2016 IEEE Intelligent Vehicles Symposium (IV)
Gothenburg, Sweden, June 19-22, 2016

Robust Localization based on Radar Signal Clustering

F. Schuster[1], M. Wörner[1], C.G. Keller[1], M. Haueis[1], C. Curio[2]

Abstract— **Significant advances have been achieved in mobile robot localization and mapping in dynamic environments, however these are mostly incapable of dealing with the physical properties of automotive radar sensors. In this paper we present an accurate and robust solution to this problem, by introducing a memory efficient cluster map representation. Our approach is validated by experiments that took place on a public parking space with pedestrians, moving cars, as well as different parking configurations to provide a challenging dynamic environment. The results prove its ability to reproducibly localize our vehicle within an error margin of below 1% with respect to ground truth using only point based radar targets. A decay process enables our map representation to support local updates.**

I. INTRODUCTION

This paper investigates the Simultaneous Localization and Mapping (SLAM) problem [1] applied to an automotive radar sensor in outdoor scenarios, focusing on the challenging task of lifelong mapping in a dynamic outdoor environment. Especially pose estimation in highly dynamic environments like public parking spaces provides a challenging setting. Radar sensors are indifferent to changing weather conditions, inexpensive and can be placed behind the bumpers of the car because of the penetration depth of millimeter waves. Therefore they are well established in the automotive industry [2], although the sensors lack in accuracy, have a high noise floor [3] and have rarely been applied to the task of vehicle localization. Unlike laser scanners, the radar measurements are not taken within fixed angular intervals, but rather come as a mixed point cloud with varying numbers of detections. Additionally the penetration depth depends on the type of reflective matter.

We show how a robust and low-noise map representation can still be obtained by applying a clustering algorithm to the noisy radar measurements of these sensors. The resulting map is stored in an R-tree data structure to maintain maximum flexibility, while keeping access times low. For the purpose of validating this map representation, extensive data has been collected covering a parking space with changing vehicle configurations over an extended period of time. Our experimental results indicate, that the presented algorithm named Cluster-SLAM makes a step towards lifelong navigation in urban scenarios.

Many of the current approaches to localization involve the use of particle filters on grid maps to accumulate data over time and reduce noise in the map. In radar based SLAM, existing approaches often use the full spectral information of the scan [4], [3], [5]. In contrast to these, Cluster-SLAM utilizes reduced data in form of point based radar targets common in the automotive industry [2]. These contain amplitude, velocity and spatial coordinates without covariances. Cluster-SLAM solves several problems simultaneously. Its noise handling capabilities, low memory requirements and adaptability to changing environments make it outperform the classical grid representation in numerous aspects.

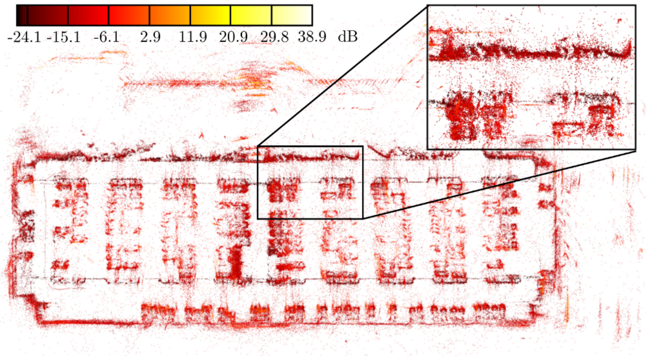

Fig. 1. Example of raw radar data collected in the parking space scenario. The coloring scheme indicates the amplitude of the signal in dB.

Cluster-SLAM is implemented as a particle filter, which is specifically designed to cope with radar data directly. By applying a density based stream clustering algorithm to the incoming sensor readings, noisy measurements can be disregarded before becoming part of the map.

An exemplary data set of a parking space is depicted in Fig. 1 showing the residual amplitude returned to the radar sensor. The top right cutout shows a magnified version of the data illustrating its noise level. The parking space is approximately a $150\,\mathrm{m}$ by $35\,\mathrm{m}$ area with nine double rows of parking spots and is surrounded by fences and vegetation.

In context of producing an accurate position estimate in the scenario of once manually driving into a designated parking lot and having an actuator reproduce the trained trajectory, the pose of the vehicle has to be determined reliably. The parking lot scenario proved to be a very complex scenario due to the different vehicle configurations, sensor occlusion in tight spaces and dynamic objects. This is well modeled with the given data set, since changing occupancy of the parking space has been recorded for several days.

