Radar Artifact Labeling Framework (RALF): Method for Plausible Radar Detections in Datasets

Simon T. Isele[1,3,*], Marcel P. Schilling[1,2,*], Fabian E. Klein[1,*], Sascha Saralajew[4] and J. Marius Zoellner[3,5]

[1]*Dr. Ing. h.c. F. Porsche AG, Weissach, Germany*

[2]*Institute for Automation and Applied Informatics, Karlsruhe Institute of Technology, Eggenstein-Leopoldshafen, Germany*

[3]*Institute of Applied Informatics and Formal Description Methods, Karlsruhe Institute of Technology, Karlsruhe, Germany*

[4]*Bosch Center for Artificial Intelligence, Renningen, Germany*

[5]*FZI Research Center for Information Technology, Karlsruhe, Germany*

Keywords: Radar Point Cloud, Radar De-noising, Automated Labeling, Dataset Generation.

Abstract: Research on localization and perception for Autonomous Driving is mainly focused on camera and LiDAR datasets, rarely on radar data. Manually labeling sparse radar point clouds is challenging. For a dataset generation, we propose the cross sensor Radar Artifact Labeling Framework (RALF). Automatically generated labels for automotive radar data help to cure radar shortcomings like artifacts for the application of artificial intelligence. RALF provides plausibility labels for radar raw detections, distinguishing between artifacts and targets. The optical evaluation backbone consists of a generalized monocular depth image estimation of surround view cameras plus LiDAR scans. Modern car sensor sets of cameras and LiDAR allow to calibrate image-based relative depth information in overlapping sensing areas. K-Nearest Neighbors matching relates the optical perception point cloud with raw radar detections. In parallel, a temporal tracking evaluation part considers the radar detections' transient behavior. Based on the distance between matches, respecting both sensor and model uncertainties, we propose a plausibility rating of every radar detection. We validate the results by evaluating error metrics on semi-manually labeled ground truth dataset of $3.28 \cdot 10^6$ points. Besides generating plausible radar detections, the framework enables further labeled low-level radar signal datasets for applications of perception and Autonomous Driving learning tasks.

1 INTRODUCTION

Environmental perception is a key challenge in the research field of Autonomous Driving (AD) and mobile robots. Therefore, we aim to boost the perception potential of radar sensors. Radar sensors are simple to integrate and reliable also in adverse weather conditions (Yurtsever et al., 2020). Post processing of reflected radar signals in the frequency domain, they provide 3D coordinates with additional information e.g. signal power or relative velocity. Such reflection points are called detections. But drawbacks such as sparsity (Feng et al., 2020) or characteristic artifacts (Holder et al., 2019b) call for discrimination of noise, clutter, and multi-path reflections from relevant detections. Radar sensors are classically applied for Adaptive Cruise Control (ACC) (Eriksson and As, 1997) and state-of-the-art object detection (Feng et al., 2020). But to the authors best knowledge, radar raw signals are rarely used directly for AD or Advanced Driver Assistance Systems (ADAS).

Publicly available datasets comparable to KITTI (Geiger et al., 2013) or Waymo Open (Sun et al., 2020) lack radar raw detections, and recently published datasets of nuScenes (Caesar et al., 2020) or Astyx (Meyer and Kuschk, 2019) are the only two available datasets containing both radar detections and objects respectively. However, transferability suffers from undisclosed preprocessing of radar signals e. g. (Caesar et al., 2020) or only front facing views (Meyer and Kuschk, 2019). Investigations for example on de-noising of radars by means of neural networks in supervised learning or other radar applications for perception in AD, currently require

*Authors contributed equally.

Isele, S., Schilling, M., Klein, F., Saralajew, S. and Zoellner, J.
Radar Artifact Labeling Framework (RALF): Method for Plausible Radar Detections in Datasets.
DOI: 10.5220/0010395100220033
In *Proceedings of the 7th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2021)*, pages 22-33
ISBN: 978-989-758-513-5

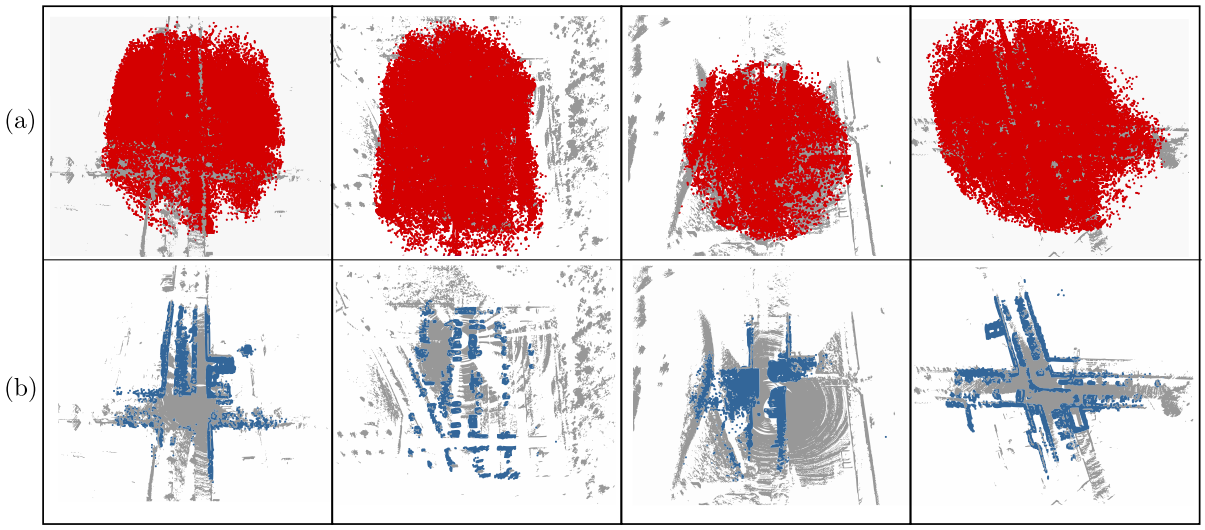


Figure 1: Scene comparison of four exemplary sequences with (a) raw radar detections (red) with underlying LiDAR (grey) and (b) manually corrected ground truth labels (blue) with plausible detections ($\hat{y}(\mathbf{p}_{r,i,t}) = 1$).

expensive and non-scaleable manually labeled datasets.

In contrast, we propose a generic method to automatically label radar detections based on on-board sensors denoted as Radar Sensor Artifact Labeling Framework (RALF). RALF enables a competitive, unbiased, and efficient data-enrichment pipeline as an automated process to generate consistent radar datasets including knowledge covering plausibility of radar raw detections, see Figure 1. Inspired by Piewak et al. (2018), RALF applies the benefits of cross-modal sensors and is a composition of two parallel signal processing pipelines as illustrated in Figure 2: Optical perception (I), namely camera and LiDAR as well as temporal signal analysis (II) of radar detections. Initially, false labeled predictions of RALF can be manually corrected, so one obtains hereby a ground truth dataset. This enables evaluation of RALF, optimization of its parameters, and finally unsupervised label predictions on radar raw detections.

