Pole-based Localization for Autonomous Vehicles in Urban Scenarios

Robert Spangenberg[1] and Daniel Goehring[1] and Raúl Rojas[1]

***Abstract*— Localization is a key capability for autonomous vehicles especially in urban scenarios. We propose the use of pole-like landmarks as primary features in these environments, as they are distinct, long-term stable and can be detected reliably with a stereo camera system. Furthermore, the resulting map representation is memory efficient, allowing for easy storage and on-line updates. The localization is performed in real-time by a stereo camera system as a main sensor, using vehicle odometry and an off-the-shelf GPS as secondary information sources. Localization is performed by a particle filter approach, coupled with an Kalman filter for robustness and sensor fusion. This leads to a lateral accuracy below 20 cm in various urban test areas. The system has been included in our autonomous test vehicle and successfully demonstrated the full loop from mapping to autonomous driving.**

I. INTRODUCTION

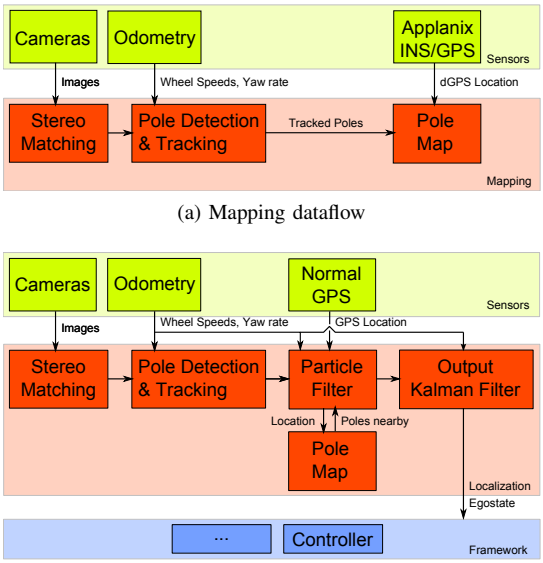
Autonomous vehicles need maps to reduce the perceptual ambiguity while navigating through traffic. Accurate maps make the full problem tractable. In order to use additional information provided by map layers, the vehicle has to localize itself under real-time constraints and with a very high precision. The classic approach to use very accurate INS/DGPS based localization solutions [1] has several drawbacks. The sensor itself is costly and in urban areas GPS suffers from low availability due to non-visibility of satellites and systematic error sources as multi-path [2], [3].

This has lead to approaches that rely on the use of either implicit or explicit landmarks to localize autonomous vehicles. These landmarks are detected using active sensors like Lidar and Radar or camera systems. Landmark classes vary from flat features like road markings or visual monocular features to three-dimensional features like planes, corners or poles. Stereo camera-based systems are an interesting and cost-effective option to replace the classic approaches.

II. RELATED WORK

The intensity information provided by a rotating laser scanner is sourced in [4] for correlation with a dense intensity map resulting in robust results even in dynamic scenarios. In [5] a hierarchical approach to direct matching of occupancy grids created from laser scanner data is detailed, leading to a reliable but still costly approach. The fraction of dynamic objects in the visible area influences the achievable accuracy, as they are not explicitly modeled.

The explicit extraction of landmarks for vehicle localization, like planes and pole-like objects, is proposed in [6]. The ambiguity arising from lower feature density by an automotive laser scanner is tackled by combined super-features as proposed in [7]. In contrast to pure intensity-based approaches a decreased change rate of the map is expected as well as a reduced weather dependency. Furthermore, the majority of dynamic objects can be excluded from the map by landmark-design.