II. RELATED WORK

The research field of SLAM has been popular over the past years, resulting in a large variety of SLAM algorithms for different settings and unknown environments. An overview is outlined in [1]. The algorithms can be

[1] Vehicle Localization, Department of Environment Perception, Daimler AG {frank.fs.schuster, marcus.woerner, christoph.g.keller, martin.haueis}@daimler.com

[2] Cognitive Systems, Department of Computer Science, Reutlingen University cristobal.curio@reutlingen-university.de

978-1-5090-1821-5/16/$31.00 ©2016 IEEE

divided into feature based approaches, such as [6] and grid based ones, as overviewed in [7]. Feature based approaches store a highly reduced amount of information about the environment, which result in low memory consumption, at the cost of robustness due to association problems [8]. These techniques are common for camera sensors [9] and have recently been applied to the radar sensor by Rapp et. al [10]. Grid based algorithms on the other hand have proven to be effective combined with range sensors, such as laser scanners and radars [11]. Because particle filters constantly validate multiple hypotheses to determine the most likely map, a lot of effort was put into making grid search and access operations more efficient. This can be done by approximating regularly shaped areas in the map by polygons [12], by making spatial search more efficient with octrees [13], [14] or by modifying the measurement model of the particle filter such that it operates in a single global map [11].

Based on these advances, we will focus our attention on the specific setting given for this paper using an FMCW radar sensor, which emits millimeter waves and records the reflected power distribution with respect to distance, the so called A-scope. In the automotive industry, A-scopes are preprocessed and packaged into target lists, containing a reduced amount of information [2].

Exploration of radar based SLAM algorithms has yielded insular results with very different approaches. One of the biggest concerns in radar based mapping remains sensor noise and thus the robustness [3]. Hence in [4] for seaborne vessels, a grid map is generated over several successive scans. The resulting map is treated as an image, such that common feature detection algorithms generate landmarks on the coast used for SLAM. Another approach [5] uses scan matching on the reflected radar spectra in combination with EKF-SLAM. By Fourier-Mellin transformation, the authors find an efficient technique to match scans and generate position hypotheses. This approach relies on the full spectral information. For the task of object classification using shape information, e.g. cars [15], grid maps can be utilized.

For the scenario of repeatedly returning to a specific parking space at different times of the day, it is vital to refresh the map continuously with every passing. While with feature based SLAM, the map update can be achieved fairly easy, more involved methods have to be devised to keep grid maps up to date [8], [16]. Because of the static nature of grid maps, transition matrices have to be defined in order to calculate the probability of grid cell occupancy. The presented Cluster-SLAM approach does not require this.

Feature and grid based mapping both have their merits and drawbacks. In this paper we introduce a method that effectively combines advantages of both map types by use of a stream clustering algorithm [17]. In data mining applications, evolving data sets need to be clustered. Popular classes of algorithms include density based solutions, most notably DBSCAN [18], because it does not assume a specific number or shape of clusters. In recent years, research was focused on developing algorithms for data streams, processing an arbitrary amount of data with limited time and memory, demanding a single pass solution. The algorithm described in [19] is specifically designed for evolving data sets with noise. We have chosen to adapt this algorithm for Cluster-SLAM, because it suits the radar sensor's characteristics very well. Incoming data points are organized into micro-clusters representing regions of high data density.

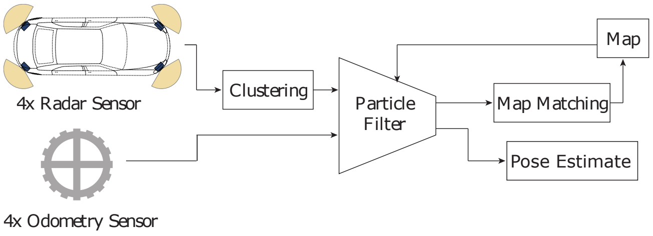

Fig. 2. System architecture used to localize the vehicle based on radar and odometric sensors.

A similar idea using graph based SLAM and a Normal Distribution Transformation (NDT), intended for lifelong navigation, was shown in [20]. We present a more flexible approach for map generation, using a FAST SLAM algorithm based on particle filters. We introduce a new approach combining stream clustering and FAST SLAM to make a step towards lifelong navigation, which is currently an active area of research [21]. The goal of this work is to continuously update the map. We show that successful map update is accomplished easily. This results in accurate localization results with noisy sensors in real world scenarios. The contributions of this paper are thus:

1) A robust and memory efficient map representation
2) A measurement model approximating the physical behavior of a radar sensor
3) A robust SLAM implementation combining the above in a particle filter approach.