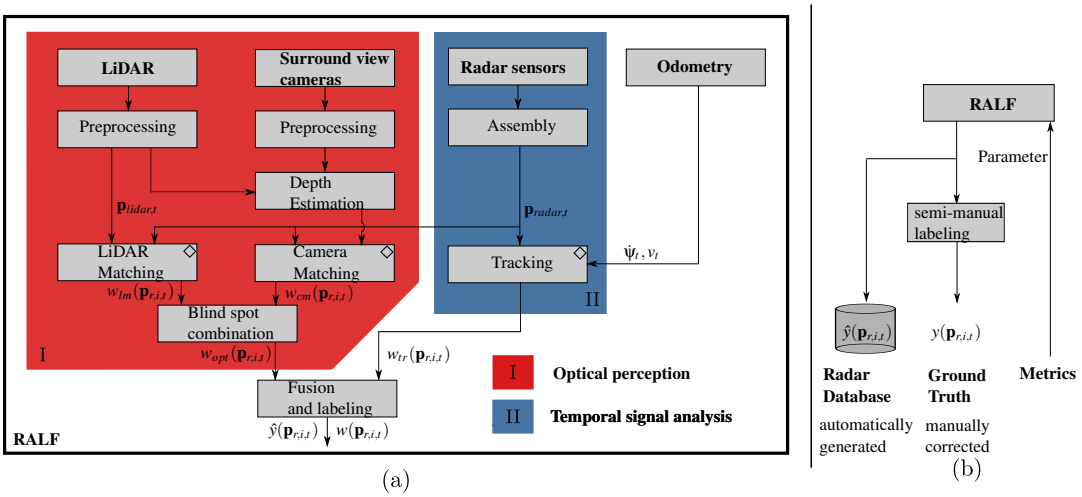
Our key **contributions** are the following:

1. An evaluation method to rate radar detections based on their existence plausibility using LiDAR and a monocular surround view camera system.
2. A strategy to take the transient signal course into account with respect to the detection plausibility.
3. An automated annotation framework (RALF) to generate binary labels for each element of radar point cloud data describing its plausibility of existence, see Figure 1.
4. A simple semi-manual annotation procedure using predictions of RALF to evaluate the labeling results, optimize the annotation pipeline, and generate a reviewed radar dataset.

2 RELATED WORK

To deploy radar signals more easily in AD and ADAS applications, raw signal de-noising[1] is compulsory. Radar de-noising can be done on different abstraction levels, at lowest on received reflections in the time or frequency domain (e. g. Rock et al. (2019)). At a higher signal processing level of detections in 3D space, point cloud operations offer rich opportunities. For instance, point cloud representations profit from the importance of LiDAR sensors and the availability of many famous public LiDAR datasets (Sun et al. (2020); Caesar et al. (2020); Geiger et al. (2013)) in company with many powerful point cloud processing libraries such as pcl (Rusu and Cousins, 2011) or open3D (Zhou et al., 2018). Transferred to sparse LiDAR point cloud applications, Charron et al. (2018) discussed shortcomings of standard filter methods (e. g. image filtering approaches to fail at sparse point cloud de-noising). DBSCAN (Ester et al., 1996) is an adequate measure to cope with sparse noisy point clouds (Kellner et al., 2012). Point

[1]We use the term de-noising to distinguish between plausible radar detections and artifacts, denoting no limitation only to noise in the classical sense. Our understanding of radar artifacts is based on the work of Holder et al. (2019b). To name an example, an artifact could be a mirror reflection, a target could be a typical object like a car, building or poles.



(a) (b)

Figure 2: Annotation pipeline RALF (a) with branches (I, II) and crucial components (⋄) as well as the overall method (b).

Cloud Libraries Libraries (e. g. Zhou et al., 2018; Rusu and Cousins, 2011) provide implementations of statistical outlier removal and radius outlier removal. Radius outlier removal is adapted considering proportionality between sparsity and distance (Charron et al., 2018), but the problem of filtering sparse detections in far range still remains unsolved. To generate maps of the static environment with radar signals in a pose GraphSLAM (Thrun and Montemerlo, 2006), Holder et al. (2019a) applies RANSAC (Fischler and Bolles, 1981) and M-estimators (Huber, 1964) to filter detections by their relative velocity information. Applying neural networks is an alternative strategy to filter out implausible points considering traffic scenes as a whole (Heinzler et al., 2020).

However, considering supervised deep learning approaches to be trained for filtering, ground truth labels are required. To the authors' best knowledge, there are no publicly available radar point cloud datasets explicitly enriched with raw point detection and related labels. Point-wise manual labeling is too time-consuming and therefore an automated labeling process is necessary. Piewak et al. (2018) developed an auto-labeling to create a LiDAR dataset. in this framework, the underlying idea is to make use of a state-of-the-art semantic segmentation model for a camera. With that, pixel-wise class information is associated to LiDAR detections by projection of the point cloud into the segmented image. The main correspondence problem for such a method are different Field of Views (FoVs), resulting in obstruction artifacts or differing aspect ratios. To compensate this, Piewak et al. (2018) suggested sensors to be mounted in closest possible proximity to avoid correspondence problems. Recently, Behley et al. (2019)) published a semantically segmented LiDAR dataset, whose structural shell allows to transfer their workflow to other point cloud data.

2.1 Method

The proposed framework, visualized in Figure 2, consists of an optical perception branch (I), namely camera and LiDAR, and temporal signal analysis (II). These branches are fused in the framework to output a consistent label of radar detections. RALF is implemented in the Robot Operating System (Quigley et al., 2009).

2.2 Problem Formulation and Notation

Inspired by other notations (Fan and Yang (2019); Qi et al. (2017)), a point cloud at time t is represented by spatial coordinates and corresponding feature tuples $\mathcal{P}_t = \{(\mathbf{p}_{1,t}, \mathbf{x}_{1,t}), \ldots, (\mathbf{p}_{N_t,t}, \mathbf{x}_{N_t,t})\}$, where N_t denotes the total number of detections at time t. Spatial information is typically range r, azimuth angle φ, and elevation angle ϑ. Additionally, radar detection specific information, e. g. Doppler velocity or signal power of the reflection, is contained in the feature vector $\mathbf{x}_{i,t} \in \mathbb{R}^C$ of point $\mathbf{p}_{r,i,t}$. The basic concept of the proposed annotation tool is to enrich each radar detection $\mathbf{p}_{r,i,t}$ with a corresponding feature attribute, namely plausibility $w(\mathbf{p}_{r,i,t}) \in [0,1]$. The term plausibility describes the likelihood of a radar detection to represent an existing object ($y(\mathbf{p}_{r,i,t}) = 1$) or an artifact ($y(\mathbf{p}_{r,i,t}) = 0$).