(a) Mapping dataflow

(b) Localization dataflow

Fig. 1: Dataflows for mapping and localization

Road markings detected in aerial photographs are used to create maps in [8] and are detected online by a stereo-camera rig. Laser-scanner landmarks are combined with lane markings, vehicle odometry and GPS using a particle filter in [9]. An extended Kalman filter is used in [10] to fuse radar targets, vehicle odometry and lane markings and GPS. In [11] a sigma-point filter fuses structural road information as lane markings and curbs as well as visual features utilizing data from an inertial measurement unit (IMU). A combination of front and rear-facing cameras provides the necessary images. Lane markings detected with two side-cameras provide lateral and orientation constraints in [12], processed with GPS and IMU data.

Visual features detected by a Maximally Stable Extremal Region (MSER) detector [13] are leveraged in [14] for localization purposes. Lidar-based pole like objects are fused with lane markings detected by front and rear-facing cameras in [15] with a particle filter, using vehicle odometry for prediction. Hereby the lane markings are detected using an MSER approach. The poles are detected by applying the same detector on an occupancy map of the lidar data.

[1]all authors are with Institut für Informatik at Freie Universität Berlin, Arnimallee 7, 14195 Berlin, Germany robert.spangenberg@fu-berlin.de

While sharing the approach of detecting pole-like features in depth data and the particle filter approach with the last approach, this contribution focuses on the detection using only one front-facing stereo camera as a primary sensor. Pole-like structures as trees, lamp-posts, sign-posts, bollards or traffic-lights can be detected. Vehicle odometry and an off-the shelf GPS serve as secondary sensor inputs. We show that a particle filter approach suffices for autonomous driving in a wide variety of urban and dynamic scenarios.

Section III explains the key ideas of the approach, while Section IV provides details about the localization algorithm. The evaluation measures and results for several urban scenarios are explained in Section V. The final discussion of the results and future directions of work are covered in Section VI.

III. SYSTEM DETAILS

A. Overview

Figure 1 shows the system components used during mapping and localization. In the mapping step, a precise DGPS position provided by an Applanix system is used to put the tracked pole-like structures in a global pole map. Pole detection and tracking is performed in the same way in the mapping and localization stage working on the results of a disparity map computed by rSGM [16] (Fig. 2). Vehicle odometry is used for prediction, as the poles are tracked in a vehicle coordinate system.

The pole hypotheses are built by grouping contours generated from depth edges in the disparity image. Depth edges are detected using occlusion effects in the left and right disparity image, similar to ideas in [17]. They are classified using logistic regression before being fed into tracking. The currently tracked poles serve as the model of the environment of the vehicle and provide the measurements for the localization.

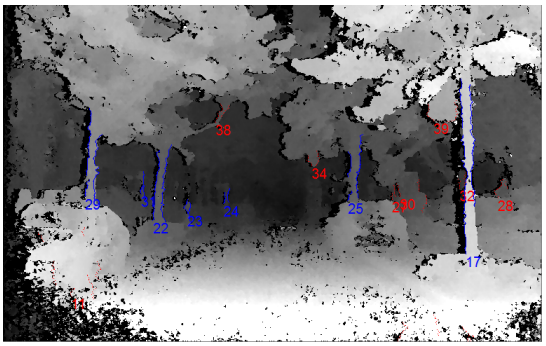
In the localization stage, an off-the-shelf GPS provides a coarse initialization for the particle filter. The pole map is queried for poles in the vicinity of the current location estimate. Based on these landmarks and the currently tracked poles the particle filter estimate is updated. An output Kalman filter is used to enable a 100 Hz update rate and to hide computational latencies. The localization result can be directly used in the autonomous framework for tasks like behavior planning and low-level control.

