Simultaneous Localization and Mapping (SLAM) for Synthetic Aperture Radar (SAR) Processing in the Field of Autonomous Driving

Timo Grebner, *Graduate Student Member, IEEE*, Ron Riekenbrauck, *Graduate Student Member, IEEE*, and Christian Waldschmidt, *Fellow, IEEE*

***Abstract*— Autonomous driving technology has made remarkable progress in recent years, revolutionizing transportation systems and paving the way for safer and more efficient journeys. One of the critical challenges in developing fully autonomous vehicles is accurate perception of the surrounding environment. Radar sensor networks provide a capability for robust environmental detection. It become apparent that the principle of a synthetic aperture radar (SAR) can be employed not only in the field of earth observation but also increasingly in the field of autonomous driving. With the help of radar sensors mounted on vehicles, huge synthetic apertures can be created and thus a high angular resolution is achieved, which ultimately allows detailed images to be obtained. Increasing image quality, however, also increases the demands on position accuracy and thus the localization of the vehicle in the map. Since relative localization accuracies in the millimeter range over long trajectories cannot be achieved with conventional Global Navigation Satellite Systems (GNSS) so-called simultaneous localization and mapping (SLAM) algorithms are often employed. This paper presents a purely radar-based SLAM algorithm, which allows high-resolution SAR processing in the automotive frequency domain of 77 GHz. The presented algorithm is evaluated by measurements for trajectories with a length of up to 500 m and a measurement duration of more than two minutes.**

***Index Terms*— Chirp-sequence radar sensors, radar imaging sensors, radar sensor networks, synthetic aperture radar, SAR, simultaneous localization and mapping, SLAM.**

I. Introduction

The field of autonomous driving is advancing rapidly, thus putting increasing demands on environment recognition and sensor systems. A variety of sensor systems can be used for this purpose, each with its own strengths and weaknesses. LIDAR systems and cameras offer high angular resolution [1], but they lack robustness in adverse weather conditions [2]. Radar systems, on the other hand, have lower angular resolution but are more robust to weather conditions. In recent years, the use of radar sensor networks has offered significant advantages in terms of detection capability and accuracy. References [3], [4], and [5], allowing these sensors to be utilized for mapping as well.

The target information obtained from radar raw data can be used to create environmental maps, e.g. grid maps [6]. These offer the advantage of a cell-based representation of the environment, which makes them efficient to process. Grid maps can be generated either from target lists (T-GM) or from raw data (R-GM). Target list-based mapping approaches can be divided into amplitude grid maps and probabilistic occupancy grid maps. Probabilistic occupancy grid maps have been shown to provide a more robust representation of the environment with significantly increased resolution.

One way to represent the environment with a much higher resolution without changing the hardware is to use synthetic aperture radar (SAR). SAR processing of the raw data from a mm-wave radar sensor mounted on a moving vehicle can create a larger aperture, which results in increased angular accuracy compared to target list-based mapping approaches [7], [8], [9]. Since SAR processing also represents the environment in a cell-based manner, it is called a raw databased grid map (R-GM). SAR maps provide a much more accurate representation of the environment than target list-based grid maps due to the huge aperture spanned [10], [11]. Target list-based grid maps, on the other hand, are very computationally efficient and can be implemented in real time without much hardware overhead.

Although T-GMs and R-GMs represent the map differently, their processing has one major common feature – the position of the radar sensors and the vehicle must be known at each measurement interval. While the position of the radar sensors at each frame must be known for T-GMs, the position of the radar sensors at each transmit interval must be known for SAR processing. T-GMs have a lower localization requirement due to their lower maximum map resolution, which results in a position accuracy of 3 cm being sufficient to obtain an artifact-free image of the environment. Nowadays, such position accuracy in the lower centimeter range can be achieved with a differential Global Navigation Satellite Systems (GNSS) [12]. In SAR processing, however, the range-compressed data must be phase-corrected according to the position of the radar sensor. Therefore, in theory, a relative accuracy of less than $\lambda/4$ should be maintained. For radar sensors typically used in automotive applications with a frequency of 77 GHz, the maximum localization error corresponds to 1.8 mm. This accuracy cannot be achieved

Manuscript received 28 July 2023; revised 18 October 2023 and 20 December 2023; accepted 25 December 2023. Date of publication 28 December 2023; date of current version 5 January 2024. *(Corresponding author: Timo Grebner.)*

The authors are with the Institute of Microwave Engineering, Ulm University, 89081 Ulm, Germany (e-mail: timo.grebner@uni-ulm.de).

Digital Object Identifier 10.1109/TRS.2023.3347734

nowadays with ordinary GNSS. Furthermore, dense cloud conditions, high buildings or driving in tunnels or underground garages impede satellite-based localization and the localization accuracy is significantly degraded or fails completely. Internal measurement systems such as an internal measurement unit (IMU) or a wheel speed sensor have the advantage of being able to measure the speed of the vehicle independently of the external environment. However, their suitability for long-term localization is limited. The speeds are integrated according to dead reckoning and thus a small speed error leads to a large position error for long trajectories. Another problem of all systems is on the one hand the expensive price and on the other hand the time synchronization between the radar sensors and the corresponding localization systems, which must be in the range of a few microseconds for SAR in order to guarantee a coherent SAR processing.

The problem of insufficient position accuracy for SAR processing in the frequency range of 77 GHz is well known and is being attempted to be solved with the aid of various algorithms. At the early stage of the research the radar sensor was placed at thousands of different positions for this purpose and a measurement was performed at each position to span a synthetic aperture [13]. Following on these results, the radar sensor was mounted on a moving vehicle to generate a synthetic aperture while driving. In this process, the trajectory length could be extended stepwise from a few meters [9], [14], [15] up to 75 m to generate high-resolution SAR images [16], [17], [18]. Although ground truth information was a prerequisite, the velocity errors had to be compensated with different autofocusing methods to obtain a sharp image of the environment. As a consequence, it could be shown that an ego-motion based dead reckoning exclusively from the radar sensors used allows a trajectory estimation on the basis of which a SAR processing is possible [8], [19], [20]. However, all these methods allow SAR processing only for rectilinear trajectories and limited coherence.

Therefore, the paper describes an accurate position estimation algorithm based solely on radar data, allowing SAR processing over long and curved trajectories. This enables localization completely independent of external systems such as GNSS or an IMU. For this purpose, different algorithms like ego-motion-based dead-reckoning and scan-matching methods are fused using a graph-based SLAM algorithm. Due to the long trajectory and the high requirements on accuracy, cross-frame constraints are added to the graph. Thus, landmarks from different spatial directions are fused with each other. To avoid increasing the computational complexity of the already compute-intensive SLAM further, the SLAM is processed in combination with T-GM. Furthermore, another advantage of applying scan-matching to T-GM rather than directly to SAR imaging is the decoupling of the image quality from the ego-motion estimation. Once the trajectory is estimated, the trajectory is used for the significantly higher resolved SAR processing.

The paper is organized as follows: The sensor setup and the radar architecture are presented in Section II. In Section III the basic structure of the signal processing chain is described. In Section IV the conventions of SAR processing and probabilistic T-GM are derived. Subsequently, possibilities for radar-based ego-motion estimation and corresponding dead reckoning are derived in Section V part V-A, which are fused with presented scan-matching algorithms from Section V part V-B to set up the graph-based SLAM described in Section V part V-C. The presented algorithms are subsequently evaluated qualitatively and quantitatively in Section VI for different measurement runs on a parking lot with a length of several hundred meters.

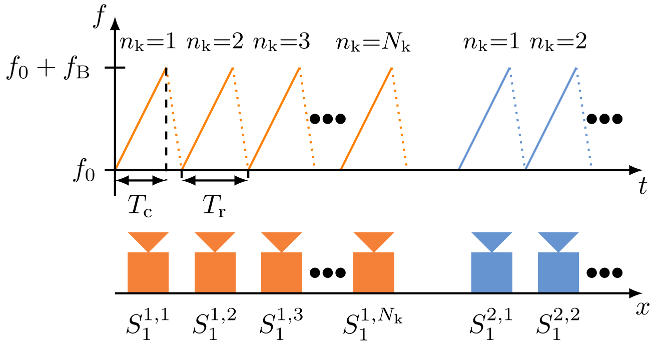


Fig. 1. Top: chirp-sequence modulation scheme for N_K ramps and two frames (1: orange, 2: blue). Bottom: sensor position for each transmitted ramp spanning a synthetic aperture [18].

II. Concept and System Architecture

The sensor configuration consists of incoherently-networked chirp-sequence radar sensors mounted on a non-stationary vehicle. The chirp-sequence radar sensors have multiple transmit and receive antennas enabling multiple input multiple output (MIMO) operation. The radar sensors emit frequency modulated continuous wave (FMCW) ramps. These are characterized by the bandwidth f_B, the up-chirp time T_c, the ramp repetition time T_r and the start frequency f_0 [21], [22]. The radar sensors are frame-based and emit N_k ramps within each of the N_f frames. This notation and the modulation scheme are shown in Fig. 1.

The time signal s_Tx of the emitted frequency ramps as shown in Fig. 1 are calculated by the equation [23], [24]

$$s_\mathrm{Tx}(t) \propto e^{\mathrm{j}\left(2\pi f_0 t + \pi\kappa t^2 + \phi_0\right)}. \tag{1}$$

For comprehensibility, both the amplitude in (1) and the counting index n_k are neglected. κ describes the slope of the frequency ramp which results from the ratio between the bandwidth f_B and the up-chirp time T_c. ϕ_0 describes the start phase of the emitted chirp. This generated transmit signal is emitted with the help of the transmit antenna, reflected at the target and then received again with the help of the receive antenna. For this, the signal in air needs a transit time of $\tau = {}^{2r}/_{c_0}$, where r describes the distance between the sensor and the target. Thus, the received signal corresponds to the time-delayed transmitted signal and is represented by

$$s_\mathrm{Rx}(t) \propto e^{\mathrm{j}\left(2\pi f_0 (t-\tau) + \pi\kappa (t-\tau)^2 + \phi_0\right)}. \tag{2}$$

Due to the radar architecture, the transmit signal and the receive signal are mixed analogously and lowpass filtered,

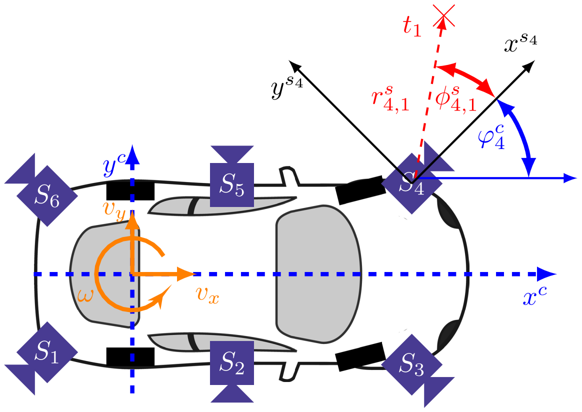


Fig. 2. Radar network with $N_\mathrm{r} = 6$ radar sensors in violet with different orientations. Vehicle coordinate system $\mathbf{P}^\mathrm{c} = \left[x^\mathrm{c},\, y^\mathrm{c}\right]$ in blue, dashed and corresponding velocity components in orange, solid lines [18], [25].

resulting in the intermediate frequency (IF) signal

$$s_\mathrm{IF}(t) \propto e^{\mathrm{j}\left(2\pi f_0 \tau + 2\pi\kappa t\tau - \pi\kappa\tau^2\right)}. \tag{3}$$

For each of the N_k emitted ramps and N_f frames, this IF signal is subsequently processed as described in Section III.

The radar sensor network consists of any number of sensors and is sketched for $N_\mathrm{r} = 6$ sensors in Fig. 2. The vehicle coordinate system $\left[x^\mathrm{c},\, y^\mathrm{c}\right]$ is shown in blue. The velocity vectors for all degrees of freedom (DoF) is shown in orange. These DoFs include the vehicle velocity in the local x and the local y directions, as well as the yaw rate ω acting at the center of the rear axle.

III. Processing Chain

Based solely on the raw data s_IF of all radar sensors, a high-resolution SAR image is processed. The algorithm used for this is shown in Fig. 3 for N_r sensors, and N_f frames. For clarity, each frame consists in the sketch of only $N_\mathrm{k} = 3$ ramps as opposed to $N_\mathrm{k} = 384$ as in the later evaluated measurement.

In the first step, the signals s_IF of all radar sensors are digitized and transformed independently into the frequency domain to obtain the range-compressed fast-time samples (🟦). Based on these fast-time samples, SAR processing is performed afterwards (🟩) as described in Section IV-A. The problem in this case is the localization and therefore the knowledge about the trajectory of the vehicle. To determine this trajectory a SLAM algorithm is applied (🟨). A graph-based SLAM is employed in this case, which determines the most likely trajectory based on the fusion of ego-motion estimation and kaze feature-based scan matching. However, this SLAM does not optimize the SAR-processed environment but a robust and significantly more efficient occupancy target-list based grid map as described in Section IV-B. To create such a target list based occupancy grid map the target lists have to be determined independently for each sensor (🟦). For this, common signal processing techniques (windowing, slow-time Fourier transform, constant false alarm rate (CFAR), and angle of arrival (AoA) estimation) are applied to the IF data of all N_k emitted ramps of a frame. Thus, with each of the N_r chirp-sequence radar sensors used, the azimuth angle $\phi^\mathrm{s}_{n_\mathrm{r},n_\mathrm{t}}$, the range $r^s_{n_\mathrm{r},n_\mathrm{t}}$, and the radial velocity $v^\mathrm{r}_{n_\mathrm{r},n_\mathrm{t}}$ in the local sensor coordinate system (^{s}) is determined for each n_t-th target, where N_t describes the number of targets detected.

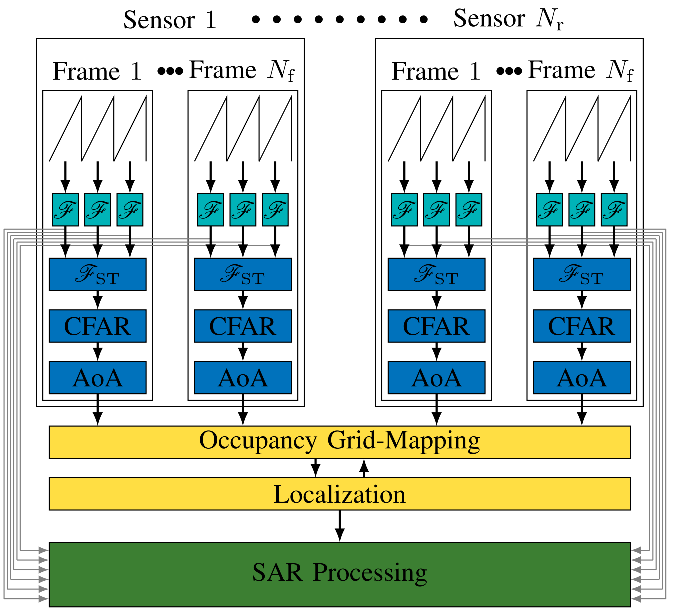


Fig. 3. Signal flow chart of the target list-based (🟦) SLAM algorithm (🟨) for creating high-resolution SAR images (🟩).

Using the occupancy grid map (OGM) for the SLAM algorithm enables velocity-independent environment mapping for one frame, reducing the complexity of the computationally expensive algorithm.

IV. Mapping

In general, for this purpose, the environment $\mathbf{M}$ to be mapped is gridded into N_i cells. Subsequently, for each cell m_i, either the occupancy probability is computed based on OGM processing or the received power is computed based on SAR processing. Thus, either an efficient OGM or a high-resolution SAR image can be processed based on the same radar data. Due to the efficient processing, the SLAM from Section V-C is processed based on the OGM. The subsequent high-resolution environment mapping is created using SAR processing based on the SLAM estimated trajectory.

A. Synthetic Aperture Radar (SAR)

In general, there are many different methods to generate SAR images of the environment. Basically, a distinction is made between algorithms in the frequency domain and algorithms in the time domain [26], [27], [28]. The advantage of the time-domain algorithms lies in their universal application range since the trajectory does not have to meet any special requirement like linearity. Therefore, the backprojection (BP) algorithm, which is operating in the time domain, is applied in the following.

