vorveröffentlichungen so in die Arbeit rein?

The following earlier publications by the author contain parts of this thesis.

Bibliography of some earlier papers