2.3 Annotation Pipeline of RALF

In the following, we describe the single modules that align a sensor signal with the reference system.

Algorithm 1: LiDAR matching.

Require: $\mathcal{P}_{\text{radar},t}, \mathcal{P}_{\text{lidar},t}$
Ensure: $w_{\text{lm}}(\mathbf{p}_{r,i,t})$
  for $it = 1, \dots N_{\text{radar},t}$ **do**
    $\mathbf{q} \leftarrow$ K-NN($\mathcal{P}_{\text{lidar},t}$, $\mathbf{p}_{i,t}$, K)
    $d \leftarrow 0$
    for $l = 1, \dots K$ **do**
      $p_{x,l,t}, p_{y,l,t}, p_{z,l,t} \leftarrow \mathcal{P}_{\text{lidar},t}$.get_point($\mathbf{q}[l]$)
      $r_{l,t}, \varphi_{l,t}, \vartheta_{l,t} \leftarrow \mathcal{P}_{\text{lidar},t}$.get_features($\mathbf{q}[l]$)
      $\sigma_{d,i,l} \leftarrow$ MODEL($r_{r,i,t}, \varphi_{r,i,t}, \vartheta_{r,i,t}, r_{l,t}, \varphi_{l,t}, \vartheta_{l,t}$)
      $d \leftarrow d + \sqrt{\frac{\Delta p_{x,t}^2 + \Delta p_{y,t}^2 + \Delta p_{z,t}^2}{\sigma_{d,i,l}^2 + \varepsilon}}$
    end for
    $w_{\text{lm}}(\mathbf{p}_{r,i,t}) \leftarrow \exp(-\beta_{\text{lm}} \frac{d}{K})$
  end for

To enable comparison of multi-modal sensors, time synchronization (Faust and Pradeep, 2020) and coordinate transformation (Dillmann and Huck, 2013) into a common reference coordinate system (see Section 3.1) are necessary.

LiDAR Matching. This module aligns radar detections with raw LiDAR reflections as described in Algorithm 1. The LiDAR reflections are assumed to be reliable and unbiased. Based on a flexible distance measure, plausibility of radar detections is determined. The hypothesis of matching reliable radar detections with LiDAR signals in a single point in space does not hold in general. LiDAR signals are reflected on object shells, while radar waves might also penetrate objects. Thus, some assumptions and relaxations are necessary in Algorithm 1. We assume for the assessment no negative weather impact on LiDAR signals and comparable reflection modalities. Furthermore, since radar floor detections are mostly implausible, we estimate the LiDAR point cloud ground plane parameters via RANSAC (Fischler and Bolles, 1981) and filter out corresponding radar points. Applying a k-Nearest Neighbor (k-NN) clustering (Zhou et al. (2018); Rusu and Cousins (2011)) in Algorithm 1, each radar detection $\mathbf{p}_{r,i,t}$ of the radar point cloud $\mathcal{P}_{\text{radar},t}$, is associated with its K nearest neighbors of the LiDAR scan $\mathcal{P}_{\text{lidar},t}$. Notice that values of K greater than one improve the robustness due to less sparsity in LiDAR scans. Measurement equations

$$h_x = r_{r,i} \cos\vartheta_{r,i} \cos\varphi_{r,i} + {}^{\text{v}}x_{\text{radar}} - (r_l \cos\vartheta_l \cos\varphi_l + {}^{\text{v}}x_{\text{lidar}}), \quad (1)$$

$$h_y = r_{r,i} \cos\vartheta_{r,i} \sin\varphi_{r,i} + {}^{\text{v}}y_{\text{radar}} - (r_l \cos\vartheta_l \sin\varphi_l + {}^{\text{v}}y_{\text{lidar}}), \quad (2)$$

$$h_z = r_{r,i} \sin\vartheta_{r,i} + {}^{\text{v}}z_{\text{radar}} - (r_l \sin\vartheta_l + {}^{\text{v}}z_{\text{lidar}}) \quad (3)$$

are introduced as components of L^2 norm d in Cartesian coordinates. Radar $(r_{r,i}, \varphi_{r,i}, \vartheta_{r,i})$ and LiDAR detections $(r_l, \varphi_l, \vartheta_l)$ are initially measured in the local sphere coordinate system. Constant translation offsets $({}^{\text{v}}x_{\text{radar}}, {}^{\text{v}}y_{\text{radar}}, {}^{\text{v}}z_{\text{radar}})$ and $({}^{\text{v}}x_{\text{lidar}}, {}^{\text{v}}y_{\text{lidar}}, {}^{\text{v}}z_{\text{lidar}})$ relate the local sensor origins to the vehicle coordinate system. Assuming independence between uncertainties of radar coordinate measurements $(\sigma_{r,\text{radar}}, \sigma_{\varphi,\text{radar}}, \sigma_{\varphi,\text{radar}})$ as well as Time-of-Flight LiDAR uncertainty $(\sigma_{r,\text{lidar}})$, error propagation in Cartesian space can be obtained by

$$\sigma_{d,i,l}^2 = \left(\frac{\partial d}{\partial r_{r,i}}\right)^2 \sigma_{r,\text{radar}}^2 + \left(\frac{\partial d}{\partial \varphi_{r,i}}\right)^2 \sigma_{\varphi,\text{radar}}^2 + \left(\frac{\partial d}{\partial \vartheta_{r,i}}\right)^2 \sigma_{\vartheta,\text{radar}}^2 + \left(\frac{\partial d}{\partial r_l}\right)^2 \sigma_{r,\text{lidar}}^2 \quad (4)$$

denoted as MODEL in Algorithm 1. Rescaling the mismatch in each coordinate dimension enables the required flexibility in the LiDAR matching module. To ensure $w_{\text{lm}}(\mathbf{p}_{r,i,t}) \in [0,1]$ and also increase resolution in small mismatches d, an exponential decay function with a tuning parameter $\beta_{\text{lm}} \in \mathbb{R}^+$ is applied subsequently.

Camera Matching. Holding for general mountings, we undistort the raw images and apply a perspective transformation to obtain a straight view, Figures 3 and 4. We derive the modified intrinsic camera matrix $\mathbf{A} \in \mathbb{R}^{3\times3}$ for the undistorted image (Scaramuzza et al., 2006). To match 3D radar point clouds with camera perception, a dense optical representation is necessary.

Structure-from-Motion (SfM) (Mur-Artal et al., 2015) on monocular images reconstructs sparsely. Reconstruction is often incomparable to radar, due to few salient features in poorly structured parking scenarios e. g. plain walls in close proximity. Moreover, initialization issues in low-speed situations naturally degrades SfM to be reliable for parking.