B. Sensor setup and stereo preprocessing

The test vehicle *Made in Germany* (MIG) is a Volkswagen Passat B6 equipped with a front-facing stereo camera system with a baseline of 30 cm. The cameras provide greyscale images at a resolution of 768×480 pixels with a high dynamic range image of 12 bit at a frame rate of 22 Hz covering a horizontal field of view of approximately 50 degrees. A standard GPS receiver (u-blox 6T) and a connection to the vehicle wheel speed and yaw rate sensors by CAN-bus complete the inputs. A high-precision DGPS/INS (Applanix POS LV 510) is used for mapping and as a reference system.

rSGM stereo matching is calculated on a CPU and uses 4 stripes, a border width of 16 pixels and 5×5 Census data costs. This adapts well to the effective angular resolution and delivers reliable depth data at comparably low processing times. The matching parameters are $P_1 = 7, P_{2_{min}} = 17, \alpha = 0.25, \gamma = 50$ (notation and parameter definition of rSGM in [16]).

(a) Rectified left image of stero camera (Campus area)



(b) rSGM disparity map with hypotheses of pole-like structures

Fig. 2: Hypothesis generation for pole-like structures – Blue hypotheses are classfied as pole-like structures and red ones as invalid.

IV. LOCALIZATION

A. State and prediction

The vehicle pose $\mathbf{p}$ is estimated by a particle filter with state $\mathbf{p}_k^i$ for time k and particle $i = 1, \ldots, M$. Particles having Universal Transverse Mercator (UTM) coordinates Easting E, Northing N and heading ψ as components: $\mathbf{p}_k^i = [E_i\, N_i\, \psi_i]^T$. Initialization is done with the standard GPS using horizontal dilution of precision for position uncertainty. Prediction is carried out with speed $\widetilde{v_k}$ and (offset compensated) yaw rate $\widetilde{\dot{\psi}_k}$ over Δt (bicycle model):

$$\Delta\psi_k = \widetilde{\dot{\psi}_k} \cdot \Delta t \quad (1)$$

$$\psi_{k+1} = \psi_k + \Delta\psi_k + \widetilde{\gamma} \quad (2)$$

$$E_{k+1} = E_k + \frac{\widetilde{v_k}}{\widetilde{\dot{\psi}_k}} \cdot [\sin(\psi_k + \Delta\psi_k) - \sin(\psi_k)] \quad (3)$$

$$-a\cos(\psi_k) + a\cos(\psi_{k+1}) \quad (4)$$

$$N_{k+1} = N_k + \frac{\widetilde{v_k}}{\widetilde{\dot{\psi}_k}} \cdot [\cos(\psi_k) - \cos(\psi_k + \Delta\psi_k)] \quad (5)$$

$$-a\sin(\psi_k) + a\sin(\psi_{k+1}) \quad (6)$$

As we use a front-axle-coordinate system, the axle-distance a is needed as well. We set $\widetilde{v} = v + \varepsilon_{\alpha_1}$, $\widetilde{\dot{\psi}} = \dot{\psi} + \varepsilon_{\alpha_2}$, with ε_σ being a zero-mean error variable with standard deviation σ, since the offsets are compensated. The yaw rate offset is computed during stand-still as the sum of the yaw rate measurements during stand-still shall be zero. The rotation error $\widetilde{\gamma} = \varepsilon_{\min(\alpha_3 \widetilde{\dot{\psi}}_k, \alpha_4)}$ is modeled as yaw rate dependent with zero mean and a maximum variance of α_4. α_3 is a scaling factor for the yaw rate dependency. This additional rotation is not justified by physics, but necessary as at least three noise variables are needed to sample the three-dimensional pose space degeneracy-free (see [18], p.129).

B. Likelihood calculation

Likelihood calculation for a particle is dependent on an optimal mapping of landmarks in the map $\mathcal{M} = \{m^1, \ldots, m^{m_k}\}$ to measurements: $\mathcal{Z} = \{z^1, \ldots, z^{z_k}\}$. The chosen algorithm is similar to the approach followed by Wiest et al. in [15], but adapted to the camera error model and with changed clutter handling. This optimal mapping $\theta : \{1, \ldots, m_k\} \rightarrow \{0, 1, \ldots, z_k\}$ is found by the Jonker-Volgenant algorithm (JV,[19]). A mapping to zero is hereby indicating the association to a clutter pole. JV is in our application faster than the classic approach, the Munkres algorithm [20]. Let $d(z, m)$ be distance between landmark and measurement respective to position and width of the pole:

$$\mathrm{d}(z, m) = \beta_p \cdot \mathrm{d}_M(z, m)^\top S_z^{-1} \, \mathrm{d}_M(z, m) + \frac{d_w^2}{\sigma_w^2} \quad (7)$$