III. SENSOR SETUP

The approach described in IV is based on the experimental setup as displayed in Fig. 2. The sensors are mounted behind the front and rear bumper of a Mercedes E Class, which makes them invisible from the outside. The field of view spans across 360 degrees, while each sensor covers 150 degrees. Each radar operates at $76\,\mathrm{GHz}$ and has a range of $40\,\mathrm{m}$ with an accuracy of $0.15\,\mathrm{m}$ and the angular resolution is listed as below one degree. The Electronic Control Unit (ECU), generating the radar targets from the raw spectra, transmits up to 64 targets per radar, resulting at a maximum of 256 radar points per scan at a rate of $20\,\mathrm{Hz}$. The preprocessed data consists solely of distance r, angle ϕ and amplitude A. No explicit covariance information is given per point, thus only the above mentioned accuracies are taken into account. The vehicle's odometry is derived from a yaw rate reading from an IMU and wheel encoders with 96 ticks per revolution. Such a setup is common in use for driver assistance system development and is therefore also used for this localization approach

Ground truth is acquired by the iMAR iTrace F400-E [22], a precise DGPS receiver combined with INS sensors, with an accuracy of up to $2\,\mathrm{cm}$.

IV. CLUSTER-SLAM

Raw radar data has a high noise floor, because of the interference of the transmitted wave packets with each other or due to multiple reflections increasing the transmission delay of the waves. Additionally, unlike laser scanners, radar waves have a macroscopic penetration depth into matter, depending on the type of material.

Our system is displayed schematically in Fig. 2. Raw data is clustered to form a scan of the environment. Using scan matching, this scan is compared and merged with a reference map to obtain the weights necessary for the particle filter utilized by Cluster-SLAM. Each particle contains its own hypothesized version of the reference map, to which the current scan is compared.

A. Map Representation

The raw data consisting of the two dimensional position x, y, and the amplitude A is clustered by a stream clustering algorithm inspired by [19]. They introduce two types of *micro clusters* defined as a group of close points. The circular clusters are depicted in Fig. 3 and can be described by the following properties:

$$W_C = \sum_{i=1}^{n} w_i = \sum_{i=1}^{n} \prod_{j=1}^{m-1} 1 - \mathcal{N}(d_{ij}|C_j, R_j) \quad (1)$$

$$C_i = \frac{1}{W_C} \sum_{i=1}^{n} c_i = \frac{1}{W_C} \sum_{i=1}^{n} p_i w_i \quad (2)$$

$$C_{sq} = \frac{1}{W_C} \sum_{i=1}^{n} p_i^2 w_i \quad (3)$$

$$R_i = \sqrt{C_{sq} - C^2} \quad (4)$$

In the context of this work, the weight w_i for all n radar targets is calculated from the Gaussian probability $\mathcal{N}(x|\mu, \sigma)$ that a radar target is visible through the $m-1$ other clusters in the map, as illustrated in Fig. 3. This is done to approximate the radar sensor's physical properties of multiple scattering and penetration depth. We assume that another cluster does not completely obstruct the line of sight, but rather only the center of a cluster where the Gaussian distribution is maximal, the line of sight is interrupted.

Hence, the total weight W_C of the cluster is the sum of the individual target weights w_i. Two cluster types are defined: *Potential* and *outlier* micro-clusters, further on called p and o-micro-clusters, respectively. All particles exceeding a threshold $W_C \geq W_{\text{thresh}}$ are called p-micro-clusters. The center point C is given by the center of mass of the radar targets according to their weights. The radius R is defined as the standard deviation of the target distribution, whereas the squared weighted sum C_{sq} is used for simplification. In practice, we limit the weight by a parameter $W < W_{\max}$ to define a saturation limit.

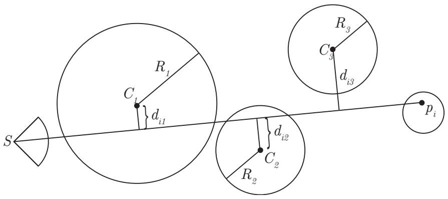

Fig. 3. The cluster weight for a given point p_i is calculated using the normal distance d_{ij} of the center points C_j from the line of sight originating from the sensor S for each p_i. The radius R_j indicates the 1σ environment.