For each cell m_i of the environment to be imaged $\mathbf{M}$, the measured range information is phase corrected according to the range to be evaluated. This corresponds to a matched-filter-based phase correction. If the cell to be evaluated m_i represents a target, the signals of all N_k ramps overlap constructively after phase correction, whereas these pointers overlap randomly if the cell represents noise.

According to Fig. 3, range compression is first performed based on the fast time samples $\mathbf{S}_{\mathrm{IF}}^{n_{\mathrm{f}},n_{\mathrm{k}}}$, which describe the sampled IF signal $s_{\mathrm{IF}}(t)$ from (3). For clarity, the time t_{k} is used in the following to describe the time of an emitted ramp. In contrast, t_{f} describes the time of a frame, which is relevant for the calculation of grid maps. This results in the following assignment:

$$\mathbf{S}_{\mathrm{IF}}^{n_{\mathrm{f}},n_{\mathrm{k}}} = \mathbf{S}_{\mathrm{IF}}^{t_{\mathrm{k}}} \tag{4}$$

The range-compressed data corresponds to the transformation of the fast-time samples into the frequency domain and is described with a Fourier transformation

$$\mathbf{A}^{t_{\mathrm{k}}} = \mathscr{F}\{\mathbf{S}_{\mathrm{IF}}^{t_{\mathrm{k}}}\}. \tag{5}$$

where $\mathbf{A}^{t_{\mathrm{k}}}$ is the range information vector at time t_{k} with

$$\mathbf{A}^{t_{\mathrm{k}}} = \left[A_1 \ A_2 \ \ldots \ A_{\mathrm{N_s}}\right]^{\mathrm{T}} \tag{6}$$

N_{s} corresponds to the number of samples of a frequency ramp. As long as no zero-padding is performed and IQ sampling is applied, this also corresponds to the length of the frequency vector. Each entry of the vector $\mathbf{A}^{t_{\mathrm{k}}}$ corresponds to a complex value, which is assigned to a certain frequency and thus to a certain distance $\mathbf{r}$.

Based on the range information $\mathbf{r}$ and the assigned complex amplitudes $\mathbf{A}^{t_{\mathrm{k}}}$ a SAR processing with the BP algorithm is performed for the environment $\mathbf{M}$ to be imaged. To each of the cells a received power $\mathbf{G}(m_i)$ is assigned. This complex value is iteratively calculated based on all measurements (Ξ) and all position data (χ) using the backprojection algorithm [29], [30], [31].

$$\begin{aligned}\mathbf{G}\big(m_i|\Xi_{1:t_{\mathrm{k}}}, \chi_{1:t_{\mathrm{k}}}\big) &= \mathbf{G}\big(m_i|\Xi_{1:t_{\mathrm{k}}-1}, \chi_{1:t_{\mathrm{k}}-1}\big)\\ &\quad + \mathbf{A}\big(m_i|\Xi_{t_{\mathrm{k}}}, \chi_{t_{\mathrm{k}}}\big) \cdot \exp(j \cdot \Phi).\end{aligned} \tag{7}$$

The received power of the cell $\mathbf{G}(m_i)$ at measurement time t_{k} is composed of the cell's received power at time $t_{\mathrm{k}}-1$ and the phase-corrected measured value of the current measurement $\mathbf{A}^{t_{\mathrm{k}}}$. Φ describes a distance-dependent phase correction to ensure constructive superposition for targets. The assignment of the correct frequency bin of the current measurement $\mathbf{A}^{t_{\mathrm{k}}}$ to the cell m_i is described with

$$\mathbf{A}\big(m_i|\Xi_{\mathrm{t_k}}, \chi_{\mathrm{t_k}}\big) = \mathbf{A}^{t_{\mathrm{k}}}\big(\mathbf{u}\big(m_i|\Xi_{\mathrm{t_k}}, \chi_{\mathrm{t_k}}\big)\big), \tag{8}$$

where $\mathbf{A}^{t_{\mathrm{k}}}\big(\mathbf{u}\big(m_i|\Xi_{\mathrm{t_k}}, \chi_{\mathrm{t_k}}\big)\big)$ describes the $\mathbf{u}$-th entry of the vector $\mathbf{A}^{t_{\mathrm{k}}}$. The position index $\mathbf{u}$ for the vector $\mathbf{A}^{t_{\mathrm{k}}}$ is given by the following ratio rounded to the nearest integer.

$$\mathbf{u}\big(m_i|\Xi_{\mathrm{t_k}}, \chi_{\mathrm{t_k}}\big) = \left\lfloor \frac{d_{m_i}}{\Delta r} \right\rceil. \tag{9}$$

In this case d_{m_i} describes the distance between the n_{r}-th radar sensor at the global ($^{\mathrm{g}}$) position $\mathbf{P}^{\mathrm{g}} = [P_{\mathrm{x}}^{\mathrm{g}}, \ P_{\mathrm{y}}^{\mathrm{g}}]$ and the cell m_i.

The final SAR image is processed exclusively in the xy-plane at the height of the radar sensors, given that the radar sensors, along with the vehicle's motion, cover a synthetic aperture limited to the xy-plane.

According to (8), each cell of the map to be imaged is assigned a distance and thus also a complex-value received power according to (6). This complex pointer must be rotated around the phase Φ according to the backprojection algorithm (7) to obtain a constructive or destructive overlay.

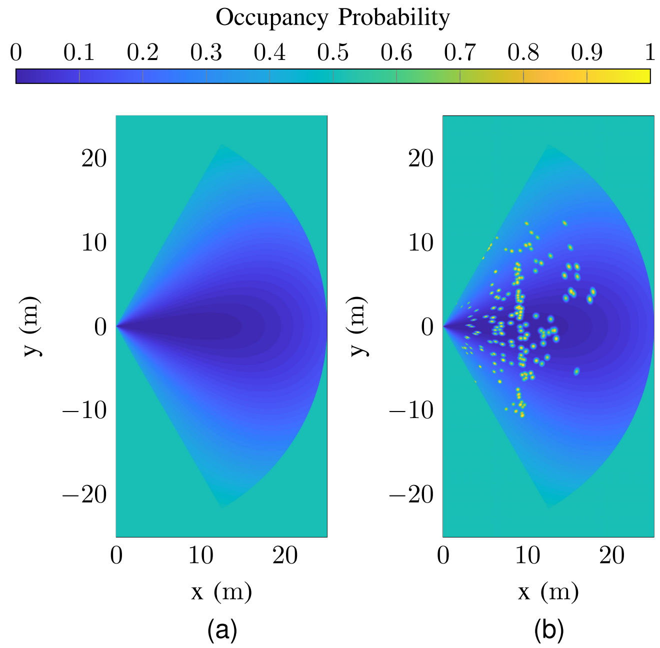


Fig. 4. Representation of the free space model only (4a) and the free space model + target model (4b) for an exemplary measurement.

$$\Phi = -\mathrm{j}2\pi\frac{f_0}{c_0} \cdot 2\mathbf{r}\big(m_i|\chi_{\mathrm{t_k}}\big). \tag{10}$$

These steps are performed for each of the N_{i} cells of the rasterized map, as well as for all N_{k} emitted ramps of each emitted frame. The movement of the vehicle leads to a continuously changing sensor position which results in a large synthetic aperture. As soon as a cell represents a target, all complex-valued received powers constructively overlap, whereas the received powers overlap randomly if the respective cell does not represent a target.

B. Occupancy Grid Mapping

The creation of target list-based grid maps is fundamentally different from the creation of SAR mappings. While SAR processing is a ramp-based processing, OGM processing is a frame-based processing. According to Fig. 3, all ramps of a frame are taken to obtain a target list. Subsequently, the targets are added to the map $\mathbf{M}$ according to their probabilistic properties.

Each of the N_i cells of the map $\mathbf{M}$ represents a probability in OGM and not an integrated received power as it is the case in SAR processing. The processing of the OGMs is fundamentally divided into three steps:

- modeling of free-space
- calculating target probability
- updating probabilistic map

These steps are described below with reference to [32], [33], and [34]. The goal in this case is to determine the occupancy probability $\mathbf{p}\big(m_i|\Xi_{1:t_{\mathrm{f}}}, \chi_{1:t_{\mathrm{f}}}\big)$ of cell m_i at time t_{f} based on all measurements $\Xi_{1:t_{\mathrm{f}}}$ and all pose information $\chi_{1:t_{\mathrm{f}}}$ up to the current time t_{f}.

The first step of probabilistic OGM processing involves modeling the free space. This free-space model describes the occupancy probability $\mathbf{p}_{\mathrm{FS}}$ of each cell if no target is detected with the CFAR at that position and is both range and angle dependent. While the free space probability is modeled proportional to the distance squared, it is approximated with a cosine in the azimuth plane.

$$\begin{aligned}&\mathbf{p}_{\mathrm{FS}}(m_i)\\&\quad=\frac{0.5-c_1}{2}\cdot\left(\frac{d_{m_i}^2}{R_{\max}^2}+\left(1-\cos\left(\frac{\pi}{2}\frac{\phi_{m_i}}{\mathrm{FoV}_\phi}\right)\right)\right)+c_1\end{aligned}\tag{11}$$

The parameter c_1 describes the minimum probability of the free space model and has to be chosen according to the radar sensor and the applied CFAR and angle estimation algorithm. For the radar sensors, $c_1 = 0.3$ is used to avoid suppressing missing detections due to clustered CFAR detection. The variables $R_{\max}$ and FoV_ϕ describe the maximum measurable distance of the radar sensor and the beam angle of the utilized antenna. According to the properties of the free-space model, this probability $\mathbf{p}_{\mathrm{FS}}\left(m_i|x_{1:t_{\mathrm{F}}}\right)$ must generally lie in the interval [0; 0.5] and must be normalized accordingly. The variable ϕ_{m_i} describes the azimuth angle in sensor coordinates between the sensor and the cell m_i to be evaluated. The free-space model is shown in Fig. 4a for a sensor at position $\mathbf{P}^{\mathrm{g}} = [0\,\mathrm{m}\ 0\,\mathrm{m}]$.

Next, for each of the N_{t} detected targets, the target probability p_{T} must be determined. This is modeled based on the measured distance $r^s_{n_r,n_t}$, the azimuth angle $\phi^{\mathrm{s}}_{n_r,n_t}$, and the SNR. In the following, the indices are neglected for clarity.

$$p_r = 1-\left(0.5\cdot\frac{r^s}{R_{\max}}\right)\tag{12}$$

$$p_{\phi^s} = \cos\left(\frac{\arccos(0.5)}{\mathrm{FoV}_\phi}\cdot\phi^s\right)\tag{13}$$

$$p_{\mathrm{SNR}} = 1-(1-0.5)\cdot e^{-h\cdot\mathrm{SNR}}\tag{14}$$

The variable h is taken to vary the slope of the function p_{SNR} which adapts the function to the radar sensor employed. The target probability p_{T} describes the arithmetic mean of all three sub-probabilities. The Gaussian distribution is used to model the probability of each target within the interval [0.5; 1], accounting for the uncertainties σ_r, and σ_φ associated with the radar sensor.

$$f_K(\mathbf{k}) = \frac{1}{\sqrt{(2\pi)^3\det(\boldsymbol{\Sigma})}}e^{\left(-\frac{1}{2}(\mathbf{k}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{k}-\boldsymbol{\mu})\right)}\tag{15}$$

$$\mathbf{p}_{\mathrm{T}}(m_i) = \begin{cases}0,\ (|\mathbf{k}-\boldsymbol{\mu}|) > \mathrm{diag}\left(3\sqrt{\boldsymbol{\Sigma}}\right)\\ \dfrac{f_K(\mathbf{k})}{\max(f_K(\mathbf{k}))}\cdot(p_{\mathrm{T}}-0.5)+0.5,\\ \qquad\text{otherwise.}\end{cases}\tag{16}$$

The matrix $\boldsymbol{\Sigma}$, represents the covariance matrix of the measurement Ξ, while the extracted target information is denoted as $\boldsymbol{\mu}$. The vector $\mathbf{k}$ specifies the cell that is being evaluated.

$$\mathbf{k} = \left[m_{i,\mathrm{r}}\ m_{i,\phi}\right].\tag{17}$$

Consequently, the probability distribution of each detected target extends across multiple cells in the free-space map. The registration of each of the N_t detected targets is performed, allowing the probability of a cell m_i at the current measurement time t_{f} to be characterized by

$$\mathbf{p}\left(m_i|\Xi_{t_{\mathrm{f}}},\chi_{t_{\mathrm{f}}}\right) = \max\left(\mathbf{p}_{\mathrm{T}}(m_i),\mathbf{p}_{\mathrm{FS}}(m_i)\right).\tag{18}$$

If this procedure is performed for many measurements, an OGM can be calculated. For this, the probabilities of the current measurement $\mathbf{p}\left(m_i|\Xi_{t_{\mathrm{f}}},\chi_{t_{\mathrm{f}}}\right)$ are combined with the probabilities of all previous measurement $\mathbf{p}\left(m_i|\Xi_{1:t_{\mathrm{f}}-1},\chi_{1:t_{\mathrm{f}}-1}\right)$ and all sensors.

$$\begin{aligned}&\mathbf{p}\left(m_i|\Xi_{1:t_{\mathrm{f}}},\chi_{1:t_{\mathrm{f}}}\right)\\&\quad=\mathbf{p}\left(m_i|\Xi_{t_{\mathrm{f}}},\chi_{t_{\mathrm{f}}}\right)\circledcirc\mathbf{p}\left(m_i|\Xi_{1:t_{\mathrm{f}}-1},\chi_{1:t_{\mathrm{f}}-1}\right)\\&\quad=\frac{1}{\left(1+E_A^{-1}\cdot E_B^{-1}\cdot E_C^{-1}\right)}\end{aligned}\tag{19}$$

$$E_A = \frac{\mathbf{p}\left(m_i|\Xi_{t_{\mathrm{f}}},\chi_{t_{\mathrm{f}}}\right)}{1-\mathbf{p}\left(m_i|\Xi_{t_{\mathrm{f}}},\chi_{t_{\mathrm{f}}}\right)}\tag{20}$$

$$E_B = \frac{\mathbf{p}\left(m_i|\Xi_{1:t_{\mathrm{f}}-1},\chi_{1:t_{\mathrm{f}}-1}\right)}{1-\mathbf{p}\left(m_i|\Xi_{1:t_{\mathrm{f}}-1},\chi_{1:t_{\mathrm{f}}-1}\right)}\tag{21}$$

$$E_C = \frac{1-\mathbf{p}(m_i)}{\mathbf{p}(m_i)}\tag{22}$$

The symbol $\circledcirc$ describes the probabilistic update step corresponding to [35].

V. Self-Localization

The mapping algorithms described in Section IV enable an environment mapping once the global vehicle pose $\mathbf{P}^{\mathrm{g}}_{\mathrm{v}} = \left[P_{\mathrm{v,x}}\ P_{\mathrm{v,y}}\ P_{\mathrm{v,\Phi}}\right]$ is known. This vector consists of the x and y position of the vehicle and the orientation $P_{\mathrm{v,\Phi}}$. In general, however, this vehicle pose $\mathbf{P}^{\mathrm{g}}_{\mathrm{v}}$ is not known and must be determined which is done below using a graph-based SLAM method for an unknown environment $\mathbf{M}$. The graph-based SLAM algorithm consists of three steps:

- radar-based ego motion estimation
- scan matching
- solving the graph equation

These steps are derived below to enable stand-alone radar-based self-localization.