The position distance is hereby a Mahalanobis distance, as the measurements have position covariances S_z from the tracking and $\mathrm{d}_M(z, m)$ is the Euclidean distance of z and m. β_p scales the weight of the position in relation to the width. d_w is the difference of z and m in width and σ_w^2 the expected pole width variance. $\mathrm{d}(z, m)$ is used as the cost function for the optimal mapping. Following the approaches in [21], [15] the likelihood now depends on the assignment θ:

- for an assigned landmark $g(z^j | m^l) = \frac{p_D}{\kappa(z^j)} exp(-\frac{1}{2}(d(z^j, m^l))$
- for a not detected landmark $g(z^0 | m^l) = 1 - p_D$.

p_D is the detection probability and $\kappa(z^j)$ the intensity of a Poisson clutter process. The weight of particle p_k^i is now defined as

$$w_k^i = \prod_{l=1}^{m_k} g(z^{\theta(l)} | m^l) \quad (8)$$

C. Resampling strategy, Exploration and State estimation

The Low Variance Sampler [18] is only applied if a degenerate particle distribution is detected. We use the number of effective particles N_{eff} as degeneracy criterion. If the short-term normalized likelihood falls in relation to the long-term filtered likelihood, active exploration is applied to increase the robustness of the filter (a variant of Augmented MCL, [18]). The normalization is done by taking the k-th root of the likelihood, if it was created by k matched poles. This reduces the influence of the map and stresses the uncertainty of the associations.

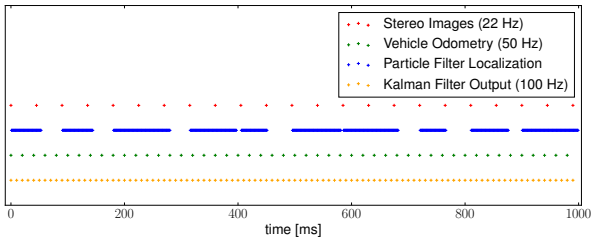


Fig. 3: Schematic diagram of measurement timings and then filter output cycle: Odometry and pose measurements arrive asynchronously and independently at the Kalman filter and are integrated as soon as they are available. The pose by the particle filter is available after the processing delay and shares the time stamp of the images, it is based on. Open loop prediction generates the output independently of the inputs.

The localization result $\mathbf{p}$ is the arithmetic mean of all particles. A full covariance matrix $\Sigma_\mathbf{p}$ is extracted as well to measure the uncertainty of the filter. More robust and accurate albeit complex approaches like a mean around the mode, clustering or Mean-Shift [22] were not applied as the localization quality is sufficient for autonomous driving.

D. Output Kalman filter

Robustness is reached by not directly using the result of the particle filter. Instead, it is fed into an extended Kalman filter together with vehicle odometry in order to provide a smooth state estimation with an update rate of 100 Hz (a requirement from the low-level controller). It uses a simple Constant Turn Rate and Velocity Model (CTRV) to model the vehicle dynamics, as more complex models are only favorable at high accelerations [23]:

$$\mathbf{x} = (x, y, \psi, v, \dot{\psi})^\top, \quad (9)$$

x and y are UTM coordinates, v vehicle speed and $\dot{\psi}$ the vehicle yaw rate. Odometry information arrives with 50 Hz and can be processed instantly. Localization results from the particle filter arrive with 22 Hz or less and have bigger delay due to the heavy processing load especially of the stereo preprocessing (Fig. 3). The Kalman filter includes all results as soon as they are available, time stamping makes sure that the filter is updated correctly, partially redoing calculations if measurements arrive too late. The output is generated independently from the sensor input by filter prediction. This approach is a big advantage compared to using a particle filter alone, as the computation latencies can be hidden.