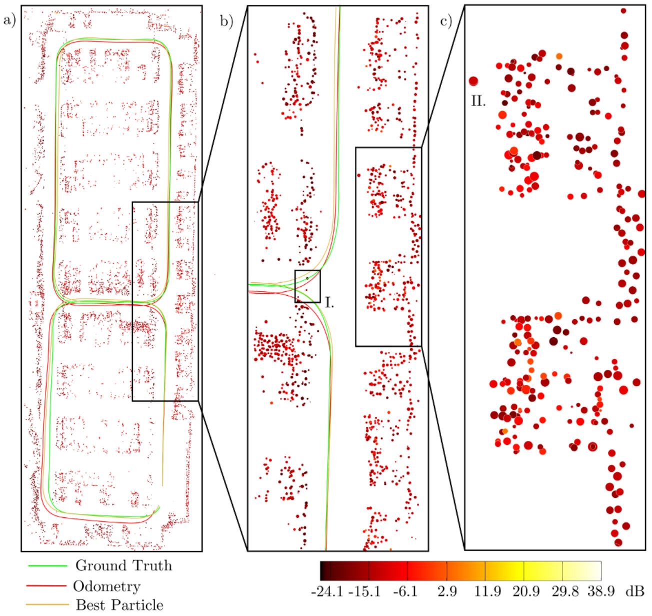

Fig. 4. Map representation using p-micro-clusters in three levels of detail. As before, the applied color scheme represents the mean amplitude of the clusters. The ground truth trajectory is shown in green, odometry in red and the best particle hypothesis in orange.

These micro-clusters can be maintained incrementally allowing temporal accumulation of the radar targets [19]. Any new measurement p_{n+1} is absorbed into the nearest cluster according to the Mahalanobis distance, if the new radius is below the radius ϵ, encompassing the measurement uncertainties mentioned above. Then the properties (1) – (4) are evaluated accordingly. Note that adding another data point to a cluster can be done iteratively, such that only a single pass is necessary. In our approach, we accumulate data for $t_{acc} = 1\,\text{s}$ to enhance the amount of information that is compared to the reference map. Thus, the map consists of a set of clusters $\mathcal{M}_p$ containing the accumulated radar targets with properties (3) – (4). Unlike in grid structures, clusters can be placed arbitrarily in 3D space containing the spatial coordinates x,y and amplitude A. Individual clusters are stored in an R-tree data structure for the purpose of efficient spacial search, as well as fast insertion and deletion.

In Fig. 4, a cluster map is displayed in various levels of detail. Let the reader be reminded that only p-type micro-clusters are kept in the final map, thus leaving regions of low measurement density free of clusters. In image 4 c), one can see the outline of four parked cars grouped in pairs leaving a space between them in the cluster representation. A common problem when dealing with measurements from automotive radar sensors are reflections caused by metal surfaces. The single outlier (II.) is due to a strong reflection from one of the cars. Similarly, in image 4 b), one can see a few clusters (I.) blocking the trajectory of the car. This is caused by reflections emitted from a rain gutter embedded in the road's surface. Note that overlapping clusters exist, as can be seen in the right image. This is due to the dimensionality of the micro-clusters using (x, y, A).

To further increase the robustness of the map and to handle dynamic environments, each micro-cluster decays with time. We reduce the weight W_C of the p-micro-clusters in such a way that if a cluster is not observed over an extended period of time, it gets degraded to an outlier, which is disregarded instantly. This is modeled by an effective cluster weight

$$W_{C,\mathrm{decay}} = \sum_{i=1}^{n} \prod_{j=1}^{m-1} 1 - \mathcal{N}(d_{ij}|C_j, R_j) - \beta w_m. \qquad (5)$$

The free parameter β is a decay constant and w_m is the weight of a unit cluster, one where all measurements have been directly observed without occlusions. Thus each cluster has to be observed frequently in order to remain a p-micro-cluster. If the weight drops below the above mentioned threshold W_{thresh}, it is degraded to an outlier and thus disregarded for the pose estimate in the particle filter. On the other hand, if a cluster has not been observed recently but is present once again, an outlier will be promoted to p-micro-cluster within a few observations. The threshold is found by determining the average number of sightings of a highly reflective object scaled down by an empiric constant.

The reader may notice the disturbing factors (i.e. cars, pedestrians, reflections and noise) are usually non stationary, thus they can be suppressed by a decay on the individual clusters. Therefore only the stationary objects will remain in the map. Reflections are usually non stationary if the vehicle is moving, as reflecting objects are measured from different viewing angles.

Hence, when a car moves past the reflecting object, the decay eliminates the clusters produced by reflection. For the same reason, the decay helps eliminate moving objects while mapping, because they are not detected in the same location every time. Lastly, the decay is used to update the map due to environmental changes, such as cars not being parked in the same location as before. Even though such clusters usually have a large weight, since no measurements are updating the cluster, they are eliminated eventually, since they are not measured at all in the second passing.