A. Ego-Motion Estimation

In [36], [37], [38], [39], [40], and [41] it was shown that three velocity components v_{x} and v_{y} as well as the yaw rate ω (corresponding to Fig. 2) can be determined using a radar sensor network($N_{\mathrm{r}}\geq 2$) based on the target lists. For this purpose, the following model describing the relationship between the measured target information $v^{\mathrm{r}}_{\mathrm{n_r,n_t}}$, $\phi^{\mathrm{s}}_{\mathrm{n_r,n_t}}$ and the intrinsic velocity $\mathbf{V}_p$ is used:

$$\begin{aligned}\underbrace{\begin{bmatrix}-v^{\mathrm{r}}_{\mathrm{n_r},1}\\-v^{\mathrm{r}}_{\mathrm{n_r},2}\\\vdots\\-v^{\mathrm{r}}_{\mathrm{n_r},N_{\mathrm{t}}}\end{bmatrix}}_{\mathbf{V}^{\mathrm{r}}_{\mathrm{n_r}}} &= \underbrace{\begin{bmatrix}\cos\left(\phi^{\mathrm{c}}_{\mathrm{n_r},1}\right) & \sin\left(\phi^{\mathrm{c}}_{\mathrm{n_r},1}\right)\\\cos\left(\phi^{\mathrm{c}}_{\mathrm{n_r},2}\right) & \sin\left(\phi^{\mathrm{c}}_{\mathrm{n_r},2}\right)\\\vdots & \\\cos\left(\phi^{\mathrm{c}}_{\mathrm{n_r},N_{\mathrm{t}}}\right) & \sin\left(\phi^{\mathrm{c}}_{\mathrm{n_r}N_{\mathrm{t}}}\right)\end{bmatrix}}_{\mathbf{D}_{\mathrm{n_r}}}\\&\quad\cdot\underbrace{\begin{bmatrix}-y^{\mathrm{c}}_{\mathrm{n_r}} & 1 & 0\\x^{\mathrm{c}}_{\mathrm{n_r}} & 0 & 1\end{bmatrix}}_{\mathbf{S}_{\mathrm{n_r}}}\cdot\begin{bmatrix}\omega\\v_x\\v_y\end{bmatrix}.\end{aligned}\tag{23}$$

In this case, $x^{\mathrm{c}}_{\mathrm{n_r}}$ and $y^{\mathrm{c}}_{\mathrm{n_r}}$ denote the sensor positions and $\phi^{\mathrm{c}}_{\mathrm{n_r,n_t}}$ denote the incidence angles relative to the vehicle coordinate system. These are calculated with

$$\phi^{\mathrm{c}}_{\mathrm{n_r,n_t}} = \phi^{\mathrm{s}}_{\mathrm{n_r,n_t}} + \varphi^{\mathrm{c}}_{\mathrm{n_r}} \tag{24}$$

As soon as more than two radar sensors are employed on the vehicle, the equations can be transformed into a common system equation, and thus the velocity vector $\mathbf{V_p}$ can be uniquely determined.

$$\underbrace{\begin{bmatrix} \mathbf{V^r_1} \\ \mathbf{V^r_2} \\ \vdots \\ \mathbf{V}^r_{N_\mathrm{r}} \end{bmatrix}}_{\mathbf{V^r}} = \underbrace{\begin{bmatrix} \mathbf{D_1} \cdot \mathbf{S_1} \\ \mathbf{D_2} \cdot \mathbf{S_2} \\ \vdots \\ \mathbf{D}_{N_\mathrm{r}} \cdot \mathbf{S}_{N_\mathrm{r}} \end{bmatrix}}_{\mathbf{Q}} \cdot \underbrace{\begin{bmatrix} \omega \\ v_x \\ v_y \end{bmatrix}}_{\mathbf{V_p}}. \tag{25}$$

Subsequently, (25) is transformed according to the searched velocity vector $\mathbf{V_p}$, for example, using a Moore-Penrose inverse of $\mathbf{Q}$.

$$\tilde{\mathbf{V}}_\mathrm{p} = \mathbf{Q}^+ \cdot \mathbf{V^r} \tag{26}$$

The velocity of the vehicle is calculated by considering stationary targets, as explained in (25). To accurately estimate the vehicle's speed, it is necessary for at least two sensors to detect a minimum of three targets, enabling velocity estimation for all three degrees of freedom (DoFs). As (25) applies exclusively to stationary targets, any non-stationary targets or false detections need to be eliminated. To filter out these outliers that do not conform to the expected motion model, a random sample consensus (RANSAC) algorithm [42] is employed.

1) RANSAC-Filtering: In order to ensure a dependable and resilient estimation of ego-motion, it is essential to eliminate outliers. One effective approach is to utilize an iterative RANSAC algorithm with a specified number of iteration steps, denoted as N_it. During each iteration step $n_\mathrm{it} \leq N_\mathrm{it}$ of the RANSAC algorithm, the velocity model $\mathbf{V^{n_{it}}_p}$ is estimated by selecting three targets randomly from the target lists $(\mathrm{TL}_1, \ldots, \mathrm{TL}_{n_\mathrm{t}})$. Subsequently, the velocity difference $\mathbf{D}_\mathrm{V}$ between $\mathbf{V^{n_{it}}_p}$ and all measured radial velocities $\mathbf{V}_\mathrm{r}$ is calculated

$$\mathbf{D}^{n_\mathrm{it}}_\mathrm{V} = \left| \mathbf{Q} \cdot \mathbf{V^{n_{it}}_p} - \mathbf{V_r} \right|. \tag{27}$$

Following that, the accuracy of the velocity estimation is evaluated based on the number of inliers. Inliers represent targets with velocity errors below a suitable threshold T_v relative to the currently estimated motion model. It is crucial to select a threshold that encompasses the majority of real stationary targets, taking into account measurement inaccuracies and noise. To ensure this, a threshold of $T_\mathrm{v} = 0.7\,\frac{\mathrm{m}}{\mathrm{s}}$ is chosen for the evaluation. This iterative RANSAC process is outlined as follows:

$$K_\mathrm{it} = \underset{n_\mathrm{it} \leq N_\mathrm{it}}{\arg\max} \left(\sum_{j=1}^{J} \mathbb{1}_{\left(\mathbf{D}^{n_\mathrm{it}}_{\mathrm{V},j} < T_\mathrm{v}\right)} \right) \tag{28}$$

where $\mathbf{D}^{n_\mathrm{it}}_{\mathrm{V},j}$ describes the j-th element ($j \leq J$, $J = O(\mathbf{D}^{n_\mathrm{it}}_{\mathrm{V},j})$ where O describes the cardinality and therefore the number of detections) of the vector $\mathbf{D}^{n_\mathrm{it}}_\mathrm{V}$ with

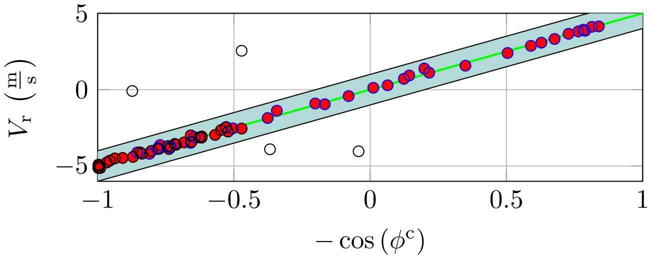


Fig. 5. Representation of the ego-motion estimation based on detected targets of a simulated straight line drive for two sensors with the orientation $\boldsymbol{\varphi}^\mathrm{c} = \left[\varphi^\mathrm{c}_1\ \varphi^\mathrm{c}_2\right] = [0°\ 90°]$ and the corresponding inliers.

$$\mathbf{D}^{n_\mathrm{it}}_\mathrm{V} = \left[D^{n_\mathrm{it}}_{\mathrm{V},1}\ D^{n_\mathrm{it}}_{\mathrm{V},2}\ \ldots\ D^{n_\mathrm{it}}_{\mathrm{V},J} \right]^\mathrm{T} \tag{29}$$

and $\mathbb{1}$ specifies the indicator function with:

$$\mathbb{1}_{\left(\mathbf{D}^{n_\mathrm{it}}_{\mathrm{V},j} < T\right)} = \begin{cases} 1 & \mathbf{D}^{n_\mathrm{it}}_{\mathrm{V},j} \leq T_\mathrm{v} \\ 0 & \text{otherwise.} \end{cases} \tag{30}$$

The model that provides the best fit to the current measurement, indicated by having the highest number of inliers, corresponds to the most probable velocity vector $\mathbf{V}^{K_\mathrm{it}}_\mathrm{p}$ of all N_it iteration steps.

As the RANSAC filtering relies on N_it iteration steps and three randomly selected targets, $\mathbf{V}^{K_\mathrm{it}}_\mathrm{p}$ represents the best estimation derived from three stationary targets rather than encompassing all stationary targets. In order to enhance the accuracy of velocity estimation, the vehicle's velocity is re-estimated, this time considering all targets that conform to the model within the threshold tolerance T_v for the velocity vector $\mathbf{V}^{K_\mathrm{it}}_\mathrm{p}$.

This principle is shown schematically for two sensors in Fig. 5. The two sensors have an orientation of $\boldsymbol{\varphi}^\mathrm{c} = \left[\varphi^\mathrm{c}_1\ \varphi^\mathrm{c}_2\right] = [0°\ 90°]$ and are color coded. The measurements selected as inliers by the RANSAC algorithm are shown as red-filled scatters. Those detected targets which do not correspond to the most probable velocity model are filled transparent. Based on the selected inliers, the most likely vehicle speed is estimated, which is shown in green (——).

2) Dead-Reckoning: Based on the estimated velocity vector $\tilde{V}_\mathbf{p}$, the vehicle pose $\mathbf{P}^\mathrm{g}_\mathrm{v} = \left[P_\mathrm{v,x}\ P_\mathrm{v,y}\ P_{\mathrm{v},\Phi} \right]$ is determined using the dead-reckoning algorithm. In this case, the pose at time $t{+}1$ is determined based on the vehicle position, vehicle velocity, and vehicle orientation at time t:

$$\begin{bmatrix} \tilde{P}^{t+1}_\mathrm{v,x} \\ \tilde{P}^{t+1}_\mathrm{v,y} \\ \tilde{P}^{t+1}_{\mathrm{v},\Phi} \end{bmatrix} = \begin{bmatrix} P^t_\mathrm{v,x} - r_\mathrm{v}\left(\sin\left(P^t_{\mathrm{v},\Phi} + \Phi^t_{\mathbf{V}_\mathrm{p}}\right) - \sin\left(\omega \cdot \mathrm{dt} + P^t_{\mathrm{v},\Phi} + \Phi^t_{\mathbf{V}_\mathrm{p}}\right)\right) \\ P^t_\mathrm{v,y} + r_\mathrm{v}\left(\cos\left(P^t_{\mathrm{v},\Phi} + \Phi^t_{\mathbf{V}_\mathrm{p}}\right) - \cos\left(\omega \cdot \mathrm{dt} + P^t_{\mathrm{v},\Phi} + \Phi^t_{\mathbf{V}_\mathrm{p}}\right)\right) \\ P^t_{\mathrm{v},\Phi} + \omega \cdot dt \end{bmatrix} \tag{31}$$

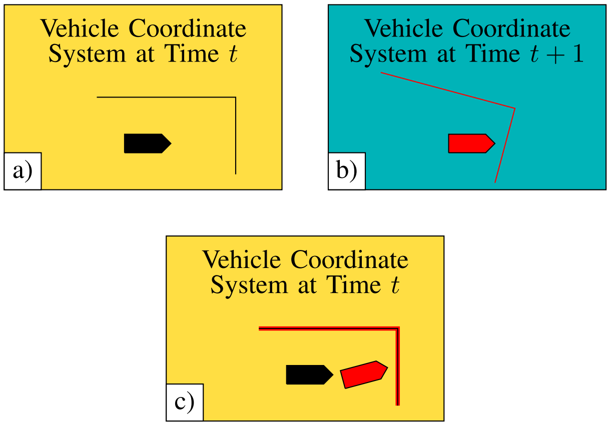

Vehicle Coordinate System at Time t
a)
Vehicle Coordinate System at Time $t+1$
b)
Vehicle Coordinate System at Time t
c)

Fig. 6. Illustration of the scan matching principle. a) measurement at time t, b) measurement at time t+1, and c) estimated transformation of the vehicle pose based on the reference image and the object image.

The variable r_{v} denotes the radius of the curve, which is given by

$$r_{\mathrm{v}} = \frac{\sqrt{v_x^2 + v_y^2}}{\omega} \tag{32}$$

The variable dt describes the time interval between two consecutive time steps and $\Phi_{\mathbf{v}_{\mathrm{p}}}$ the angle of the estimated velocity. Since the processing of grid maps is frame-based, the time interval dt in this case corresponds to the reciprocal of the frame rate of the radar sensors, which in the later measurement is 37 frames per second (fps).

Dead reckoning is a very simple way to estimate the a posteriori pose of a vehicle. However, since the pose estimation is based solely on velocity estimation, erroneous velocity estimations will be integrated and lead to erroneous vehicle poses for long trajectories. To minimize these errors, the pose is additionally determined by scan matching.

Dead reckoning is very well suited for short-time position estimation within a frame and thus for each transmitter ramp, which is important for SAR processing. Incorrect dead reckoning within a frame leads to a distorted image of the environment during SAR processing. However, since GMs process target lists, and are frame-based, the resulting map is independent of the ramp positions and thus independent of the dead reckoning. Since the dead reckoning and the scan matching should be as independent as possible for the later presented SLAM algorithm, this property is exploited and the scan matching is processed on the basis of the velocity independent OGMs.

B. Scan-Matching

Rather than using velocity information, scan matching estimates the pose change of the vehicle based on environment maps. The measurements from different times are compared with each other. The pose difference of these two measurements is subsequently converted into a transformation which determines a relative pose information between the two measurement times. This is sketched in Fig. 6.

The Fig. 6 consists of three sub-figures. The upper left figure shows the environment of a vehicle (⬛) in relation to the vehicle at time t. The vehicle is surrounded by a possible target, a wall, on the front side and on the left side. As soon as the vehicle is in motion and a second measurement is performed, at time $t+1$, a different relative environmental image results as shown in the upper right sub-image. The vehicle (🟥) is still at position $(0, 0)$ in the local vehicle coordinate system, but the targets have shifted. Based on this target shift, the pose change of the vehicle is determined in this section. If this pose estimation is performed by scan-matching methods, the result is the lower sub-image, which is again shown in the vehicle coordinate system at measurement time t. It is evident that the two walls are congruently matched, and the vehicle (⬛) has moved forward (🟥) during one frame and has gone through a left turn.

The aim is to estimate the transformation as precisely and robustly as possible. Since the sensor environment according to Section IV-B can be mapped very efficiently with an occupancy grid map for one frame, methods from image registration are applied for scan matching [43], whereas these have to be adapted according to the sensing properties of radar sensors.

In general, image registration describes a method for determining the transformation between two images, in the sensing case, of two images of the environment [43]. In this case, a basic distinction is made between the reference image (Fig. 6a) and the object image (Fig. 6b). According to Fig. 6c), the reference image is at a fixed position while the object image is shifted to the reference image as best as possible by means of a transformation. This is done with the help of automated feature detectors such as the robust kaze features [44], [45], [46]. In this process, scan matching is performed in three basic steps, which will be discussed in more detail below:

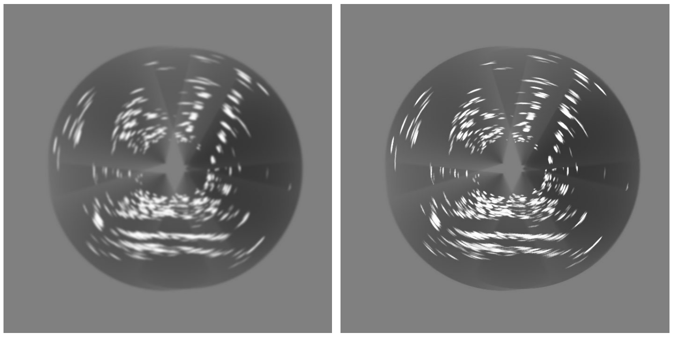
1) feature detection
2) feature description
3) determination of correspondences

1) Feature Detection: The detection of suitable features is elementary for an accurate scan-matching-based pose estimation, since all post-filtering steps are based on the detected features. As the subsequent algorithms have their origin in image processing, the occupancy grid maps are converted to gray-scale images. Each cell m_i in this case no longer describes the probability $\mathbf{p}(m_i)$ but the concentration $\mathbf{C}(m_i)$. To ensure robust and reliable feature detection, the image $\mathbf{C}(m_i)$ is simplified in several stages and will be referred to as scale space $\mathbf{W}$ in the following. This reduces noise and makes prominent structures such as edges or corners more prominent. In the simplest case, this could be achieved by convolving the image with a linear (Gaussian) kernel function $\mathbf{B}_{\mathrm{G}}(m_i)$.

$$\mathbf{W}(m_i, \sigma_k, s_k) = \mathbf{B}_{\mathrm{G}}(m_i, \sigma_k, s_k) * \mathbf{C}(m_i) \tag{33}$$

$$\mathbf{B}_{\mathrm{G}}(m_i, \sigma_k, s_k) = \frac{1}{2\pi\sigma_k} e^{-\frac{m_{i,x}^2 + m_{i,y}^2}{\sigma_k^2 s_k}} \tag{34}$$

The parameter σ_k describes the standard deviation of the Gaussian filter whereas the parameter s_k describes a factor for the modification of the Gaussian kernel. This parameter will be discussed in more detail later. In general, the image $\mathbf{C}$ is smoothed more for higher values of s_k, which results



(a) (b)

Fig. 7. Illustration of the convolution with a linear kernel function (a) and a non-linear kernel function (b).

in more noise suppression. This is exemplarily shown for a measurement in Fig. 7a.