Hence, we apply a pre-trained version of DiverseDepth (Yin et al., 2020) on pre-processed images to obtain relative depth image estimations. Thanks to the diverse training set and the resulting generalization, it outperforms other estimators trained only for front cameras on datasets such as KITTI (Geiger et al., 2013). Thus, it is applicable to generic views and cameras. To match the depth image estimations to a metric scale, LiDAR detections in the overlapping FoV are projected into each camera frame considering intrinsic and extrinsic camera parameters (world2cam). The projected LiDAR reflections serve as sparse sampling points from which the depth image pixels are metrically rescaled. Local scaling fac-

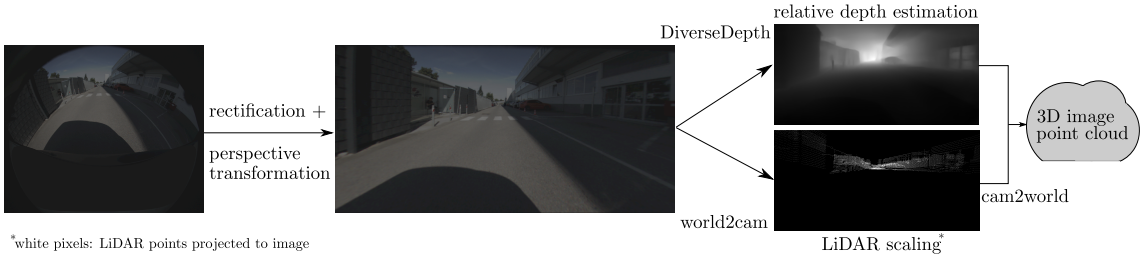


Figure 3: Camera matching pipeline.

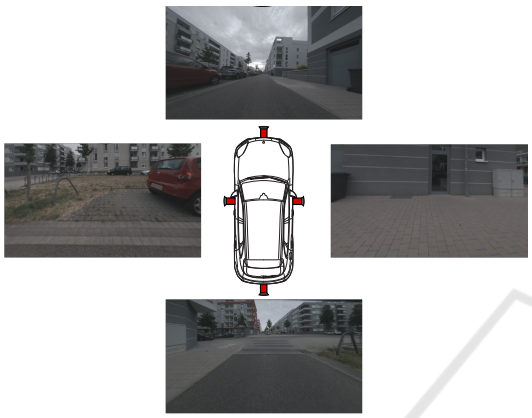


Figure 4: Preprocessed surround view camera in example scene.

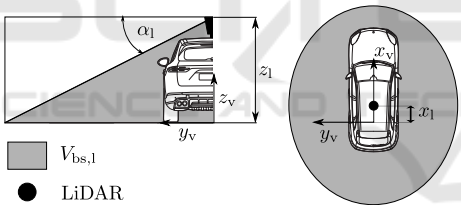


Figure 5: Blind spot LiDAR.

tors outperform single scaling factors from robust parameter estimation (Fischler and Bolles (1981); Huber (1964)) at metric rescaling. Equidistant samples of depth image pixels are point-wisely calibrated with corresponding LiDAR points via KNN. Figure 3 shows how the calibrated depth is projected back to world coordinates by the (cam2world) function. Afterwards, the association to radar detections analogously follows Algorithm 1, but considers the uncertainty of the depth estimation and its propagation in Cartesian coordinates.

Though, being aware of camera failure modes, potential model failures requests for manual review of automated RALF results. Measuring inconsistencies between camera depth estimation and extrapolated LiDAR detections or LiDAR depth in overlapping FoVs, indicate potential failures.

Blind Spot Combination. In the experimental setup, described in Section 3.1, optical perception utilizing a single, centrally mounted LiDAR sensor lacks to cover the whole radar FoV as illustrated in Figure 5. The set

$$V_{\text{bs,l}} = \Big\{ \mathbf{p}_i = (p_{r,x,i}, p_{r,y,i}, p_{r,z,i})^\top \in \mathbb{R}^3 \mid p_{r,z,i} \in [0, z_\text{l}] \wedge \sqrt{(p_{r,x,i} - x_\text{l})^2 + p_{r,y,i}^2} \leq \frac{z_\text{l} - p_{r,z,i}}{\tan \alpha_\text{l}} \Big\} \tag{5}$$

describes the LiDAR blindspot resulting from schematic mounting parameters ($y_\text{l} = 0, z_\text{l} > 0$) and opening angle α_l greater than zero. Considering the different FoVs,

$$w_{\text{opt}}(\mathbf{p}_{r,i,t}) = \begin{cases} w_{\text{cm}}(\mathbf{p}_{r,i,t}) & \mathbf{p}_{r,i,t} \in V_{\text{bs,l}} \\ w_{\text{lm}}(\mathbf{p}_{r,i,t}) & \text{otherwise} \end{cases} \tag{6}$$

summarizes the plausibility of the optical perception branch (I). Far range detection relies only on LiDAR sensing, while only cameras sense the nearfield. At overlapping intermediate sensing ranges, both rankings from camera and LiDAR scan are available instead of being mutually exclusive. Experiments yielded more accurate results for this region by preferring LiDAR over camera instead of a compromise of both sensor impressions.

Tracking. Assuming Poisson noise (Bühren and Yang, 2007) on radar detections, it is probable that real existing objects in space form hot spots over consecutive radar measurement frames, whereas clutter and noise is almost randomly distributed. Since labeling is not necessarily a real-time capable online process, one radar point cloud $\mathcal{P}_{\text{radar},t_k}$ at t_k forms the reference to evaluate spatial reoccurence of detections in radar scan sequences. Therefore, a batch of $n_\text{b} \in \mathbb{N}$ earlier and subsequent radar point clouds are buffered around the reference radar scan at time t_k. Considering low speed planar driving maneuvers, applying a kinematic single-track model based on wheel odometry is valid (Werling, 2017). Based on the measured yaw rate $\dot{\psi}$, the longitudinal vehicle velocity v, and the known time difference Δt between radar scan $k+1$

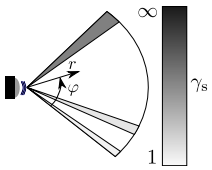


Figure 6: Reliabilty scaling.

and k, the vehicle state is approximated by

$$\begin{aligned}\left(x_\mathrm{v}, y_\mathrm{v}, z_\mathrm{v}, \psi\right)_{k+1}^\top = \left(x_\mathrm{v}, y_\mathrm{v}, z_\mathrm{v}, \psi\right)_k^\top \\ +\Delta t \cdot \left(v\ \cos\psi, v\ \sin\psi, 0, \dot{\psi}\right)_k^\top.\end{aligned} \tag{7}$$