B. Particle Filter

We follow the classical *predict − update* cycle to propagate the particle distribution with respect to the odometry and radar sensor information acquired from the motion and measurement model, respectively.

1) Prediction: The vehicle velocity $\mathbf{v}$ and yaw rate ψ_v are extracted from the Controller Area Network (CAN) using on board sensors. Using a standard single track model, the odometry pose is estimated. We assume that within two consecutive time steps $\Delta t = t_2 - t_1$, the car translates by $\mathbf{\Delta} x_v = \mathbf{v}\Delta t$ and is rotated by $\Delta\psi_v$. We assume $\mathbf{v}$ to be constant within Δt. The motion model is an adaptation of the sampling motion model described in standard literature [23], applied to the integrated velocity and yaw rate. The error is modeled with four free parameters α_i given in the sampling algorithm.

2) Update: Each particle consists of a reference set $\mathcal{M}_p$ filled with $n_{\mathcal{M}_p}$ clusters called *map* in further elaboration. Let a scan $\mathcal{S}_t$ be a set of $n_{\mathcal{S}_t}$ p-micro-clusters accumulated over time t_{acc} at time t. Map matching between $\mathcal{S}_t$ and $\mathcal{M}_p$ is applied to derive the particle weight.

Each micro-cluster $s_i \in \mathcal{S}_t$ is assigned to its nearest neighbor $m_j \in \mathcal{M}_p$ in terms of the Mahalanobis distance as described above. Similar to [24] the scan S_t is assigned a score

$$D(\mathcal{S}_t, \mathcal{M}_p) = \sum_{i=1}^{n_{\mathcal{S}_t}} \sum_{j=1}^{n_{\mathcal{M}_p}} D_j \mu_{ij}^T (\Sigma_i' + \Sigma_j)^{-1} \mu_{ij}, \qquad (6)$$

whereas $\Sigma_i' = R^T \Sigma_i R$ is the covariance matrix transformed into the coordinate system of $\mathcal{M}_p$. The mean vector distance $\mu_{ij} = R\mu_i + T - \mu_j$ is transformed accordingly. The parameter D_j represents the mean distance between a micro-cluster in a particle's reference map and any pose, this micro-cluster was observed from. The scoring function is applied to all clusters in the scan $\mathcal{S}_t$. According to this scoring function, the particles are weighted.

3) Resampling: Once all particle weights have been obtained, resampling is done using a low variance sampling algorithm. This standard approach agrees very well with the requirements of this paper's setting, especially a low computing complexity and robustness of the technique due to statistically dependent sampling the weight distribution of the particle set $\mathcal{P}$.

4) Lost Robot: After the initial mapping, particles can be scattered across the entire map, depending on the knowledge of the initial conditions to solve the lost robot problem. In this paper, we assume that we know the position of the robot within the accuracy of a standard vehicle GNSS system, Hence, we scatter the initial pose of the vehicle over an area of $16\,\mathrm{m}$ and 90 degrees in heading around the origin of the map. This is merely done to reduce the initial size of the state space such that the particles converge quickly to the true vehicle pose.

V. EXPERIMENTAL RESULTS

As mentioned in the introduction, the described method not only allows the generation of a single map but also a continuous update. We evaluate the localization and map update capability on a large scale dataset collected with a prototype vehicle from Daimler AG. The set includes 32 trajectories on the parking lot in front of one of Daimler's R&D facilities.

A. Characterization of Ground Truth

Ground truth has been recorded across the entire data set and a statistical analysis of the standard deviation reported by the GNSS/INS system is given by the following quantities. The mean error margin of the ground truth for the entire data set is given by: $\bar{\sigma}_{\mathrm{GT,d}} = (0.094 \pm 0.315)\,\mathrm{m}$ The deeply

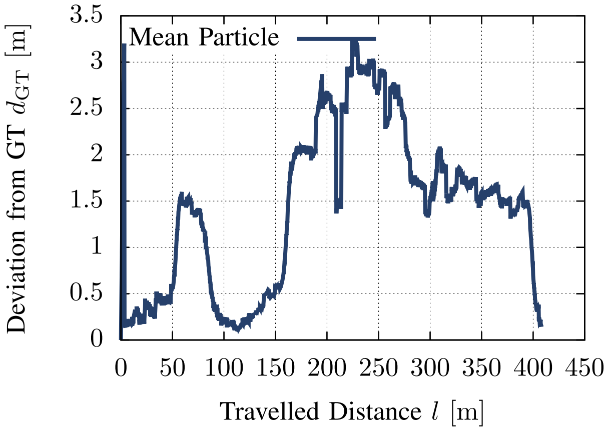

Fig. 5. Deviation from ground truth (Euclidean distance) d_{GT} for the pose error during the map creation phase of the best particle at 16:00 when the parking lot is highly occupied. The loop closure has been detected with an error at the end point of $0.143\,\mathrm{m}$