The problem in this case, however, is that not only the noise of the image is smoothed by a convolution with a Gaussian kernel, but also all prominent structures, which are essential for a robust feature extraction. This is remedied by nonlinear kernel functions such as those used in kaze features [46]. The goal of this nonlinear kernel function is to suppress noise, but not to smooth details and structures. For this purpose, the intensity of the smoothing is adapted to the local image structure. To achieve this, a diffusion description method is applied.

Fick's law describes in general the diffusion and thus the energy-less smoothing process of concentrations. The concentration corresponds to the gray values $\mathbf{C}(m_i)$ of the image of the environment. By applying the continuity equation, the diffusion equation to be solved is obtained

$$\frac{\partial \mathbf{C}}{\partial t} = \nabla \cdot (\Pi(m_i, \nabla \mathbf{C}, t_\mathrm{s}) \cdot \nabla \mathbf{C}). \tag{35}$$

The variable of time t_s corresponds to a scaling factor, which has a direct effect on the diffusion. A higher value of t_s leads to a stronger smoothing and thus a stronger simplification of the image. This value will be discussed in more detail later. In order to achieve the desired nonlinear diffusion filtering, the conductivity function Π has to be determined as a function of the gradient, which can be done by the following equation

$$\Pi(m_i, \nabla \mathbf{C}, t_\mathrm{s}) = g(|\nabla \mathbf{C}_\sigma(m_i, t_\mathrm{s})|). \tag{36}$$

The luminance function $\mathbf{C}_\sigma$ represents the gradient of a Gaussian-filtered rendition, with standard deviation σ, of the initial image $\mathbf{C}$. The function g must be chosen according to the image [47]. Since the radar images according to Fig. 4 do not have clear edges and corners as it is the case for example with camera or LIDAR images, the function g is chosen accordingly to favor wide regions over smaller regions.

$$g = \frac{1}{1 + \frac{|\mathbf{C}_\sigma|^2}{k_\mathrm{c}^2}}. \tag{37}$$

The factor k_c is a contrast factor which influences the degree of diffusion. The diffusion (35) is subsequently solved numerically, since an analytical solution is not possible due to its complexity. One way to do this is by the linear-implicit method [48]. In this case, (35) is discretized and solved iteratively for each iteration it

$$\mathbf{C}^{it+1} = \left(\mathbf{I} - \tau \sum_{k_l=1}^{2} \mathbf{\Pi}'_{k_l}\left(\mathbf{C}^{it}\right)\right)^{-1} \mathbf{C}^{it} \tag{38}$$

The unit matrix is described by $\mathbf{I}$ and the discrete time step by τ. $\mathbf{\Pi}'$ is a matrix that encodes the image conductivities for each of the two dimensions. This result of a nonlinear kernel function is shown in Fig. 7b. Especially in comparison to the linear kernel function as shown in Fig. 7a, the different weighting of the diffusion in the nonlinear case is visible. Once prominent structures are detected, they are further highlighted and irrelevant structures are filtered out by Gaussian smoothing.

Based on the described diffusion (35), a nonlinear scale space (NSS) is spanned in the following. Based on this nonlinear scale space the kaze features are extracted. The nonlinear scale space corresponds to a two-dimensional matrix **NSS** with K_o columns, which will be called octaves in the following, and K_s rows, which will be called sub-levels in the following. Each matrix entry of the two-dimensional NSS contains a specific simplification of the image $\mathbf{C}_{\sigma_{n_\mathrm{NSS}}}$.

$$\mathbf{NSS}(k_\mathrm{o}, k_\mathrm{s}) = \mathbf{C}_{\sigma_{n_\mathrm{NSS}}} \tag{39}$$

The different simplifications are described by different standard deviations σ_{n_NSS}.

$$\sigma_{n_\mathrm{NSS}}(k_o, k_s) = \sigma_0 2^{k_o + \frac{k_s}{K_s}} \tag{40}$$

The parameter σ_0 is the base scale level. Based on this, the nonlinear scale space is constructed. Since the diffusion (35) is computed in time and not in discrete scale levels, σ_{n_NSS} these must be converted to a diffusion time t_{n_NSS} according to [46]. In the case of a Gaussian distributed scale space, convolving an image with a Gaussian kernel with standard deviation σ corresponds to image filtering for a specific time.

$$t_{n_\mathrm{NSS}} = \frac{1}{2}\sigma^2_{n_\mathrm{NSS}} \tag{41}$$

The advantage of the nonlinear scale space is the invariance of the detected features to noise and features, since the features are extracted into different simplifications and thus for different diffusions. Based on this NSS, the features are subsequently extracted. For this, a normalized Hessian matrix $\mathbf{C}_{\sigma_{n_\mathrm{NSS}},H}$ is computed for different image levels.

$$\mathbf{C}_{\sigma_{n_\mathrm{NSS}},H} = \sigma^2_{n_\mathrm{NSS}}\left(\frac{\partial}{\partial x}\frac{\partial}{\partial x}\mathbf{C}_{\sigma_{n_\mathrm{NSS}}}\frac{\partial}{\partial y}\frac{\partial}{\partial y}\mathbf{C}_{\sigma_{n_\mathrm{NSS}}} - \left(\frac{\partial}{\partial x}\frac{\partial}{\partial y}\mathbf{C}_{\sigma_{n_\mathrm{NSS}}}\right)^2\right) \tag{42}$$

As the amplitude of the spatial derivative decreases with increasing uncertainty σ_{n_NSS} it must be normalized [49]. The second order derivatives are approximated by a consecutive application of sharp filters [50]. To allow a robust estimation, detector responses are performed for different scale levels σ_{n_NSS}. Furthermore, the maximum search is not only performed spatially, and thus in the two-dimensional image, but also in cross-level between the different scale levels σ_{n_NSS}. The search

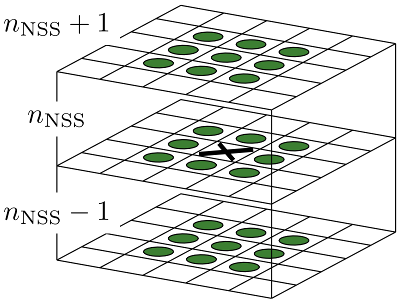


Fig. 8. Search for local maxima in multidimensional space for feature detection.

for maxima is performed in all levels excluding the first level $n_{\text{NSS}} = 0$ and the last level $n_{\text{NSS}} = K_{\text{o}} \cdot K_{\text{s}}$. The search for a local maxima is performed over a $\sigma_{n_{\text{NSS}}} \times \sigma_i$ large area on three levels ($n_{\text{NSS}} - 1, n_{\text{NSS}}, n_{\text{NSS}} + 1$). This is shown schematically in Fig. 8.

For this purpose, the maximum value is determined in this cuboid based on all cells, excluding the cell under test (CUT), according to Fig. 8. As soon as this minimum value is smaller than the gray value of the CUT, this CUT describes a local maximum and thus a kaze feature. Once a kaze feature is detected, the position of this feature is localized to sub-pixel accuracy according to [51] to ensure more accurate pose estimation. Overall, this procedure extracts all K_{kf} kaze features.

To further increase the robustness of feature detection, the grid map is not based on a single frame but on many frames. This allows structures to be mapped more clearly, which ensures that kaze features are detected more reliably. The fusion of several frames is based on dead reckoning. Therefore, the more frames are accumulated, the sharper the structures become, but the more dependent the scan matching is on dead reckoning. Due to the later fusion of dead reckoning and scan matching, both localization methods should be independent of each other. Chapter VI shows that an optimal value is around eight frames.

2) Feature Description and Matching: In order to form correspondences between the features found in each frame, they must be characterized. This is done by descriptors, which describe the environment of a feature.

The feature description is divided into two parts and is done sequentially for each of the K_{kf} detected kaze features. First, the orientation of each feature is estimated to make it rotation invariant. In the second step, the actual descriptor is built, which describes the environment of the feature. Based on this descriptor, correspondences between other features are subsequently determined to estimate the transformation and thus the relative vehicle pose.

In the first step, the orientation estimation, a circular area with a radius of $6\sigma_{n_{\text{NSS}}}$ is defined around the extracted kaze feature k_{kf}. In this area, the derivatives in the local x-direction $\frac{\partial}{\partial x}\mathbf{C}_{\sigma_{n_{\text{NSS}}}}$ and in the local y-direction $\frac{\partial}{\partial y}\mathbf{C}_{\sigma_{n_{\text{NSS}}}}$ are formed to determine the orientation of the derivative as shown in Fig. 9a.

Subsequently, these derivatives are weighted depending on their distance according to a Gaussian distribution with the standard deviation $\sigma = 2\sigma_{n_{\text{NSS}}}$. To find out the dominant

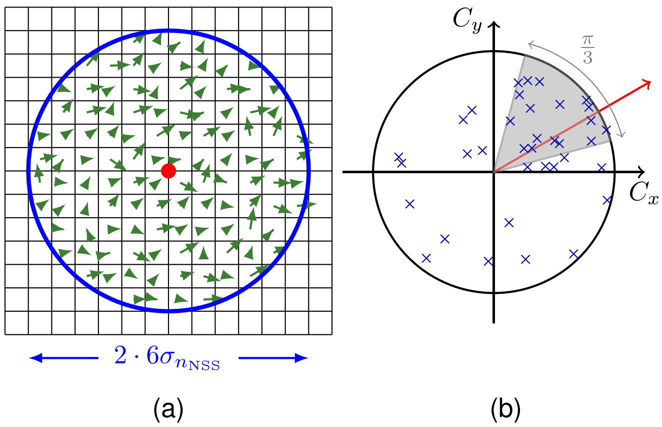


Fig. 9. Process of feature description. a) describes the local derivations in the x- and y-direction. b) describes the process of estimating the dominant orientation of the corresponding feature.

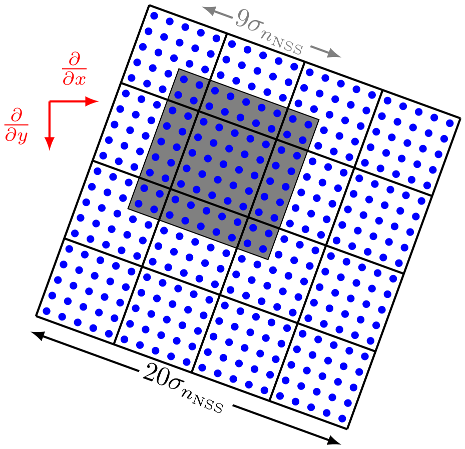

$9\sigma_{n_{\text{NSS}}}$
$\frac{\partial}{\partial x}$
$\frac{\partial}{\partial y}$
$20\sigma_{n_{\text{NSS}}}$

Fig. 10. Process of calculating the descriptor vector.

orientation $\Phi_{k_{\text{kf}}}$ of the feature these local gradients are represented as blue crosses in a vector space according to Fig. 9b.

The dominant orientation is subsequently determined using a sliding window method. The sliding window has a size of $\frac{\pi}{3}$ and is highlighted as a gray area in Fig. 9b. The maximum of the sliding window method corresponds to the dominant orientation of the feature and is shown as a red arrow in Fig. 9b.

Following the orientation determination, the environment of the feature is described as robustly and efficiently as possible. For this purpose, the M-Surf method [45] is used, which is adapted to the NSS. In the first step, an area of size $20\sigma_{n_{\text{NSS}}} \times 20\sigma_{n_{\text{NSS}}}$ is defined around the extracted feature. To make the subsequent feature description rotation invariant, this surface is rotated around the dominant orientation $\Phi_{k_{\text{kf}}}$ of the feature under investigation as shown in Fig. 9b.

The $20\sigma_{n_{\text{NSS}}} \times 20\sigma_{n_{\text{NSS}}}$ large area is subsequently divided into 16 identically sized squares. For all 16 squares, corresponding to a sliding window of size $9\sigma_{n_{\text{NSS}}} \times 9\sigma_{n_{\text{NSS}}}$, the derivatives in the x and the y directions are determined and weighted with a Gaussian distribution with a standard deviation of $\sigma = 2.5\sigma_{n_{\text{NSS}}}$ with respect to the feature position. The derivatives are subsequently assigned to a descriptor vector $\boldsymbol{\psi}_i$.

$$\boldsymbol{\psi}_i = \begin{bmatrix} \sum \frac{\partial}{\partial x} \\ \sum \frac{\partial}{\partial y} \\ \sum \left|\frac{\partial}{\partial x}\right| \\ \sum \left|\frac{\partial}{\partial y}\right| \end{bmatrix} \tag{43}$$

This is performed for each of the 16 sub-regions and combined into a common vector $\boldsymbol{\Psi}$, with each sub-descriptor normalized by a Gaussian distribution with standard deviation $\sigma = 1.5\sigma_{n_{\mathrm{NSS}}}$.

$$\boldsymbol{\Psi}_{k_{\mathrm{kf}}} = \begin{bmatrix} \boldsymbol{\psi}_1 \\ \vdots \\ \boldsymbol{\psi}_{16} \end{bmatrix} \tag{44}$$

Now a specific and rotation invariant descriptor has been determined for each feature. Therefore, it is possible to determine correspondences between the features of the reference image and features of the object image. For this purpose, the squared difference of the elements of the two descriptor vectors to be compared is summed up. As soon as the squared error is below a chosen threshold, the features are marked as correspondence. To avoid multiple correspondences between different features, the correspondence with the smallest error is selected in a second iteration, which enables a robust transformation.

3) Pose Estimation: In the final step of image registration, a transformation is estimated from the detected features and determined correspondences. First, translation and rotation are determined via a procedure that minimizes the squared error between the found feature pairs from sub-Section V-B2 [52]. The corresponding features of the object image are represented below as a two-dimensional point set $\{\mathbf{p}\}$ with $p_i = [x_i, y_i]$ whereas the features of the reference image are represented as a two-dimensional point set $\{\mathbf{p}'\}$ with $p_i' = [x_i', y_i']$. The corresponding feature positions are represented by the transformation

$$\mathbf{p}_i' = \mathbf{R}_{\mathrm{sm}} \cdot \mathbf{p}_i + \mathbf{T}_{\mathrm{sm}}, \tag{45}$$

where $\mathbf{R}_{\mathrm{sm}}$ describe a rotation matrix with dimensions 2×2 and $\mathbf{T}_{\mathrm{sm}}$ a 2×1 translation vector. The most probable matrices $\mathbf{R}_{\mathrm{sm}}$ and $\mathbf{T}_{\mathrm{sm}}$ are determined via a quadratic error minimization of all features found.

$$\left[\tilde{\mathbf{R}}_{\mathrm{sm}}, \tilde{\mathbf{T}}_{\mathrm{sm}}\right] = \underset{\mathbf{R}_{\mathrm{sm}},\mathbf{T}_{\mathrm{sm}}}{\arg\min} \left(\sum_{i=1}^{K_{\mathrm{kf}}} \left(\mathbf{p}_i' - \left(\mathbf{R}_{\mathrm{sm}} \cdot \mathbf{p}_i + \mathbf{T}_{\mathrm{sm}}\right)\right)^2 \right) \tag{46}$$

This equation can be solved very efficiently via a singular value decomposition according to [52]. Based on the rotation matrix, the relative vehicle pose $\mathbf{P}_{\mathrm{v}}^{\mathrm{g}} = \begin{bmatrix} P_{\mathrm{v,x}} & P_{\mathrm{v,y}} & P_{\mathrm{v,\Phi}} \end{bmatrix}$ is determined.