Considering Equation (7) and rotation matrix $R_{z,\psi}$, containing yaw angle ψ, allows ego-motion compensation for each point i of the buffered radar point clouds $\mathcal{P}_{\mathrm{radar},t_{k+j}}$ for $j \in \{-n_\mathrm{b}, \ldots, -1,\ 1, \ldots, n_\mathrm{b}\}$ to the reference cloud $\mathcal{P}_{\mathrm{radar},t_k}$. Each point

$$\begin{aligned}\tilde{\mathbf{p}}_{r,i,t_{k+j}} = R_{z,\psi_k}^{-1}\left(\left(x_\mathrm{v}, y_\mathrm{v}, z_\mathrm{v}\right)_{k+j}^\top - \left(x_\mathrm{v}, y_\mathrm{v}, z_\mathrm{v}\right)_k^\top\right) \\ +\ R_{z,\psi_{k+j}-\psi_k}\ \mathbf{p}_{r,i,t_{k+j}}\end{aligned} \tag{8}$$

represents the spatial representation of a consecutive radar scan after ego-motion compensation to the reference radar scan at time step t_k. Enabling temporal tracking and consistency checks on the scans, n_b should be chosen regarding the sensing cycle time. Assuming to analyze mainly static objects, Equation (8) is valid. To describe dynamic objects, Equation (8) has to be extended considering Doppler velocity and local spherical coordinates of each detection. By taking error propagation in the resulting measurement equations into account, different uncertainty dimensions are applicable. We apply spatial uncertainty as in Equation (4). The analysis of n_b scans result in a batch of distance measures d_j. Simple averaging fails due to corner-case situations in which potentially promising detections remain undetected in short sequences. Hence, sorting d_j in ascending order and summing the sorted distances, weighted by a decreasing coefficient with increasing position, yields promising results.

Fusion and Final Labeling. The outputs of optical perception and tracking are combined with a setup-specific sensor a-priori information $\gamma_\mathrm{s}(\varphi) \in [1, \infty)$, see Figure 6. Since radar sensors are often covered behind bumpers, inhomogenous measurement accuracies $\gamma_\mathrm{s}(\varphi_{r,i,t})$ arise over the azimuth range φ, see Figure 6. The a-priori known sensor specifics are modeled by the denominator in Equation (9).

The tuning parameter $\alpha \in [0,1]$ prioritizes between the tracking and optical perception module, formalized as first term $w(\mathbf{p}_{r,i,t})$ of the Heaviside function $H : \mathbb{R} \to \{0,1\}$ argument in Equation (9). The final binary labels to discriminate artifacts ($y = 0$) from promising detections ($y = 1$) are obtained by

$$\begin{aligned}\hat{y}(\mathbf{p}_{r,i,t}) &= H\left(w(\mathbf{p}_{r,i,t}) - w_0\right) \\ &= H\left(\frac{\alpha\ w_\mathrm{opt}(\mathbf{p}_{r,i,t}) + (1-\alpha)\ w_\mathrm{tr}(\mathbf{p}_{r,i,t})}{\gamma_\mathrm{s}(\varphi_{r,i,t})} - w_0\right),\end{aligned} \tag{9}$$

where $w_0 \in [0,1]$ ia a threshold on the prioritized optical perception and tracking results.

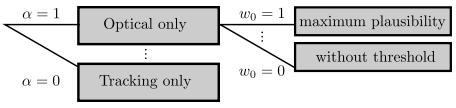


Figure 7: Parameter selection in RALF for branch weights α and plausibility threshold w_0.

2.4 Use-case Specific Labeling Policy

Motives for labeling a dataset might vary with the desired application purpose, along with conflicting parameter selection for some use cases. Our framework parameters allow to tune the automated labeling. High $\alpha = 1$ suppresses radar detections without LiDAR detections or camera detections in their neighborhood, while low $\alpha = 0$ emphasizes temporal tracking over the visual alignment, see Figure 7.

Low $\alpha = 0$ settings include plausible detections to occur behind LiDAR reflections, e. g. reflecting from inside a building. But, plausible radar detections are required to be locally consistent over several scans. On the upper bound $\alpha = 1$, the temporal tracking consistency constraint vanishes its influence on the plausible detections, resulting in an optical filtering. For instance, plausibility is rated high around LiDAR and camera perception. Examples for the relevance of temporal tracking might be the localization on radar. To recognize a known passage, the scene signature and temporal sequence of radar scans might be much more important than a de-noised representation of a scene. The other extreme might be the use-case of semantic segmentation on radar point clouds where one is interested in the nearest and shell describing radar reflections, omitting reflections from the inside of objects. Parameter w_0 acts as threshold margin on the plausibility.

RALF parameters $\alpha, \beta_\mathrm{lm}, \beta_\mathrm{cm}, \beta_\mathrm{tr}, n_\mathrm{b}$ and K can be tuned by manual inspection and according to the desired use-case. The error metrics in Equation (10)-(14), introduced in the Appendix, help to finetune the parametrization as visualized in Figure 2(b).

Optimizing RALF subsequentially leads to more accurate predictions and decaying manual label corrections. However, proper initial parameters and hav-

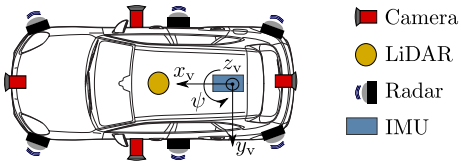


Figure 8: Sensor setup.

Table 1: Sensor setup details.

Name	Details
Camera	4 x Monocular surround view camera (series equipment)
LiDAR	Rotating Time-of-Flight LiDAR (centrally roof-mounted, 40 channel)
Radar	77 GHz FMCW Radar (FoV: 160 degree h. , ±10 degree v.)

ing a manageable parameter space is essential. After this fine-tuning step, no further manual parameter inspection of RALF is necessary. The automatically predicted radar labels can be directly used to annotate the dataset.

We tune the desired performance to achieve an overestimating function. Manual correction benefits of coarser estimates that can be tailored to ground truth labels, whereas extension of bounds requires severe interference with clutter classifications. Hence, in Table 3, we aim for high Recall values while allowing lower Accuracy.

2.5 Error Evaluation

An error measure expressing the quality of the automated labeling is essential in two aspects, namely to check if the annotation pipeline is appropriate in general and to optimize its parameters. Since this paper proposes a method to generate plausibility of radar detections, it is challenging to describe a general measure that evaluates the results. Without ground truth labels, only indirect metrics are possible. For instance distinctiveness, expressed as difference between means of weights per class in combination with balance of class members. However, several cases can be constructed in which this indirect metric misleads. Therefore, we semi-manually labeled a set of $M = 11$ different scans assisted by RALF to correctly evaluate the results, see Section 3.2 and Table 3.

3 EXPERIMENTS

In the first section, we describe the hardware sensor setup for the real world tests. After that, we discuss how the labeling policy depends on the use-case, and finally evaluate the results of the real world test.

3.1 Sensor Setup and Experimental Design

The vehicle test setup is depicted in Figure 8 and sensor set details are found in Table 1. We evaluate the radar perception for a radar-mapping parking functionality. Eleven reference test tracks (e. g. urban area, small village, parking lot and industrial park) were considered and are depicted in the Appendix. To ensure a balanced, heterogeneous dataset, they contain parking cars, vegetation, building structures as well as a combination of car parks (open air and roofed). Vehicle velocity v below 10kph during the perception of the test track environment ensures large overlapping areas in consecutive radar scans.