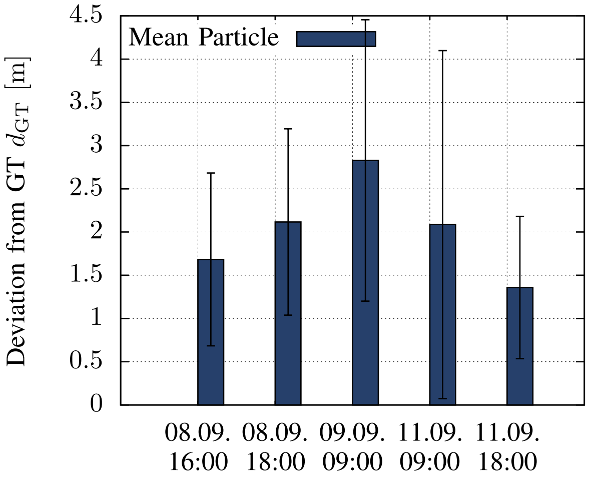

Fig. 6. The mean error of the mapping process for all sequences in the data set sorted by time of the day, which is an indicator for parking lot occupancy. The error bars indicate a 1σ environment over all sequences.

coupled GNSS/IMU solution is a suitable reference system, because of its very high accuracy and reliability. The scenario in the parking lot has explicitly been chosen due to the high availability of the DGPS system to ensure high quality ground truth data.

B. Initial Mapping Process

During the mapping process, the clustering algorithm is completely relying on sensor readings. Therefore, it is essential that the particle filter produces a map that is internally consistent. Most importantly, loop closures have to be detected and the error at the end of the trajectory has to be minimal.

In Fig. 5 an optimized map from the particle filter is shown. The particle trajectory is very close to the ground truth in most cases. One of the most important criteria for map quality is whether loop closes can be detected. Since the trajectory is closed, we compare the end point error for mapping runs at each time of the day in Fig. 6. The main thing to note is that the error is basically independent of data collection time and thus of parking configuration. At 9:00 the parking space was usually only partially filled, while at 16:00 almost every parking lot was occupied. By 18:00, almost all cars had disappeared.

C. SLAM With Existing Map

In Section V-B it was shown that the end point error can be decreased by optimizing the map with Cluster-SLAM. Therefore we apply the decay mechanism described above to model environmental changes in the measurement update. The map update is graphically shown in Fig. 7 and analyzed quantitatively in Fig. 8.

Fig. 8 shows that despite changing environments, the localization of Cluster-SLAM yields stable results, even though at 18:00, the parking space was almost empty. For the localization within the map, we assumed that the starting position is known with standard vehicle GPS accuracy of about 4 m in radius and 20 degrees in heading. After the initial convergence phase of the particle filter (about 50 m) yields an accuracy of maximally $2\,\mathrm{m}$. A localization step adds new clusters to the map, while others decay, resulting in a map size of approx. $200\,\mathrm{kB}$ for a sequence from the given data set. This is several orders of magnitude smaller than grid based map representations, which typically amount to several MB for the same area. The memory and computing requirements thus allow this algorithm to be real time capable.

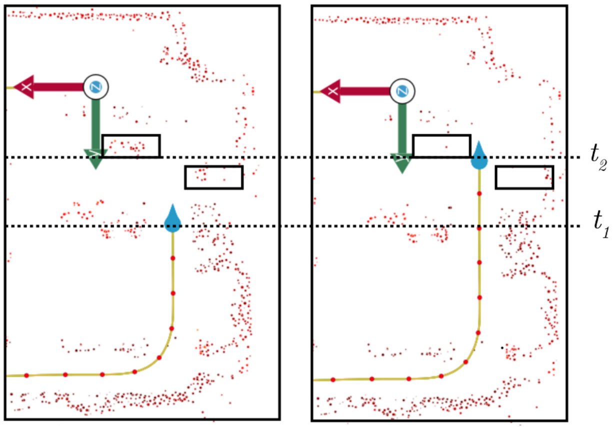

Fig. 7. The vehicle passes through a region that changed since the initial mapping. Between the time steps t_1 and t_2, the map update with cluster decay removes the two indicated cars, while preserving the immovable structures.