C. Graph-Based SLAM

With both dead reckoning and scan matching, the trajectory of the vehicle can be determined solely on the basis of the radar data. However, both methods lead to erroneous pose estimation during longer measurement runs, which significantly degrades the quality of the resulting map as illustrated in Section VI. However, since there is no existing map in which the vehicle can perform a self-localization, a SLAM approach is chosen to determine the most probable pose of the vehicle and the most probable map. For this purpose, a graph-based SLAM is used in the following because it has significant robustness and speed advantages compared to FastSLAM or Kalman-SLAM algorithms [53].

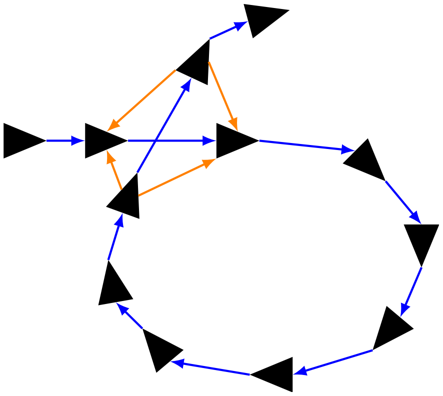


Fig. 11. Sketch of a graph with multiple nodes, consecutive constraints (blue) and constraints used for loop closures (orange).

In this case, the nodes of the graph represent the vehicle poses at different times, with the edges between the nodes representing the constraints between the poses. The nodes are shown as a triangle and the constraints as arrows in Fig. 11. These constraints can be obtained via different ways. First, constraints exist between two consecutive nodes based on dead reckoning from Section V part V-A and based on scan matching from Section V part V-B of consecutive measurements. These types of constraints are shown as blue arrows in Fig. 11. Furthermore, however, the features determined in Section V-B can also be used to establish constraints between non-consecutive nodes, where these are shown as orange arrows in Fig. 11.

Once the graph is created, which will be referred to as the front-end, it is solved using an optimization procedure, the back-end, to find a solution that minimizes the error between all constraints.

1) Front-End: The set of nodes N_{n} and the corresponding nodes $q_{n_{\mathrm{n}}}$ are combined in a vector $\mathbf{q} = [q_1, \ldots, q_{N_{\mathrm{n}}}]$. The nodes $q_{n_{\mathrm{n}}}$ and $q_{n_{\mathrm{n}}'}$ are connected by the edge $c_{n_{\mathrm{n}},n_{\mathrm{n}}'}$.

$$c_{n_{\mathrm{n}},n_{\mathrm{n}}'} = q_{n_{\mathrm{n}}} \rightarrow q_{n_{\mathrm{n}}'} \tag{47}$$

The initial constraints $c_{n_{\mathrm{n}},n_{\mathrm{n}}+1}$ are taken from the dead reckoning and scan matching and are shown as blue arrows in Fig. 11. Furthermore, general constraints are also entered into the graph. These constraints describe the recognition of already visited locations and are called loop closures. For this purpose, the described scan matching from Section V-B is not only performed for consecutive frames but for all recorded frames. According to the estimation uncertainty, these constraints are subsequently also added to the graph, which is shown as orange arrows in Fig. 11. This constructed graph is optimized and solved with the back-end in the following.

2) Back-End: The optimization of the constructed graph is done by minimizing the squared error of all constraints. The error is defined as the difference between the measurement

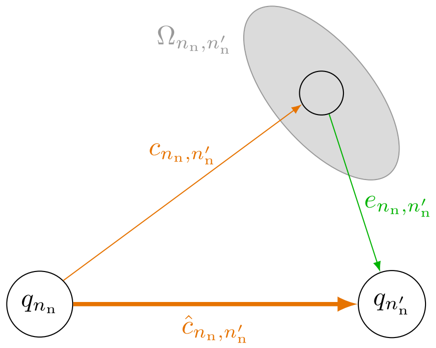

$\Omega_{n_\mathrm{n},n'_\mathrm{n}}$
$c_{n_\mathrm{n},n'_\mathrm{n}}$
$e_{n_\mathrm{n},n'_\mathrm{n}}$
q_{n_n}
$q_{n'_\mathrm{n}}$
$\hat{c}_{n_\mathrm{n},n'_\mathrm{n}}$

Fig. 12. Illustration of two nodes q_{n_n} and $q_{n'_\mathrm{n}}$ and the corresponding estimated constraint $c_{n_\mathrm{n},n'_\mathrm{n}}$ and measured constraint $\hat{c}_{n_\mathrm{n},n'_\mathrm{n}}$.

$c_{n_\mathrm{n},n'_\mathrm{n}}$ and the expected measurement $\hat{c}_{n_\mathrm{n},n'_\mathrm{n}}$. This expected measurement can be determined from the current configuration of the graph and the nodes. Thus, the error between the expected and real measurement can be calculated with

$$e_{n_\mathrm{n},n'_\mathrm{n}} = c_{n_\mathrm{n},n'_\mathrm{n}} - \hat{c}_{n_\mathrm{n},n'_\mathrm{n}} \tag{48}$$

This error and the constraints are shown in Fig. 12.

As soon as the estimate $c_{n_\mathrm{n},n'_\mathrm{n}}$ has an unequally distributed uncertainty, (48) has to be adapted

$$e_{n_\mathrm{n},n'_\mathrm{n},\Omega} = e^\mathrm{T}_{n_\mathrm{n},n'_\mathrm{n}} \boldsymbol{\Omega}_{n_\mathrm{n},n'_\mathrm{n}} e_{n_\mathrm{n},n'_\mathrm{n}} \tag{49}$$

The variable $\boldsymbol{\Omega}_{n_\mathrm{n},n'_\mathrm{n}}$ characterizes the uncertainty. Due to the high robustness and consistency of the ego-motion estimation, the uncertainty for the ego-motion-based dead reckoning is assumed to be constant. Based on the standard deviations for the velocity estimate $[\sigma_{v_x}, \sigma_{v_y}, \sigma_\omega] = [0.045\,\frac{\mathrm{m}}{\mathrm{s}}, 0.045\,\frac{\mathrm{m}}{\mathrm{s}}, 0.56\,\frac{^\circ}{\mathrm{s}}]$, the local pose uncertainties (x, y, Φ) according to the measurement data from chapter VI for dead reckoning (31) and one frame are

$$\boldsymbol{\Omega}_\mathrm{DR} = \begin{pmatrix} 0.0012 & 0 & 0 \\ 0 & 0.0008 & 0005 \\ 0 & 0.0005 & 0.0003 \end{pmatrix}. \tag{50}$$

According to [36], this results in a significantly smaller measurement uncertainty than estimating the velocity with budget IMUs or GNSS. The uncertainty for the scan matching is calculated from the uncertainty of the determined kaze features. For this, the error corresponding to (46) for the estimate $\left[\tilde{\mathbf{R}}_\mathrm{sm}, \tilde{\mathbf{T}}_\mathrm{sm}\right]$ is calculated with

$$\boldsymbol{E}_{\mathrm{SM},i,x/y} = \mathbf{p}'_i - \left(\mathbf{R}_\mathrm{sm} \cdot \mathbf{p}_i + \mathbf{T}_\mathrm{sm}\right) \tag{51}$$

for each of the K_kf kaze features ($i \leq K_\mathrm{kf}$). While the position error in x- and y-direction follows directly, the rotation error has to be determined via a transformation in polar coordinates in relation to the vehicle coordinate system:

$$\boldsymbol{E}_{\mathrm{SM},i,\Phi} = \measuredangle \mathbf{p}'_i - \measuredangle\left(\mathbf{R}_\mathrm{sm} \cdot \mathbf{p}_i + \mathbf{T}_\mathrm{sm}\right) \tag{52}$$

The covariance is obtained by

$$\boldsymbol{\Omega}_\mathrm{SM} = \mathrm{Cov}[\boldsymbol{E}_\mathrm{SM}] \tag{53}$$

$$\boldsymbol{E}_\mathrm{SM} = \begin{pmatrix} \boldsymbol{E}_{\mathrm{SM},1,x} & \boldsymbol{E}_{\mathrm{SM},1,y} & \boldsymbol{E}_{\mathrm{SM},1,\Phi} \\ \vdots & \vdots & \vdots \\ \boldsymbol{E}_{\mathrm{SM},K_\mathrm{kf},x} & \boldsymbol{E}_{\mathrm{SM},K_\mathrm{kf},y} & \boldsymbol{E}_{\mathrm{SM},K_\mathrm{kf},\Phi} \end{pmatrix} \tag{54}$$

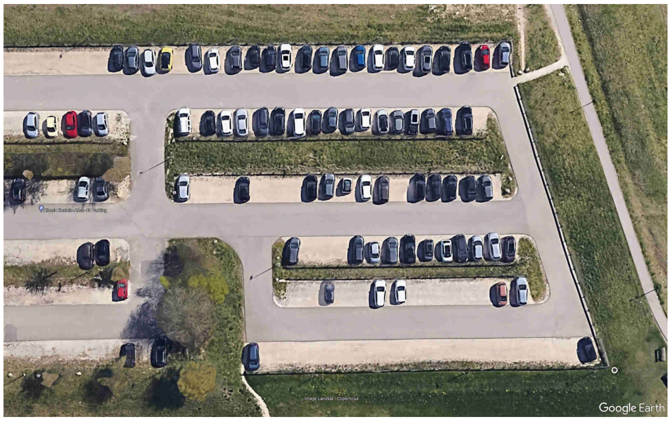


Fig. 13. Bird's eye view of the measurement environment with Google Earth ©2023, Kartendaten ©2023 GeoBasis-DE/BKG (©2009). Satellite image does not reflect real vehicle positions due to different recording times.

The error function of the entire graph is thus computed using the sum of all errors as a function of the nodes $\mathbf{q}$.

$$E_\Omega(\mathbf{q}) = \sum_{n_\mathrm{n},n'_\mathrm{n}} e_{i,j,\Omega} \tag{55}$$

The optimal and most probable node values and thus vehicle poses are obtained by minimizing the entire graph by

$$\tilde{\mathbf{q}} = \arg\min_{\mathbf{q}} \left(E_\Omega(\mathbf{q})\right). \tag{56}$$

This problem can be solved via common numerical methods such as the Gauss-Newton method, or a modification of it, the Levenberg-Marquardt method. In this work, the framework g^{2}o corresponding to [54] is applied, which is based on the Levenberg-Marquardt method. In this case, the minimum of the error function is approximated iteratively via linearizations.

VI. Measurements

The measurements to verify the theoretical derivations from the previous sections were performed in a parking lot according to Fig. 13. Driving on the parking lot allows a curved trajectory with a large loop (upper loop), which has a length of approximately 180 m, and a small loop (lower loop), which has a length of approximately 140 m.

The measurement setup consists of $N_\mathrm{r} = 6$ incoherently connected chirp-sequence radar sensors, which are exploited for the different signal distribution steps [55]. The sensors are positioned around the vehicle at equal angular intervals of 45°, ensuring a complete 360°field of view, which is depicted in Fig. 14. In general, all six sensors are evaluated for ego-motion estimation and self-localization and only the two sensors S_2 and S_5 are employed for SAR processing.

The precise locations of the sensors relative to the vehicle's coordinate system were determined using a Trimble tachymeter. The orientation of the sensors was estimated by use of self-calibration algorithms [25], [56]. To achieve time synchronization among the radar sensors, an external trigger is employed. By introducing a slight variation in the starting frequencies of the sensors, this synchronization enables interference suppression between them [57]. Since all the

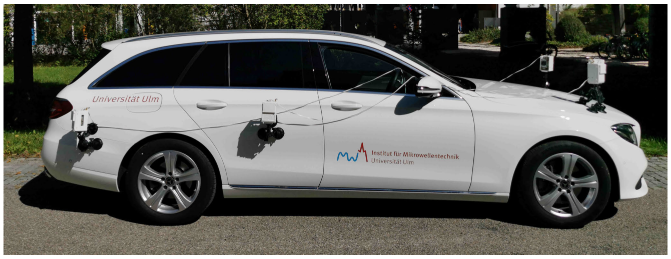


Fig. 14. Experimental system setup with six chirp-sequence radar sensors mounted on a car according to the sketch from Fig. 2.

TABLE I
USED RADAR PARAMETERS FOR SLAM-BASED SAR PROCESSING

Parameter	Value		
	S_1	S_2	S_3
Start frequency (GHz)	77.22	77.25	77.28
Approx. Orientation $\varphi^{c}_{n_r}$ (°)	−135	−90	−45
	S_5	S_6	S_7
Start frequency (GHz)	77.37	77.40	77.43
Approx. Orientation $\varphi^{c}_{n_r}$ (°)	45	90	135
Bandwidth	3.07 GHz		
Ramp repetition time	67 µs		
Sampling frequency (IQ)	10 MHz		
Number of ramps (per Tx)	128		
Max. unambiguous velocity	4.823 $\frac{\mathrm{m}}{\mathrm{s}}$		
Max. range	25 m		
Number of transmitters	1		
Number of receivers	4		
Measurement rate	37 Hz		
Average vehicle speed	5 $\frac{\mathrm{m}}{\mathrm{s}}$		

sensors share identical radar parameters except for their start frequencies, the transmit ramps of each sensor consistently exhibit a frequency offset from one another. As a result, there is no overlap or interference in the IF band. The sensors are only synchronized in time to have accurate time stamps and to avoid interference. The sensors are incoherently networked, so no common oscillator reference is distributed and each sensor operates on its own oscillator. The radar parameters are listed in Table I.

The ground truth trajectory of the vehicle was determined using a Trimble total station with an accuracy of at least 2 mm to ensure an accurate ground truth trajectory and is shown in red in Fig. 15.

Based on this precise ground-truth trajectory, a SAR map can be processed according to Section IV part IV-A, which is shown in Fig. 16 based on both side-looking radar sensors S_2 and S_5.

The Fig. 16 shows not only highly reflective objects such as vehicles or posts and thus free or occupied parking spaces, but also weakly reflective objects such as a wooden fence (distinctive targets are marked in Fig. 23a). It is clearly visible that this fence is also detectable in the area behind the vehicles, since the synthetic aperture processes a reflection under many different angles and thus a high integration gain exists. This high integration gain also leads to the fact that the asphalted road can be clearly distinguished from the graveled parking spaces. Thus, with the help of SAR imaging, not only a detection of free parking spaces is possible, but also an identification of parking areas.

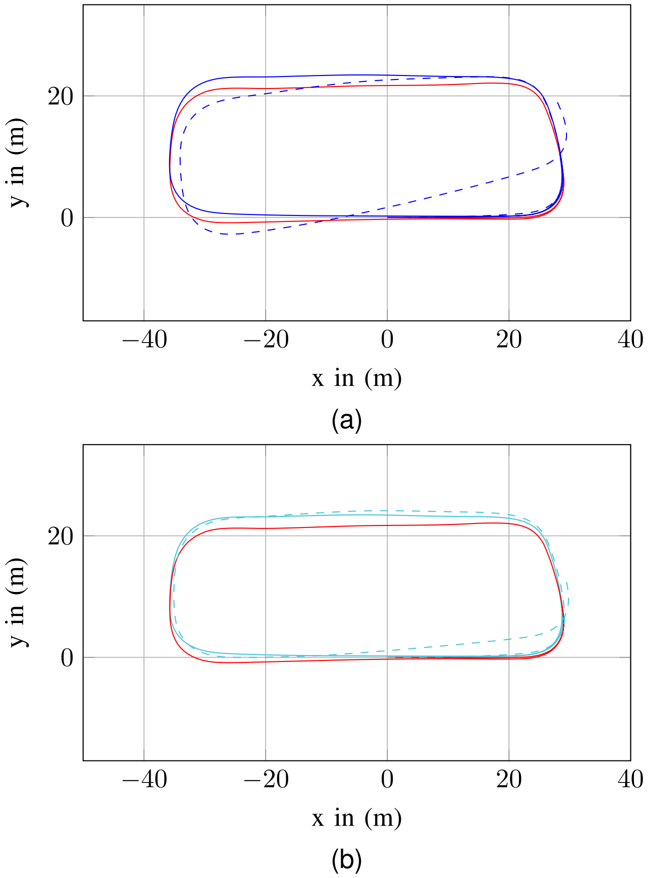


Fig. 15. Trajectory estimation based on dead reckoning (a), (—, - - -) and based on scan matching (b), (—, - - -) for a correctly chosen threshold T_v (solid) and an incorrectly chosen threshold T_v (dashed). The ground-truth trajectory is shown in red (—).