3.2 Semi-manual Labeling using Predictions of RALF

We use the predictions of RALF as a first labeling guess for which humans are responsible to correct false positives and false negatives. To ensure accurate corrections of RALF predictions, we visualize both radar and LiDAR clouds in a common reference frame and consider all parallel cameras to achieve ground truth data semi-manually.

Table 2: Test set class balance over $N = 2704$ radar scans.

ΣN	$\frac{\varnothing N(y=1)}{\varnothing N}$	$\frac{\varnothing N(y=0)}{\varnothing N}$
3 288 803	21.55 %	78.45 %

We base the evaluation on a dataset of eleven independent test tracks for which we compare the RALF labels versus the manually corrected results. Containing two classes, the overall class balance of the dataset is 78.45% clutter (2 580 066 radar points) against 21.55% plausible detections (708 737 radar points). Details can be found in Table 3. Additional imagery info and dataset statistics are found in the appendix. Please note, following our manual correction policy, radar reflections of buildings are reduced to their facade reflections. Intra-building reflections are re-labeled as clutter although the detections might correctly result from inner structures. This results in heavily distorted average IoU values of plausible detections ranging from 36.7% to 60.9%, Table 3. This assumption is essential for re-labeling and manual evaluation of plausible detections. Intra-structural reflections are hard to rate in terms of plausibility. Be-

Table 3: Test dataset of M = 11 sequences.

ID	∅N	∅Acc	∅Precision	∅Recall	F1 plausible	∅IoU	IoU plausible	IoU artifact
	Σ 2704	0.873	0.826	0.779	0.675	0.682	0.510	0.854
00	245	0.848	0.833	0.793	0.722	0.687	0.565	0.810
01	290	0.883	0.796	0.781	0.646	0.673	0.477	0.869
02	101	0.878	0.839	0.822	0.740	0.720	0.588	0.852
03	400	0.885	**0.706**	0.761	0.622	0.662	0.451	0.873
04	334	**0.927**	0.863	**0.859**	**0.765**	**0.768**	**0.620**	**0.917**
05	163	0.910	**0.869**	0.845	0.757	0.752	0.609	0.895
06	170	**0.776**	0.771	**0.678**	**0.537**	**0.554**	**0.367**	**0.742**
07	422	0.860	0.808	0.739	0.616	0.644	0.445	0.824
08	265	0.850	0.759	0.771	0.622	0.640	0.451	0.829
09	82	0.845	0.813	0.776	0.684	0.667	0.520	0.814
10	232	0.874	0.844	0.798	0.715	0.703	0.556	0.850

sides, in a transportation application, the sensory hull detection of objects needs to be reliable.

Using the data format of SemanticKitti dataset for LiDAR point clouds (Behley et al., 2019), the evaluation of reliable radar detections orientates on the following clusters: `human`, `vehicle`, `construction`, `vegetation`, `poles`. Our proposed consolidation of original SemanticKITTI classes to a reduced number of clusters is found in the Appendix, see Table 5. Especially the `vegetation` class imposes labeling consistency. E. g. grass surfaces can be treated as relevant if the discrimination of insignificant reflections from ground seems possible. On the other hand, grass and other vegetation are source of cluttered, temporally and often also spatially unpredictable reflections.

Different labeling philosophies impose the necessity of a consistent labeling policy. In the discussed dataset of this work, we emphasize on grass surfaces as plausible radar reflections in order to discriminate green space from road surface. Stuctural reflections are labeled based on facade reflections, while intra-vehicle detections are permitted as relevant. Please note, road surface reflections are labeled as clutter. Table 2 shows the class distribution of the labeled reference test of M = 11 sequences with in average ∅N = 1216 radar detections for evaluation in Section 3.3. Inspecting Table 2, please note, that the dataset has imbalanced classes, so a detection is more likely to be an artifact.

3.3 Results

The following section discusses the results on real world data and includes an evaluation.

Monocular Depth Estimation. Figure 9 illustrates that the pre-trained depth estimation network provides fair results on pre-processed surround view cameras.

Figure 9: Input image (a) for depth estimation (b) with highlighted objects; depth encoded by increasing brightness (b).

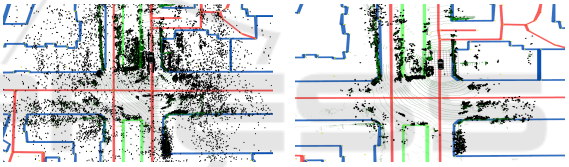


Figure 10: Birds-eye-view on example a scene with colored OpenStreetMap data (OpenStreetMap Contributors, 2017). Comparison accumulated radar scans (black points): (a) all detections (b) plausible RALF detections ($\hat{y}(\mathbf{p}_{i,t}) = 1$) without relabeling.

Key contribution to achieve reasonable results on fish-eye images without retraining are a perspective transformation and undistortion. However, in very similar scenes as shown in Figure 9, it is challenging to estimate the true relative depth. By using local LiDAR scales as we propose, this issue can be solved elegantly and thus an overlap of LiDAR and camera FoV is a helpful benefit. Moreover, the results in Figure 3 and Figure 9 show that the depth estimation network generalizes to other views and scenes.

Qualitative Comparison Raw vs. Labeled Data. Figure 10 illustrates an example scene. The results of RALF in this scene compared to unlabeled raw data are shown in Figures 4 and 10. Scene understanding is considerably facilitated.

Quantitative Evaluation. We use the prefaced manually labeled test set to evaluate the proposed

Table 4: Confusion matrix on dataset with ΣN detections.

	$y=1$	$y=0$
$\hat{y}=1$	2 432 440	268 869
$\hat{y}=0$	157 076	430 418

pipeline. The confusion matrix of the dataset are found in Table 4. By achieving a mean error $L =$ 12.95 %
= (1-Accuracy) on the dataset, we demonstrate the capability of the proposed pipeline to generate meaningful labels on real world test tracks. Please note the beneficial property of overestimation. Comparing an average Recall of 77.9 % to an average Precision 82.6 % in Table 3, there is no preferred error in the labeling pipeline of RALF. Since RALF can be parameterized, increasing Recall and decreasing Precision or vice versa is possible by tuning the introduced parameters w_0 and α. Inspecting the differences of the performance per sequence, sequence 06 is exemplary for the lower performance, while sequence 04 performs best. Interestingly, these two sequences overlap partly. Sequence 06, including an exit of a narrow garage, poses difficulties in the close surrounding of the car, which explains the performance decrease.

Robustness. Errors can be provoked by camera exceptions (lens flare, darkness, etc.) and assumption violations. Near-field reconstruction results suffer in cases when ground and floor-standing objects in low height can not distinguished accurately, yielding vague near-field labels. Furthermore, in non-planar environments containing e. g. ascents, the planar LiDAR floor extraction misleads. This causes RALF to mislabel radar floor detections. Moreover, the tracking module suffers at violated kinematic single-track model assumptions.