VI. CONCLUSION

We have presented a SLAM technique using automotive radar targets and odometric sensors, developing a map from a stream clustering algorithm. The map representation is especially suited for dynamic environments, which was validated using an extensive data set of a public parking space with different configurations of parked cars. Environment dynamics are managed by cluster decay, which adds robustness to the system. We have shown that the presented map is suitable

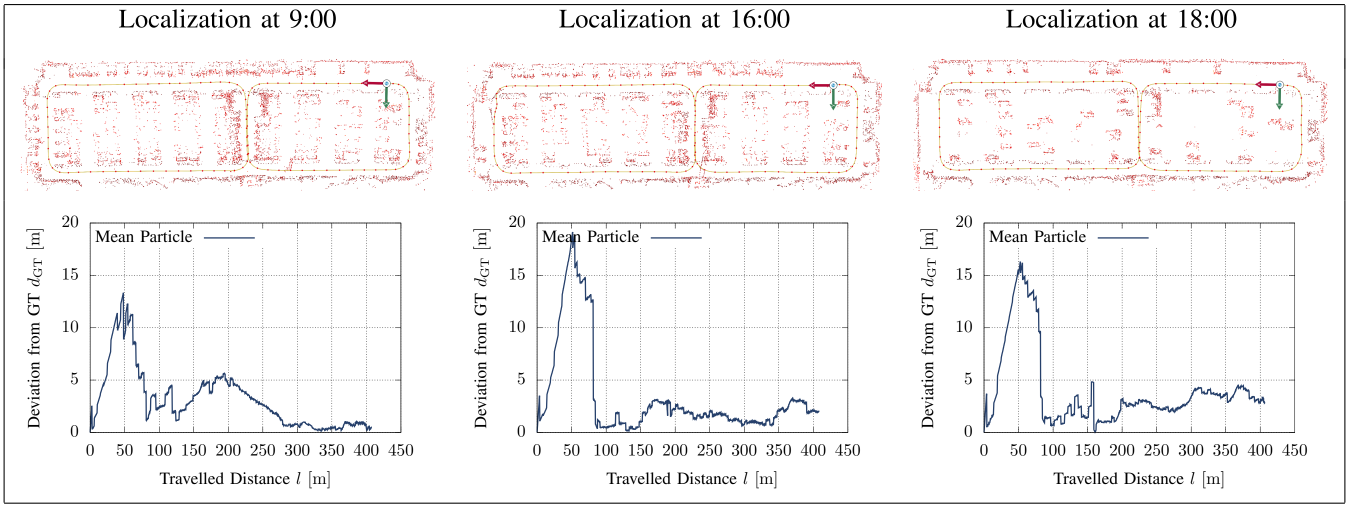

Fig. 8. Localization results after initial mapping with vehicle configuration from 16:00. The respective vehicle configurations are illustrated as ground truth maps at the top. Each localization error with respect to the traveled distance compared to the odometry result is shown directly below.

for localization using Cluster-SLAM. To suit the cluster map, a specially designed weight function has been conceived to produce an accurate localization result despite the noisy and inaccurate sensor data. The map representation combines the advantages of feature and grid based algorithms alike, containing a high information density, while consuming little memory. Thus, our approach constitutes a significant step towards lifelong mapping in unexplored scenarios.

Future work will omit the necessity of storing a map for each particle. This improves memory requirements at runtime. Improvements in localization accuracy across the entire trajectory are planned.

References

[1] J. Li, L. Cheng, H. Wu, L. Xiong, and D. Wang, "An Overview of the Simultaneous Localization and Mapping on Mobile Robot," in *IEEE International Conference on Modelling, Identification Control. Proceedings.*, June 2012, pp. 358–364.

[2] M. Bühren and B. Yang, "Simulation of Automotive Radar Target Lists using a Novel Approach of Object Representation," in *IEEE Intelligent Vehicles Symposium. Proceedings.*, 2006, pp. 314–319.

[3] J. Mullane, B.-N. Vo, M. Adams, and B.-T. Vo, "A random-finite-set approach to bayesian slam," *IEEE Transactions in Robotics*, vol. 27, no. 2, pp. 268–282, April 2011.

[4] J. Callmer, D. Törnqvist, F. Gustafsson, H. Svensson, and P. Carlbom, "Radar SLAM Using Visual Features," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, no. 1, 2011.