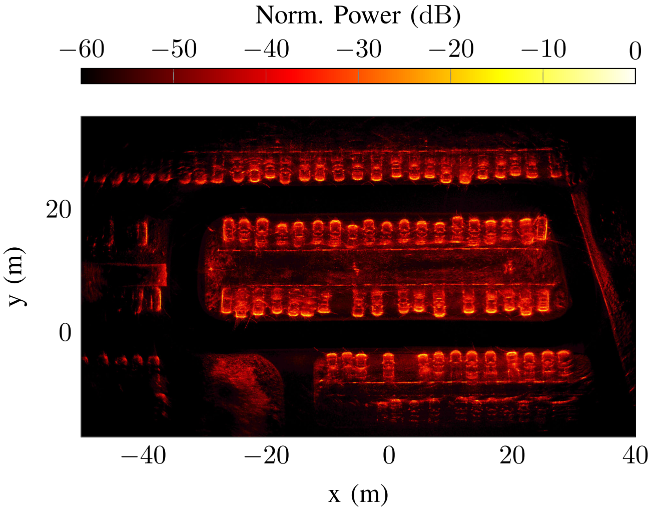


Fig. 16. SAR-based representation of the environment on the basis of the ground truth trajectory.

A. Dead-Reckoning Versus Scan-Matching

As soon as a ego-motion estimation with infinite quality or a scan matching with infinite accuracy would be possible, an identical SAR mapping of the environment could be generated. However, this is not a given due to measurement

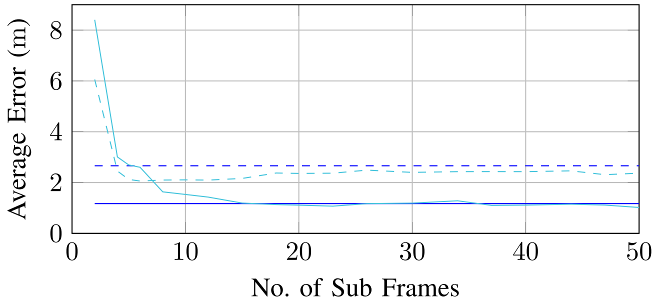


Fig. 17. Illustration of the influence of the number of sub frames on ego-motion based dead reckoning (——, - - -) and scan matching based trajectory estimation (——, - - -).

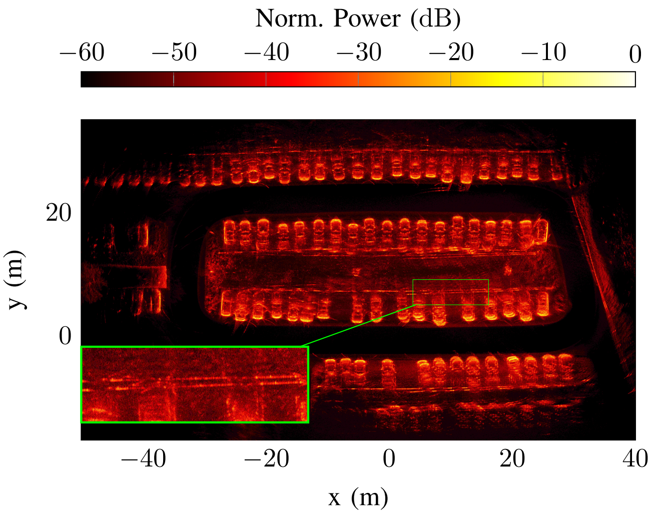


Fig. 18. SAR-based representation of the environment on the basis of dead-reckoning. Enlargement of the double structure of the wooden fence.

inaccuracies and deterministic errors. This becomes clear once the estimated trajectories based on these two localization methods are compared, which is shown in Fig. 15a for dead reckoning and Fig. 15b for scan matching.

The red line (——) represents the ground truth trajectory whereas the blue lines (——, - - -) describe the ego-motion based dead reckoning and the turquoise lines the scan matching (——, - - -). The solid lines represent the trajectories based on a correctly parameterized ego-motion algorithm whereas the dashed lines represent the result based on an incorrectly configured ego-motion algorithm. An incorrect parameterization of the ego-motion algorithm from Section V part V-A is present as soon as the threshold is set to low with $T_{\mathrm{v}} = 0.1\,\frac{\mathrm{m}}{\mathrm{s}}$, whereas a correct parameterization is given for a threshold of $T_{\mathrm{v}} = 0.7\,\frac{\mathrm{m}}{\mathrm{s}}$ resulting in all relevant targets being considered for ego-motion. By choosing the correct threshold, the average trajectory error over all frames is reduced from 2.66 m to 1.02 m.

A similar behavior can be observed for the scan matching algorithm in Fig. 15b. Although the algorithm in general does not depend on the ego-motion estimation, the scan matching becomes significantly worse as soon as an incorrect threshold T_{v} is chosen for the ego-motion estimation, which is shown as (- - -). The reason for this is mainly due to the fact that as described in Section V part V-B several frames are combined to get a higher target density to better detect the kaze features. The position of the vehicle for all sub-frames to be combined is determined based on ego-motion estimation and thus dead reckoning. The number of sub-frames to be evaluated has a significant influence on the quality of the scan matching which is shown in Fig. 17.

The coloring of the Fig. 17 is identical to that of Fig. 15. It is obvious that the number of sub frames is irrelevant for the ego-motion estimation and the subsequent dead reckoning, which is shown in blue (——, - - -). Pose estimation based on scan matching, on the other hand, depends significantly on the number of sub frames evaluated. As soon as only one frame is used, in general too few targets are detected, which means that no prominent structures can be detected in the grid maps and thus the average position error over the whole trajectory is between 6 m and 8 m. On the other hand, it can be clearly seen that the more sub frames based on ego-motion are linked together, the average error is significantly minimized and ultimately the accuracy of ego-motion based dead reckoning is improved.

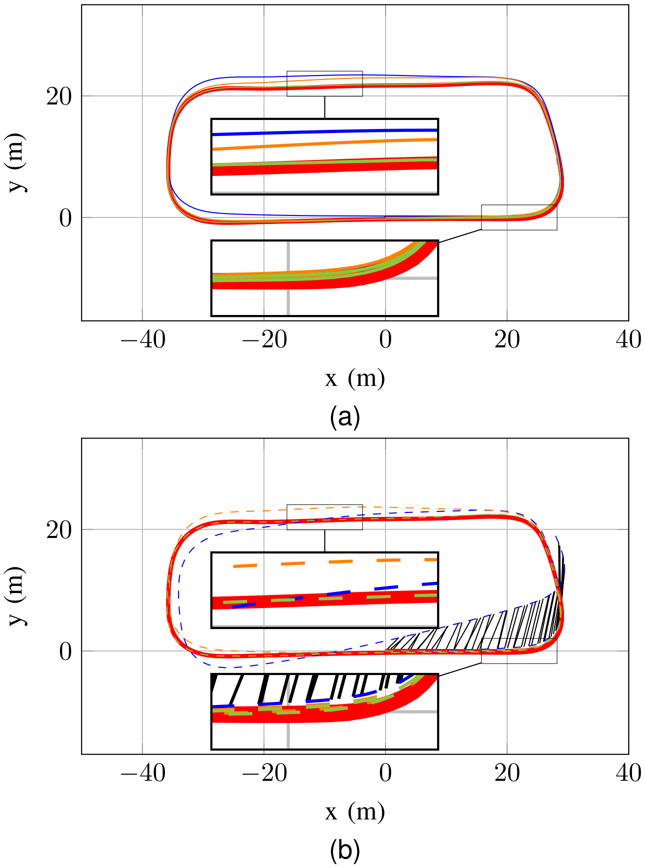
The trade-off between best scan matching and independent estimation versus ego-motion based dead reckoning is about 8 evaluated sub frames, which are also used for scan matching in the following. In the end, however, it can be seen that with a correctly parameterized ego-motion estimation as well as with a correctly parameterized scan matching no ideal trajectory estimation is guaranteed. This can also be seen when a SAR map is created based on dead reckoning with correctly chosen threshold as shown in Fig. 18.

It is noticeable, that a reasonable picture of the surroundings can already be generated at first glance. Free parking spaces can already be identified. On closer inspection, however, it becomes apparent that the image of the environment is distorted in some places and that double structures exist even though they do not occur in reality. This applies to the wooden fences which are no longer horizontal but convex and concave as well as the vehicle contours which are heavily smeared. This is the result of the integration error of the dead reckoning, whereby each speed error leads to a larger and larger trajectory error. This error is minimized in the following by using the SLAM algorithm presented in Section V part V-C.

B. Graph-Based SLAM

The trajectory estimated via ego-motion estimation and dead reckoning is used as the base trajectory. From this, the first configuration of nodes and the edges connecting consecutive nodes are generated. The sub-maps generated for each node are compared for loop closure detection. If a transformation is estimated, a constraint is created between the two nodes. The result after optimizing the graph (Section V part V-C) is shown in Fig. 19.

Fig. 19a illustrates the results with correctly selected RANSAC threshold T_{v} (solid lines) while Fig. 19b illustrates the results of an incorrectly selected threshold (dashed lines). The color coding corresponds to the previous color coding.



(a)

(b)

Fig. 19. Trajectory estimation based on (a): correctly choosen velocity threshold T_v, and (b): incorrectly choosen velocity threshold T_v. Ground truth trajectory (——), dead reckoning (——, - - -), graph-based SLAM (——, - - -), and optimized graph-based SLAM (——, - - -).

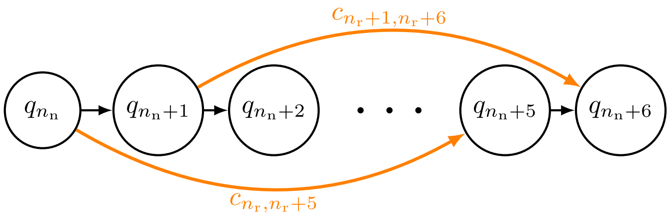


Fig. 20. Illustration of stepped scan matching for well-defined constraints without loop closures.

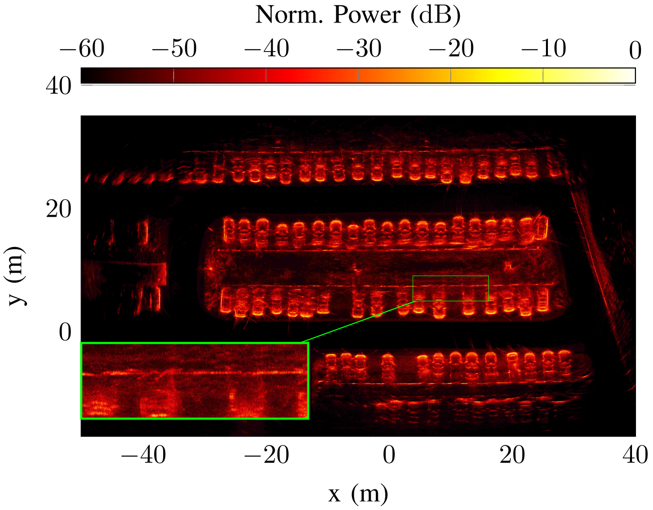


Fig. 21. SAR-based representation of the environment on the basis of the graph SLAM-based trajectory.

Red corresponds to ground truth trajectory, blue corresponds to ego-motion based dead reckoning, and orange corresponds to graph-based SLAM pose estimation. The black lines denote the constraints based on the loop closures.

It can be clearly seen that the graph-based SLAM again significantly improves the trajectory estimation. With a well parameterized ego-motion estimation, shown in Fig. 19a, the average error can be reduced from 1.02 m to 0.59 m. The same phenomenon is evident for the evaluation with a badly parameterized ego-motion, which is shown in Fig. 19b. In this case, the average error is reduced from 2.66 m to 0.83 m.

Ultimately, however, it is noticeable in both estimates that the estimate in the range $y = 21\,\mathrm{m}$ has errors of up to one meter, although the graph-based SLAM has already been applied (orange trajectory). Since this area was not traversed several times, no loop closures can be found in this area. Thus, there are no additional constraints over which the graph could be optimized. To solve this problem the scan matching from Section V part V-B is not only applied to consecutive frames but also to frames which have a larger time distance than $t = 1$ from each other. This results in redundancy, which can be added to the graph as constraints between the nodes to again significantly improve the estimation. This principle is illustrated in Fig. 20.

Fig. 20 illustrates that each node q_{n_n} has an additional exemplary constraint to the node $q_{n_\mathrm{n}+5}$. The distance should be chosen in such a way that objects from different perspectives are detected, but remain small enough to still be able to detect identical objects. The result of this optimized graph is also shown in 19a and 19b as green trajectory. It is clearly visible the estimated trajectory and the ground truth trajectory are almost perfectly superimposed and thus an optimized estimation is possible. The average error is again reduced from 0.59 m to 0.28 m for the well parameterized ego-motion and from 0.83 m to 0.33 m for the badly parameterized ego-motion. This illustrates not only a significantly increased precision of the localization algorithm but also a significantly increased robustness. Regardless of whether the ego-motion was parameterized well or badly, the result of the final pose estimation is only 17 % worse compared to the initial pose estimation error of more than 250 %.

The resulting map is shown in Fig. 21. Compared to the ground-truth based SAR map in Fig. 16, no obvious differences can be seen here. This is quantitatively illustrated by the three lanterns in the center of the image at $y = 20\,\mathrm{m}$ by comparing the position of these lanterns with the positions from the ground-truth image corresponding to Fig.16. The average position deviation is only 7 cm, making the map a true representation of the environment.

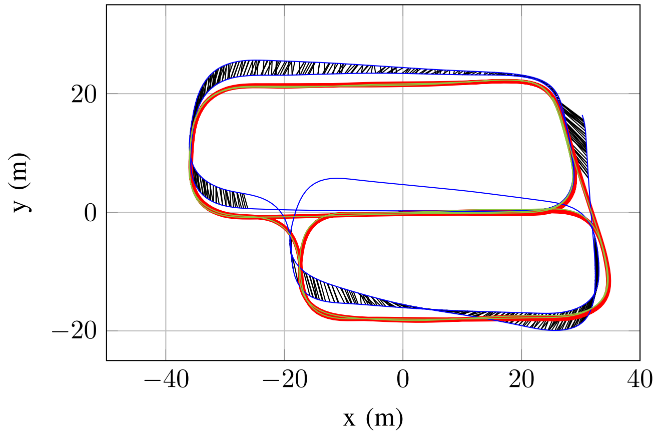


Fig. 22. Trajectory estimation based on dead reckoning (——), and optimized graph-based SLAM (——) in comparison to ground truth trajectory (——).

C. Evaluation of Longer Trajectories

The previous investigations were all performed only for a measurement trajectory with a constant measurement duration of 30 s. However, the described SLAM algorithm for the generation of high-resolution environment images is also applicable for significantly longer trajectories, which is illustrated in the following Fig. 22. The measurement duration is approximately 140 s with just under 500 m traveled during this time. Following the previous convention, the ground-truth trajectory is shown in red, the ego-motion based trajectory in blue, and the SLAM based trajectory in green. The average absolute localization error over all 4800 evaluated frames is 42 cm. This error is only 50 % larger despite a measurement run more than twice as long as in Section VI part VI-B. A visual comparison between ground-truth based SAR imaging and SLAM based SAR imaging is shown in Fig. 23.