4 CONCLUSION

We propose RALF, a method to rate radar detections concerning their plausibility by using optical perception and analyzing transient radar signal course. By a combination of LiDAR, surround view cameras, and DiverseDepth, we generate a 360 degree perception in near- and far-field. DiverseDepth yields a dense depth estimation, outperforming SfM approaches. Monitored via LiDAR, failure modes can be detected. Since considering model and sensor uncertainties respectively, a flexible comparison using different sensors is possible. From the optical perception branch, radar detections can be enriched by LiDAR or camera information as a side effect. Such a feature is useful for developing applications using annotated radar datasets. To evaluate RALF and finetune its parameters, RALF predictions can be semi-manually corrected to ground truth labels. Recorded vehicle measurements on real-world test tracks yield an average Accuracy of 87.3% at average Precision of 82.6% of the proposed labeling method, though satisfying de-noising capabilities. The evaluation reveals positive effects of an overestimating labeling performance. Time and effort for labeling are reduced significantly. As side notice, the labeling policy is coupled with the desired use-case and evaluation metrics which may differentiate. We plan to extend the work on the framework towards semantic labeling.

ACKNOWLEDGEMENTS

We thank Marc Muntzinger form Car.Software Org, Marc Runft from IAV, and Michael Frey from Institute of Vehicle System Technology (Karlsruhe Institute of Technology) for their valuable input and the discussions. Furthermore, we would like to thank the whole team at the Innovation Campus from Porsche AG. Moreover, we thank all reviewers whose comments have greatly improved this contribution.

REFERENCES

Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., and Gall, J. (2019). Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 – November 2, 2019*, pages 9296–9306. IEEE.

Bühren, M. and Yang, B. (2007). Simulation of automotive radar target lists considering clutter and limited resolution. In *Proc. of International Radar Symposium*, pages 195–200.

Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., and Beijbom, O. (2020). nuscenes: A multimodal dataset for autonomous driving. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 11618–11628. IEEE.

Charron, N., Phillips, S., and Waslander, S. L. (2018). Denoising of lidar point clouds corrupted by snowfall. In *15th Conference on Computer and Robot Vision, CRV 2018, Toronto, ON, Canada, May 8-10, 2018*, pages 254–261. IEEE Computer Society.

Dillmann, R. and Huck, M. (2013). *Informationsverarbeitung in der Robotik*, page 270. Springer-Lehrbuch. Springer Berlin Heidelberg.

Eriksson, L. H. and As, B. (1997). Automotive radar for adaptive cruise control and collision warning/avoidance. In *Radar 97 (Conf. Publ. No. 449)*, pages 16–20.

Ester, M., Kriegel, H., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In Simoudis, E., Han, J., and Fayyad, U. M., editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*, pages 226–231. AAAI Press.

Everingham, M., Eslami, S. M. A., Gool, L. V., Williams, C. K. I., Winn, J. M., and Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vis.*, 111(1):98–136.

Fan, H. and Yang, Y. (2019). PointRNN: Point recurrent neural network for moving point cloud processing. *pre-print*, arXiv:1910.08287.

Faust, J. and Pradeep, V. (2020). Message filter: Approximate time. http://wiki.ros.org/message_filters/ApproximateTime. Accessed: 2020-04-30.

Feng, D., Haase-Schutz, C., Rosenbaum, L., Hertlein, H., Gläser, C., Timm, F., Wiesbeck, W., and Dietmayer, K. (2020). Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, PP:1–20.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395.

Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics: The KITTI dataset. *Int. J. Robotics Res.*, 32(11):1231–1237.

Heinzler, R., Piewak, F., Schindler, P., and Stork, W. (2020). CNN-based LiDAR point cloud de-noising in adverse weather. *IEEE Robotics Autom. Lett.*, 5(2):2514–2521.

Holder, M., Hellwig, S., and Winner, H. (2019a). Real-time pose graph SLAM based on radar. In *2019 IEEE Intelligent Vehicles Symposium, IV 2019, Paris, France, June 9–12, 2019*, pages 1145–1151. IEEE.

Holder, M., Linnhoff, C., Rosenberger, P., Popp, C., and Winner, H. (2019b). Modeling and simulation of radar sensor artifacts for virtual testing of autonomous driving. In *9. Tagung Automatisiertes Fahren*.

Huber, P. J. (1964). Robust estimation of a location parameter. *Ann. Math. Statist.*, 35(1):73–101.

Kellner, D., Klappstein, J., and Dietmayer, K. (2012). Grid-based DBSCAN for clustering extended objects in radar data. In *2012 IEEE Intelligent Vehicles Symposium, IV 2012, Alcal de Henares, Madrid, Spain, June 3–7, 2012*, pages 365–370. IEEE.

Meyer, M. and Kuschk, G. (2019). Automotive radar dataset for deep learning based 3d object detection. In *2019 16th European Radar Conference (EuRAD)*, pages 129–132.

Mur-Artal, R., Montiel, J. M. M., and Tardós, J. D. (2015). ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robotics*, 31(5):1147–1163.

OpenStreetMap Contributors (2017). Planet dump retrieved from https://planet.osm.org. https://www.openstreetmap.org.

Piewak, F., Pinggera, P., Schäfer, M., Peter, D., Schwarz, B., Schneider, N., Enzweiler, M., Pfeiffer, D., and Zöllner, J. M. (2018). Boosting LiDAR-based semantic labeling by cross-modal training data generation. In Leal-Taixé, L. and Roth, S., editors, *Computer Vision - ECCV 2018 Workshops - Munich, Germany, September 8-14, 2018, Proceedings, Part VI*, volume 11134 of *Lecture Notes in Computer Science*, pages 497–513. Springer.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5099–5108.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. (2009). ROS: An open-source robot operating system. volume 3.

Rock, J., Tóth, M., Messner, E., Meissner, P., and Pernkopf, F. (2019). Complex signal denoising and interference mitigation for automotive radar using convolutional neural networks. In *22th International Conference on Information Fusion, FUSION 2019, Ottawa, ON, Canada, July 2-5, 2019*, pages 1–8. IEEE.

Rusu, R. B. and Cousins, S. (2011). 3d is here: Point cloud library (PCL). In *IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, 9-13 May 2011*. IEEE.

Scaramuzza, D., Martinelli, A., and Siegwart, R. (2006). A toolbox for easily calibrating omnidirectional cameras. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2006, October 9-15, 2006, Beijing, China*, pages 5695–5701. IEEE.

Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., and Anguelov, D. (2020). Scalability in perception for autonomous driving: Waymo open dataset. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 2443–2451. IEEE.

Thrun, S. and Montemerlo, M. (2006). The graph SLAM algorithm with applications to large-scale mapping of urban structures. *Int. J. Robotics Res.*, 25(5–6):403–429.