[5] P. Checchin, F. Gerossier, C. Blanc, R. Chapuis, and L. Trassoudaine, "Radar Scan Matching SLAM Using the Fourier-Mellin Transform," in *Field and Service Robotics*, ser. Springer Tracts in Advanced Robotics, A. Howard, K. Iagnemma, and A. Kelly, Eds. Springer Berlin Heidelberg, 2010, vol. 62, pp. 151–161.

[6] S. Thrun and M. Montemerlo, "The GraphSLAM Algorithm With Applications to Large-Scale Mapping of Urban Structures," *International Journal on Robotics Research*, vol. 25, no. 5/6, pp. 403–430, 2005.

[7] T. Collins, J. Collins, and C. Ryan, "Occupancy grid mapping: An Empirical Evaluation," in *IEEE Mediterranean Conference on Control and Automation. Proceedings.*, June 2007, pp. 1–6.

[8] H. Zhao, M. Chiba, R. Shibasaki, X. Shao, J. Cui, and H. Zha, "SLAM in a Dynamic Large Outdoor Environment Using a Laser Scanner," May 2008, pp. 1455–1462.

[9] J. Ziegler, H. Lategahn, M. Schreiber, C.G.Keller, C. Knoppel, J. Hipp, M. Haueis, and C. Stiller, "Video Based Localization for Bertha," in *IEEE Intelligent Vehicles Symposium. Proceedings.*, June 2014, pp. 1231–1238.

[10] M. Rapp, T. Giese, M. Hahn, J. Dickmann, and K. Dietmeyer, "A feature-based approach for group-wise grid map registration," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, Sept 2015, pp. 511–516.

[11] A. Eliazar and R. Parr, "DP-SLAM: Fast, Robust Simultaneous Localization and Mapping without Predetermined Landmarks," in *International Joint Conference on Artificial Intelligence. Proceedings.* Morgan Kaufmann, 2003, pp. 1135–1142.

[12] M. A. Paskin and S. Thrun, "Robotic Mapping with Polygonal Random Fields," in *Conference on Uncertainty in Artificial Intelligence. Proceedings.*, F. Bacchus and T. Jaakkola, Eds. AUAU Press, Arlington, Virginia, July 2005.

[13] N. Fairfield, G. A. Kantor, and D. Wettergreen, "Real-Time SLAM with Octree Evidence Grids for Exploration in Underwater Tunnels," *Journal of Field Robotics*, 2007.

[14] K. M. Wurm, A. Hornung, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: A Probabilistic, Flexible, and Compact 3D Map Representation for Robotic Systems," in *IEEE International Conference on Robotics and Automation. Proceedings.*, 2010.

[15] R. Dube, M. Hahn, M. Schutz, J. Dickmann, and D. Gingras, "Detection of Parked Vehicles from a Radar Based Occupancy Grid," in *IEEE Intelligent Vehicles Symposium. Proceedings.*, June 2014, pp. 1415–1420.

[16] D. Meyer-Delius, M. Beinhofer, and W. Burgard, "Occupancy Grid Models for Robot Mapping in Changing Environments," 2012.

[17] A. Amini, T. Wah, and H. Saboohi, "On Density-Based Data Streams Clustering Algorithms: A Survey," *Journal of Computer Science and Technology*, vol. 29, no. 1, pp. 116–141, 2014.

[18] M. Ester, H. P. Kriegel, J. S, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *AAAI Conference on Artificial Intelligence. Proceedings*. AAAI Press, 1996, pp. 226–231.

[19] M. Estert, F. Cao, W. Qian, and A. Zhou, "Density-Based Clustering over an Evolving Data Stream with Noise," in *SIAM International Conference on Data Mining. Proceedings.*, 2006, pp. 328–339.

[20] E. Einhorn and H.-M. Gross, "Generic 2D/3D SLAM with NDT Maps for Lifelong Application," in *European Conference on Mobile Robots. Proceedings.*, Sept 2013, pp. 240–247.

[21] M. Milford and G. Wyeth, "Persistent Navigation and Mapping using a Biologically Inspired SLAM System," *The International Journal of Robotics Research*, vol. 29, no. 9, pp. 1131–1153, 2010.

[22] Gesellschaft für inertiale Mess-, Automatisierungs- und Regelsysteme mbH, "iTraceRT-F400-E: Accurate Real-Time Surveying, Vehicle Trajectory and Dynamics Estimation with deeply coupled INS/GNSS Filtering."

[23] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[24] T. Stoyanov, M. Magnusson, and A. Lilienthal, "Point Set Registration Through Minimization of the L2 Distance Between 3D-NDT Models," in *IEEE International Conference on Robotics and Automation. Proceedings.*, May 2012, pp. 5196–5201.