It is evident that qualitatively the images look identical. No double structures or distorted wooden fences can be identified. Furthermore, the gravel of the parking bays is still clearly distinguishable from the road. The reasons for this are the small localization errors within small sub-apertures, affecting the SAR processing the most. For a synthetic aperture of 16 cm, which corresponds to one frame and thus 384 evaluated ramps, the standard deviation is 1.2 mm. For a synthetic aperture size of 96 cm, the standard deviation is 17 mm. This corresponds to a deviation of only four wavelengths for a synthetic aperture of almost one meter. This minimizes the influence of the localization error on the environmental mapping as illustrated in the following.

In a SLAM algorithm, not only is the accuracy of the trajectory estimation crucial important but also the accuracy of the map. For this purpose, the position of a total of nine defined targets (posts) were investigated. The comparison between ground-truth based SAR processing from Fig. 23a and SLAM-based SAR processing Fig. 23b shows that the average localization error is 16 cm, ensuring accurate mapping even over large areas. Another way to compare the quality of the image is to determine the signal-to-clutter ratio (SCR). For this, the SCR is determined based on the wooden fence (signal) and the gravel and thus an empty parking area (clutter) according to Fig. 23a. The SLAM-based SAR image shows a lower SCR by only 0.35 dB, so that weakly reflecting targets can still be clearly distinguished from the gravel, although the position of the vehicle is estimated exclusively on the basis of the radar data.

D. Evaluation of Optimal Number of Sensors

The SLAM measurement results from Section VI are based on all $N_\mathrm{r} = 6$ radar sensors installed. The extent to which the number of sensors is decisive for dead reckoning, scan matching and the subsequent graph-based SLAM is shown in Fig. 24.

The Fig. 24 illustrates the relationship between the number of sensors and the average logarithmic error of the trajectory estimation in meters based on different methods. Analogous to the measurement results, the blue curve indicates trajectory estimation based on dead reckoning, the turquoise curve indicates trajectory estimation based on scan matching, and the green curve illustrates trajectory estimation based on graph-based SLAM. It is clearly visible that on average the accuracy of the trajectory estimation can be significantly increased with increasing number of sensors. This is especially true for the trajectory estimation based on dead reckoning (blue). Already purely based on dead reckoning, with six sensors the error is reduced from almost 8 m to only 1 m.

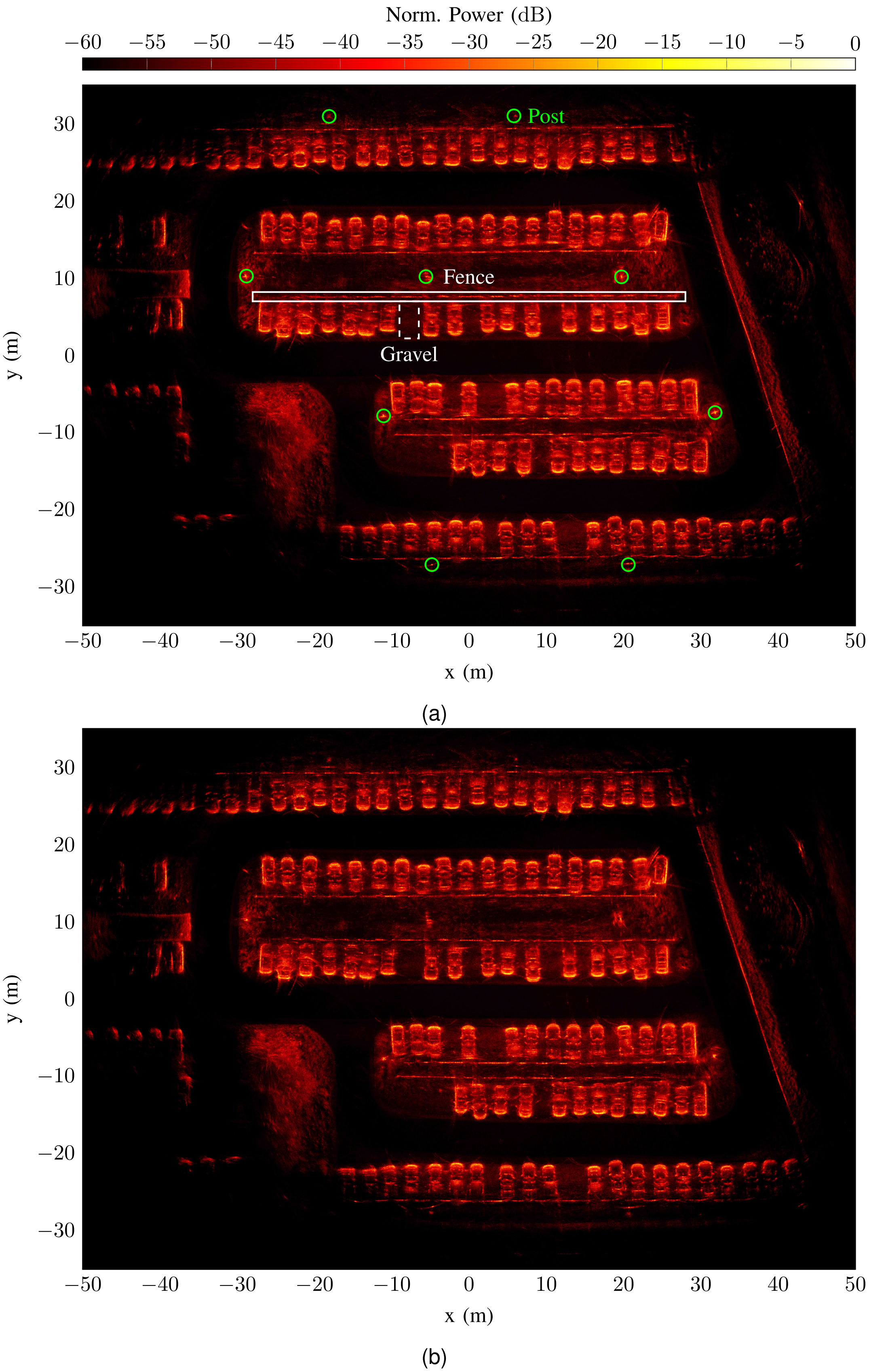
A similar behavior is also evident for scan matching. Whereby the estimation with six sensors does not bring any advantage compared to the estimation with only four sensors. In contrast to ego-motion estimation, where the data is evaluated jointly but each target is processed independently, the detected targets smear in the joint sub-map as shown in Fig. 4. Moreover, since the position and orientation of the sensors in relation to the vehicle coordinate system can only be estimated, the target peaks in the sub-map being evaluated widen when multiple detections are made by different sensors.

The graph-based SLAM finally merges the ego-motion based dead reckoning and the scan-matching based trajectory estimation. This results in the green curve in Fig. 24, which illustrates a significant reduction in the estimation error for up to four sensors. The improvement from four to six sensors is correspondingly small. This is largely due to the scan-matching based loop closures, which according to the green curve do not improve with more sensors, since the targets do not produce significantly defined kaze features when the relative pose estimation of the radar sensors is incorrect.

E. Impact of Localization Inaccuracies on the SAR Image

According to chapter VI-C, the average position error for the long trajectory is 42 cm, which is more than 200 times larger than the $\lambda/2$ requirement for SAR applications. However, according to Fig. 23a it can be seen that this has almost no effect on the imaging quality and defocusing effects. This seeming apparent contradiction can be traced back to the following three factors:

- limited detection range of the radar sensors
- attenuation
- frame-based error



(a)

(b)

Fig. 23. SAR-based representation of the environment on the basis of the ground truth trajectory, depicted in a), and based on the optimized SLAM, depicted in b).

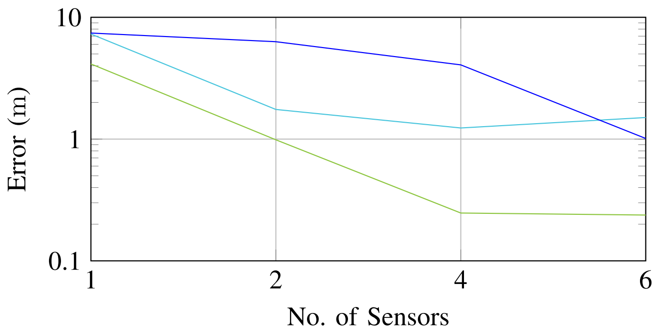


Fig. 24. Influence of the size of the radar sensor network on the accuracy of trajectory estimation for dead-reckoning (——), scan-matching (——), and the graph-based SLAM (——).

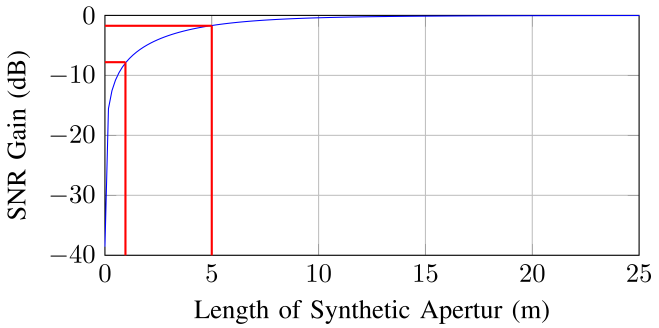


Fig. 25. Influence of different aperture lengths on the SNR of a target at the position $t = (0\,\mathrm{m}|5\,\mathrm{m})$.

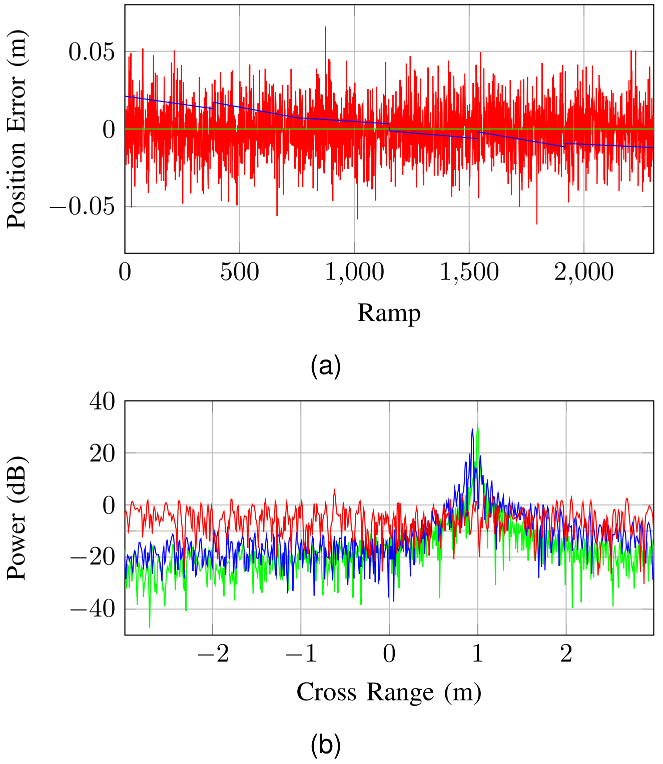


Fig. 26. Illustration of different position errors (a) and their effect on SAR mapping (b). Green - ground truth trajectory, red - random positioning, blue - noise corresponding to self-localization.

1) Limited Detection Range of Radar Sensors: The average position error of 42 cm occurs over the entire trajectory. For the final evaluation, however, the relative error in short sections of the trajectory is relevant. The parking lot, with its length of 100 m and a width of 60 m, is so spatially extended that the radar sensors cannot cover it completely with one measurement. On the one hand because of occlusion effects and on the other hand because of the limited maximum range of the radar sensors of 25 m. The relative error of the vehicle positions of the trajectory that are farthest from each other and thus have a relative error of more than 42 cm to each other ($(-15\,\mathrm{m}|-15\,\mathrm{m})$ and $(30\,\mathrm{m}|20\,\mathrm{m})$) is consequently irrelevant for the SAR processing and only distorts the final image.

2) Attenuation: Another factor is the influence of the trajectory on the SNR of a target. The received power at any synthetic antenna position is significantly dependent on two factors (in the case of an ideal target). First, on the distance due to the r^4 dependence, and second, on the angle of arrival and thus the angle-dependent attenuation due to the antenna pattern [58]. This leads to the fact that very large synthetic apertures bring only a small performance advantage compared to much smaller synthetic apertures, since the additional accumulated energy of large apertures is negligible. This is shown in Fig. 25 for a simulation without position error. For this, a typical automotive scenario with a point target at position $t = (0\,\mathrm{m}|5\,\mathrm{m})$ is studied. In each case, the synthetic aperture is centered to the target. As an example, for an aperture with a length of 1 m this results in a trajectory from $(-0.5\,\mathrm{m}|0\,\mathrm{m}) \rightarrow (0.5\,\mathrm{m}|0\,\mathrm{m})$.

Fig. 25 illustrates the SNR loss of a short synthetic aperture compared to a long synthetic aperture. The loss is shown in relation to the maximum SNR and thus for the maximum aperture length. The SNR of the target at position $t = (0\,\mathrm{m}|5\,\mathrm{m})$ is 7.8 dB lower for an aperture length of 96 cm than for an aperture length of 25 m. For a synthetic aperture length of 5 m, the difference is only 1.7 dB, which is marked in red in Fig. 25. According to the normalized measurement results from Fig. 23 the image of the environment is not SNR limited but limited by ground clutter, which results from the asphalt and gravel. This makes even the SNR loss of 7.8 dB of the trajectory over 96 cm negligible.

3) Frame-Based Error: The 96 cm long trajectory corresponds to exactly 6 evaluated frames in the measurement. According to chapter VI-C, it could be shown from the measurements that the standard deviation of the position estimate for this trajectory length is 17 mm and thus 10 times higher than $\lambda/2$. Nevertheless, according to Fig. 23a, no visual differences occurs. This is because the position error is not completely random, which is shown in Fig. 26a. The position error for 6 frames á 384 ramps is shown for no position noise (green), random position noise (red) and a real position noise from the measurement (blue).

The positional error of the measurement exhibits a distinctly piecewise steady pattern. Each frame comprising 384 ramps relies solely on dead reckoning, causing a continuously evolving positional error attributed to an inaccurate estimation of ego-motion. Following each frame, encompassing 384 ramps, the vehicle's position undergoes estimation through

TABLE II
Monte Carlo Simulation to Evaluate the SAR Performance Based on Trajectories With Different Types of Noise

	No Position Noise	Random Position Noise	Steady Position Noise
Max (dB)	29.9127	14.5261	27.2869
ISLR (dB)	-31.1763	-5.5035	-28.178
$\Delta\measuredangle$ (°)	0	24.52	0.39

scan matching, cross-frame dead reckoning, and graph-based SLAM, resulting in discontinuities in the estimate. The high accuracy of dead reckoning, coupled with cross-frame constraints, yields a trajectory that is piecewise continuous, marked by minor discontinuities between consecutive frames. The SAR processing results of these three trajectories are shown in Fig. 26b in cross range for a range of 5 m. It can be clearly seen that the ground truth trajectory (green) and the estimated trajectory (blue) lead to almost identical results, while the evaluation of the random trajectory position no longer allows SAR processing.

The reasons for accurate SAR processing based on the estimated trajectory is the piecewise steadiness of the position error. A sharp SAR image of the environment can be generated for each of the six sub-apertures even though they have an error of 1.2 mm according to VI-C. Due to this ego-motion based error, the targets have small angular errors of less than one degree [17], [59]. Ultimately, to enable coherent addition of the six sub-apertures it must be ensured that a pixel can be found that compensates for the phase errors due to the positional inaccuracies caused by its erroneous pixel position in the grid. It is apparent from Fig. 26b that such a position can be found adjacent to the green peak. However, since the height of the peak is identical, a coherent addition was possible and thus a precise angle estimate.

To show this in general, the result of a Monte-Carlo simulation with 50000 trials is presented below. For each trial, a target is positioned at a distance of 5 m from the synthetic aperture at a random angle. Subsequently, the ground-truth trajectory, a random noisy trajectory and a piecewise continuous trajectory corresponding to the measurement are simulated according to Fig. 26a with a standard deviation of 17 mm. The performance is evaluated using the following results in Table II.