Werling, M. (2017). *Optimale aktive Fahreingriffe: Für Sicherheits- und Komfortsysteme in Fahrzeugen*, pages 89–90. De Gruyter Oldenbourg.

Yin, W., Wang, X., Shen, C., Liu, Y., Tian, Z., Xu, S., Sun, C., and Renyin, D. (2020). DiverseDepth: Affine-

invariant depth prediction using diverse data. *pre-print*, arXiv:2002.00569.

Yurtsever, E., Lambert, J., Carballo, A., and Takeda, K. (2020). A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access*, 8:58443–58469.

Zhou, Q., Park, J., and Koltun, V. (2018). Open3d: A modern library for 3d data processing. *pre-print*, arXiv:1801.09847.

APPENDIX

Class Consolidation

The authors of SemanticKITTI (Behley et al., 2019) introduce a class structure in their work. To transfer this approach to radar detections, we propose a consolidation of classes as found in Table 5.

Table 5: Proposed clustering of SemanticKITTI classes (Behley et al., 2019) to determine radar artifacts.

Cluster	SemanticKITTI Classes
Vehicle	`car, bicycle, motorcycle, truck, other-vehicle, bus`
Human	`person, bicyclist, motorcyclist`
Contruction	`building, fence`
Vegetation	`vegetation, trunk, terrain`
Poles	`pole, traffic_sign, traffic_light`
Artifacts	`sky, road, parking, sidewalk, other-ground`

Metrics

The applied metrics in Table 3 are formulated based on the state-of-the art binary classification metrics True Positive (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN).

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{10}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{11}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{12}$$

$$\text{F1} = \frac{2 \cdot TP}{2 \cdot TP+FP+FN} \tag{13}$$

The metric mean Intersection-over-Union (⌀IoU) is based on the mean Jaccard Index (Everingham et al., 2015) which is normalized over the classes C. The IoU expresses the labeling performance class-wise.

$$\varnothing\text{IoU} = \frac{1}{C}\sum_{c=1}^{C}\frac{TP_c}{TP_c+FP_c+FN_c} \tag{14}$$

Sequence Description

The set of sequences are shortly introduced for visual inspection and scene understanding.

Sequence 00. Urban crossing scene with buildings, parked cars and vegetation in form of singular trees along the road; see Figure 1.

Sequence 01. Scene on open space along parked vehicles. Green area beside street and buildings in background; not displayed due to space limitation.

Sequence 02. Straight urban scene, road framed by buildings; Figure 11.

Sequence 03. Public parking lot with parking rows framed by vegetation (bushes, hedges and trees); see Figure 1.

Sequence 04. Exit of a garage and maneuver in front of building; see Figure 12.

Sequence 05. Urban crossing scene with open space around crossing, road framed by buildings; not displayed due to space limitation.

Sequence 06. Scene on open space along parked vehicles. Green area beside street and buildings in background; see Figure 1. Other driving direction as in Sequence 01. Overlapping area with sequence 04.

Sequence 07. Public parking lot with parking rows framed by vegetation (bushes, hedges, and trees); see Figure 13.

Sequence 08. Urban crossing scene with buildings, parked cars and vegetation in form of singular trees in crossbreeding road; see Figure 14.

Sequence 09. Residential area with single-family houses and front yards as road frame; see Figure 15.

Sequence 10. Urban area, straight drive along row of fishbone oriented cars on one side, opposed to a fence; see Figure 16. The fence was labeled plausible in order to represent a impassable wall.

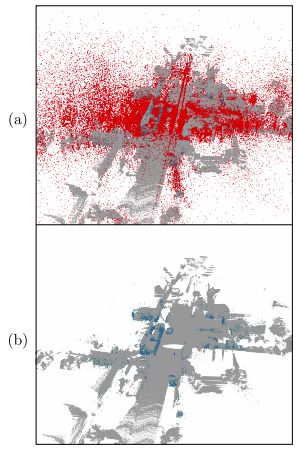


Figure 11: Sequence 02 with (a) radar raw detections (red), LiDAR (grey) and (b) corrected labels ($y(\mathbf{p}_{r,i,t}) = 1$) in blue.

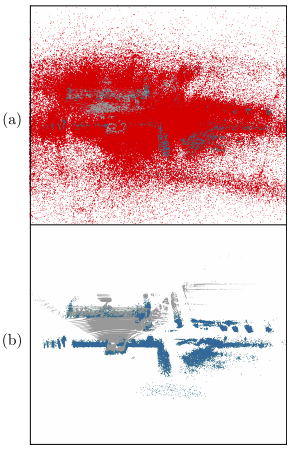


Figure 12: Sequence 04; figure description is equal to Figure 11.

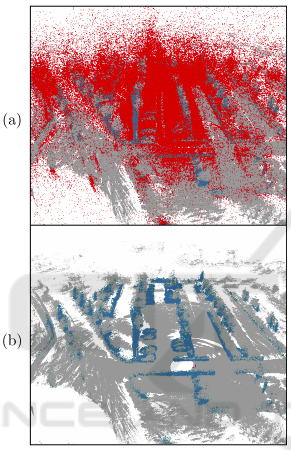


Figure 13: Sequence 07; figure description is equal to Figure 11.

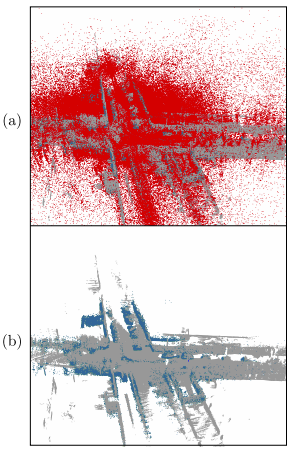


Figure 14: Sequence 08; figure description is equal to Figure 11.

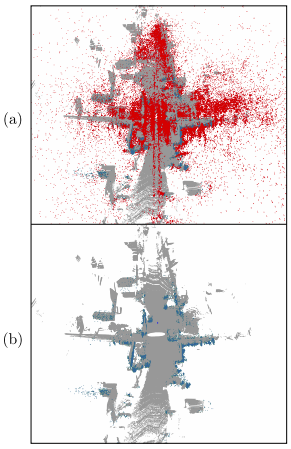


Figure 15: Sequence 09; figure description is equal to Figure 11.

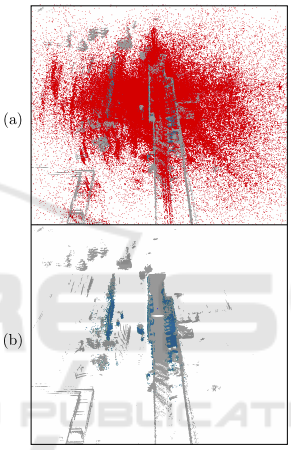


Figure 16: Sequence 10; figure description is equal to Figure 11.