According to the expectations, it is best if the trajectory would be known exactly and thus no position errors exist. For the simulation an average maximum peak of just under 30 dB and an integrated side lobe ratio (ISLR) of nearly -32 dB results. It should be noted that the ISLR is normalized to the width of the main lobes, which means that the average received power outside the main lobes is on average -32 dB lower. Due to a simulation without position error, there are no erroneous angles under which the targets are detected. In the Monte Carlo simulation with random position noise, however, a peak with 15 dB less power was detected, but on average its localization is wrong by about 25° and therefore random, which makes reliable target localization impossible. Furthermore, it can be seen by the ISLR that the peak found is also minimally above the average noise and sidelobes.

For the position uncertainty based on the presented SLAM algorithm, very similar average values as for simulation without position noise are obtained despite the high uncertainty of almost 2 cm. The maximum peak power is on average only 2 dB lower, the average ISLR is only 5 dB higher which means that unwanted signal components are still clearly below the peak. This also explains the low angular error of the simulated target of less than 0.4°. As soon as the required accuracy is maintained for each frame and the targets are located close to the radar sensor, cross-frame position errors only lead to a minimal degradation of the results, which are not visually identifiable.

VII. Conclusion

The algorithm described in this article enables for the first time SAR processing in the automotive domain of an unknown environment without having to rely on external localization systems. The algorithm enables self-sufficient self-localization based solely on radar sensors. This enables high-resolution mapping that is completely independent of GNSS, IMU or other sensors. It has been shown metrologically that the presented algorithm achieves an average absolute localization accuracy of 42 cm and an average relative localization accuracy within one frame of only 2.5 mm over a driving duration of more than 2 minutes and up to 500 m. Furthermore, it was shown that this application also benefits from cascadable radar sensor networks. A comparison between the ground truth based mapping and the SLAM based mapping reveals that the signal to clutter ratio is only 0.35 dB lower for the slam based mapping, resulting only in a minimal loss of dynamic range. With the help of a Monte-Carlo simulation it could be shown that under certain circumstances the position accuracy may be many times worse than the wavelength without leading to defocused images. The presented algorithm not only enables high-precision environment mapping with 77 GHz chirp-sequence radar sensors, but also precise self-localization in unknown environments based solely on these sensors, allowing flexible use in arbitrary automotvie applications.

References

[1] M. Mielle, M. Magnusson, and A. J. Lilienthal, “A comparative analysis of radar and LiDAR sensing for localization and mapping,” in *Proc. Eur. Conf. Mobile Robots (ECMR)*, Sep. 2019, pp. 1–6.

[2] F. Roos, J. Bechter, C. Knill, B. Schweizer, and C. Waldschmidt, “Radar sensors for autonomous driving: Modulation schemes and interference mitigation,” *IEEE Microw. Mag.*, vol. 20, no. 9, pp. 58–72, Sep. 2019.

[3] M. Gottinger et al., “Coherent automotive radar networks: The next generation of radar-based imaging and mapping,” *IEEE J. Microw.*, vol. 1, no. 1, pp. 149–163, Jan. 2021.

[4] A. Frischen, J. Hasch, and C. Waldschmidt, “Contour recognition with a cooperative distributed radar sensor network,” in *IEEE MTT-S Int. Microw. Symp. Dig.*, Apr. 2015, pp. 1–4.

[5] C. Waldschmidt, J. Hasch, and W. Menzel, “Automotive radar—From first efforts to future systems,” *IEEE J. Microw.*, vol. 1, no. 1, pp. 135–148, Jan. 2021.

[6] S. Kong, F. Shi, C. Wang, and C. Xu, “Point cloud generation from multiple angles of voxel grids,” *IEEE Access*, vol. 7, pp. 160436–160448, 2019.

[7] M. Manzoni, S. Tebaldini, A. V. Monti-Guarnieri, C. M. Prati, and I. Russo, “A comparison of processing schemes for automotive MIMO SAR imaging,” *Remote Sens.*, vol. 14, no. 19, p. 4696, Sep. 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/19/4696

[8] M. Steiner, T. Grebner, and C. Waldschmidt, "Millimeter-wave SAR-imaging with radar networks based on radar self-localization," *IEEE Trans. Microw. Theory Techn.*, vol. 68, no. 11, pp. 4652–4661, Nov. 2020.

[9] A. Laribi, M. Hahn, J. Dickmann, and C. Waldschmidt, "Performance investigation of automotive SAR imaging," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Apr. 2018, pp. 1–4.

[10] T. Grebner, P. Schoeder, V. Janoudi, and C. Waldschmidt, "Radar-based mapping of the environment: Occupancy grid-map versus SAR," *IEEE Microw. Wireless Compon. Lett.*, vol. 32, no. 3, pp. 253–256, Mar. 2022.

[11] T. Grebner, D. Schwarz, and C. Waldschmidt, "On sampling and bandwidth influences of distributed radar sensors on automotive SAR applications," in *Proc. 20th Eur. Radar Conf. (EuRAD)*, Sep. 2023, pp. 181–184.

[12] S. J. LeVoir, P. A. Farley, T. Sun, and C. Xu, "High-accuracy adaptive low-cost location sensing subsystems for autonomous rover in precision agriculture," *IEEE Open J. Ind. Appl.*, vol. 1, pp. 74–94, 2020.

[13] H. Iqbal, M. B. Sajjad, M. Mueller, and C. Waldschmidt, "SAR imaging in an automotive scenario," in *Proc. IEEE 15th Medit. Microw. Symp. (MMS)*, Nov. 2015, pp. 1–4.

[14] C. Kuai, B. Zhao, C. Si, and L. Huang, "Velocity estimation for vehicle-mounted SAR based on deep-learning framework," *IEEE Sensors J.*, vol. 22, no. 23, pp. 22952–22962, Dec. 2022.

[15] T. Grebner, L. L. T. Torres, V. Janoudi, P. Schoeder, and C. Waldschmidt, "Detection of optical road markings with automotive radar sensors," *IEEE Sensors Lett.*, vol. 7, no. 8, pp. 1–4, Aug. 2023.

[16] D. Tagliaferri et al., "Navigation-aided automotive SAR for high-resolution imaging of driving environments," *IEEE Access*, vol. 9, pp. 35599–35615, 2021.

[17] M. Manzoni et al., "Motion estimation and compensation in automotive MIMO SAR," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 1756–1772, Feb. 2023.

[18] T. Grebner, A. Grathwohl, P. Schoeder, V. Janoudi, and C. Waldschmidt, "Probabilistic SAR processing for high-resolution mapping using millimeter-wave radar sensors," *IEEE Trans. Aerosp. Electron. Syst.*, pp. 1–16, 2023.

[19] T. Gisder, M.-M. Meinecke, and E. Biebl, "Synthetic aperture radar towards automotive applications," in *Proc. 20th Int. Radar Symp. (IRS)*, Jun. 2019, pp. 1–10. [Online]. Available: https://api.semanticscholar.org/CorpusID:198145884

[20] H. Iqbal, A. Löffler, M. N. Mejdoub, and F. Gruson, "Realistic SAR implementation for automotive applications," in *Proc. 17th Eur. Radar Conf. (EuRAD)*, Jan. 2021, pp. 306–309.

[21] A. G. Stove, "Linear FMCW radar techniques," *IEE Proc. F, Radar Signal Process.*, vol. 139, no. 5, p. 343, 1992.

[22] H. Rohling and M.-M. Meinecke, "Waveform design principles for automotive radar systems," in *Proc. CIE Int. Conf. Radar*, 2001, pp. 1–4.

[23] A. Meta, P. Hoogeboom, and L. P. Ligthart, "Signal processing for FMCW SAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 11, pp. 3519–3532, Nov. 2007.

[24] A. Meta, P. Hoogeboom, and L. P. Ligthart, "Correction of the effects induced by the continuous motion in airborne FMCW SAR," in *Proc. IEEE Conf. Radar*, 2006, pp. 1–8.

[25] T. Grebner, V. Janoudi, P. Schoeder, and C. Waldschmidt, "Self-calibration of a network of radar sensors for autonomous robots," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 5, pp. 6771–6781, 2023.

[26] V. T. Vu, T. K. Sjogren, and M. I. Pettersson, "A comparison between fast factorized backprojection and frequency-domain algorithms in UWB lowfrequency SAR," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2008, p. 1284.

[27] G. Wang, F. Qi, Z. Liu, C. Liu, C. Xing, and W. Ning, "Comparison between back projection algorithm and range migration algorithm in terahertz imaging," *IEEE Access*, vol. 8, pp. 18772–18777, 2020.

[28] G.-C. Sun, Y. Liu, J. Xiang, W. Liu, M. Xing, and J. Chen, "Spaceborne synthetic aperture radar imaging algorithms: An overview," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 1, pp. 161–184, Mar. 2022.

[29] T. Lensu, "Synthetic aperture radar—Systems and signal processing," *Signal Process.*, vol. 29, no. 1, p. 107, Oct. 1992.

[30] O. Frey, C. Magnard, M. Ruegg, and E. Meier, "Focusing of airborne synthetic aperture radar data from highly nonlinear flight tracks," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 6, pp. 1844–1858, Jun. 2009.

[31] L. M. H. Ulander, H. Hellsten, and G. Stenstrom, "Synthetic-aperture radar processing using fast factorized back-projection," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 3, pp. 760–776, Jul. 2003.

[32] P. Hügler, T. Grebner, C. Knill, and C. Waldschmidt, "UAV-borne 2-D and 3-D radar-based grid mapping," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[33] J. Degerman, T. Pernstål, and K. Alenljung, "3D occupancy grid mapping using statistical radar models," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 902–908.

[34] S. Thrun, "Learning occupancy grids with forward models," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., Expanding Societal Role Robot. Next Millennium*, Feb. 2001, pp. 1676–1681.

[35] S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artif. Intell.*, vol. 99, no. 1, pp. 21–71, Feb. 1998.

[36] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using Doppler radar," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2013, pp. 869–874.

[37] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using multiple Doppler radars," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 1592–1597.

[38] T. Grebner, P. Schoeder, F. Konrad, and C. Waldschmidt, "Instantaneous ego-motion estimation based on ambiguous velocity information within a network of radar sensors," in *Proc. 14th German Microw. Conf. (GeMiC)*, May 2022, pp. 140–143.

[39] S. Zhu, A. Yarovoy, and F. Fioranelli, "DeepEgo: Deep instantaneous ego-motion estimation using automotive radar," *IEEE Trans. Radar Syst.*, vol. 1, pp. 166–180, 2023.

[40] M. Rapp, M. Barjenbruch, M. Hahn, J. Dickmann, and K. Dietmayer, "Probabilistic ego-motion estimation using multiple automotive radar sensors," *Robot. Auto. Syst.*, vol. 89, pp. 136–146, Mar. 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889016300525

[41] S. Yuan, S. Zhu, F. Fioranelli, and A. G. Yarovoy, "3-D ego-motion estimation using multi-channel FMCW radar," *IEEE Trans. Radar Syst.*, vol. 1, pp. 368–381, 2023.

[42] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with apphcatlons to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[43] B. Zitová and J. Flusser, "Image registration methods: A survey," *Image Vis. Comput.*, vol. 21, no. 11, pp. 977–1000, Oct. 2003.

[44] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[45] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. 9th Eur. Conf. Comput. Vis.*, vol. 3951. May 2006, pp. 404–417.

[46] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "KAZE features," in *Computer Vision—ECCV 2012*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Heidelberg: Springer, 2012, pp. 214–227.

[47] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.

[48] J. Weickert, B. M. T. H. Romeny, and M. A. Viergever, "Efficient and reliable schemes for nonlinear diffusion filtering," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 398–410, Mar. 1998.

[49] T. Lindeberg, "Feature detection with automatic scale selection," *Int. J. Comput. Vis.*, vol. 30, no. 2, pp. 77–116, 1998.

[50] J. Weickert and H. Scharr, "A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance," *J. Vis. Commun. Image Represent.*, vol. 13, nos. 1–2, pp. 103–118, Mar. 2002.

[51] M. Brown and D. Lowe, "Invariant features from interest point groups," in *Proc. Brit. Mach. Vis. Conf.*, 2002, ch. 2, p. 3.

[52] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 5, pp. 698–700, Sep. 1987.

[53] W. Burgard et al., "A comparison of SLAM algorithms based on a graph of relations," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2009, pp. 2089–2095.

[54] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g^2o: A general framework for graph optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 3607–3613.

[55] V. Winkler, "Range Doppler detection for automotive FMCW radars," in *Proc. Eur. Microw. Conf.*, 2007, pp. 1445–1448.

[56] T. Grebner, M. Linder, N. Kern, P. Schoeder, and C. Waldschmidt, “6D self-calibration of the position and orientation of radar sensors in a radar network,” in *Proc. 19th Eur. Radar Conf. (EuRAD)*, Sep. 2022, pp. 157–160.

[57] M. Steiner, M. Keller, J. Geiß, M. Vossiek, and C. Waldschmidt, “Synchronization of radar sensors in a network based on inter-sensor interference,” in *Proc. 16th Eur. Radar Conf. (EuRAD)*, Oct. 2019, pp. 229–232.

[58] Texas Instruments. (2020). *IWR1443BOOST Evaluation Module mmWave Sensing Solution*. [Online]. Available: https://www.ti.com/lit/ug/swru518d/swru518d.pdf?ts=1697198330047&refurl=https%253A%252F%252Fwww.google.com%252F

[59] S. Tebaldini et al., “Sensing the urban environment by automotive SAR imaging: Potentials and challenges,” *Remote Sens.*, vol. 14, no. 15, p. 3602, Jul. 2022.

Timo Grebner (Graduate Student Member, IEEE) received the master’s degree in electrical engineering from Ulm University, Ulm, Germany, in 2019, where he is currently pursuing the Ph.D. degree in electrical engineering with the Institute of Microwave Engineering.

In 2018, he was an Intern with Sony, Japan. His current research interests include the creation of high-resolution occupancy gridmaps by high accuracy ego-motion estimations in a distributed and collaborative radar sensor topology.

Ron Riekenbrauck (Graduate Student Member, IEEE) received the M.Sc. degree in electrical engineering from Ulm University, Ulm, Germany, in 2022, where he is currently pursuing the Ph.D. degree with the Institute of Microwave Engineering. His research interests include mapping and agriculture surveillance with UAV-mounted radars.

Christian Waldschmidt (Fellow, IEEE) received the Dipl.-Ing. (M.S.E.E.) and Dr.-Ing. (Ph.D.E.E.) degrees from the Karlsruhe Institute of Technology, Karlsruhe, Germany, in 2001 and 2004, respectively.

From 2001 to 2004, he was a Research Assistant with Institut für Hochfrequenztechnik und Elektronik, Universität Karlsruhe. Since 2004, he has been with Robert Bosch GmbH, in the business units Corporate Research and Chassis Systems. He was heading different research and development teams in microwave engineering, RF-sensing, and automotive radar. In 2013, he returned to academia. He was appointed as the Director of the Institute of Microwave Engineering, University Ulm, Ulm, Germany, as a Full Professor. He has authored or coauthored more than 200 scientific publications and more than 20 patents. His research interests include radar and RF-sensing, mm-wave and submillimeterwave engineering, antennas and antenna arrays, and RF and array signal processing. He is a member of the Executive Committee Board of the German MTT/AP joint chapter and a member of the German Information Technology Society. He is a reviewer for multiple IEEE TRANSACTIONS and many IEEE conferences in the field of microwaves. He has been a co-recipient of 13 best paper awards since 2014. He is the Chair of the IEEE MTT-29 Technical Committee on Microwave Aerospace Systems and he was the Chair of MTT-27 Technical Committee on Wireless Enabled Automotive and Vehicular Applications. He was a two-time TPC Chair and the General Chair of the IEEE MTT International Conference on Microwaves for Intelligent Mobility. Since 2018, he has been an Associate Editor of IEEE MICROWAVE WIRELESS COMPONENTS LETTERS (MTT).