In order to forbid sudden jumps in the vehicle pose, the particle filter measurement is controlled by a validation gate (see [24], p.236). Highly improbable measurements can be rejected. In this case the filter just includes the odometry measurements and ignores the pose input as long as the resulting normalized error is too big.

V. EXPERIMENTAL RESULTS

Six test routes in and near Berlin were chosen to represent different types of urban environments from suburbs to major streets (Table I, Fig. 4). The road classes differ as does the type of surrounding, including feature density.

TABLE I: Evaluation area properties

Name	Length of Route(s)	Type	Type of Pavement
Kleinmachnow	2.9 km	suburban neighborhood	tarmac
FUB Campus	1.1 km	small avenue	mixed
	1.4 km	small avenue	mixed
Englerallee	1.0 km	bigger avenue	tarmac
Straße des 17. Juni	3.4 km	major street, multiple moving vehicles, occlusions	tarmac
Reichstag	1.3 km	small street with pedestrians, bicycles	tarmac

A. Mapping

Figure 5 shows exemplary maps created by a single test drive without manual map editing. Tracking is used to prevent spurious detections from being included in the map. The pole density varies clearly between the different routes, with Straße des 17. Juni having less frequent landmarks at a more irregular spacing.

(a) Kleinmachnow

(b) Englerallee

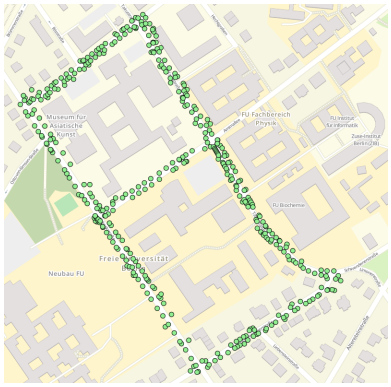
(c) Straße des 17. Juni

(d) Reichstag

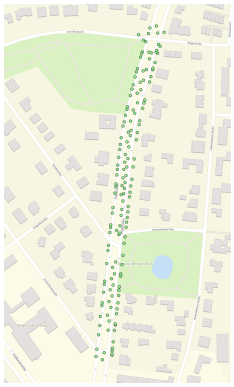
Fig. 4: Evaluation areas: example image frames

B. Repeatability Measure

While successful test runs were performed on all routes, the question of measuring the position accuracy remained open. An analysis of the results of the reference system in the test areas showed that the reported accuracies reached errors below 0.1 m only during vehicle stand-still because of multi-path effects and limited satellite visibility. Furthermore, the reference positions showed an unpredictable temporal drift which could be bigger than the reported error ranges. We therefore had to resort to calculate the lateral accuracy by a repeatability measure.



(a) Campus short - upper rectangle, Campus long - enclosing polygon



(b) Englerallee

(c) Straße des 17. Juni

(d) Straße des 17. Juni (close-up)

Fig. 5: Exemplary maps of test areas: Landmarks are shown as green circles hovering on an Open Street Map base layer (© OpenStreetMap contributors, www.openstreetmap.org/copyright)

Repeatability is tested by performing several identical laps with the test vehicle and recording the trajectory of the localization. In order to reduce the influence of the behavior and path controller, the test runs are performed by a human driver. As the tests are conducted in normal traffic, the vehicle cannot be driven identically all the time and the test driver introduces some errors as well. The average human driver varies its lateral position while driving with a standard deviation of 0.2 m within a range of 0.10 m - 0.27 m [25]. As the experiments were performed by a test driver aiming for good repeatability and the test routes were comparably short, an error of at least 0.1 m produced by the testing procedure alone seems likely. Obvious evasive maneuvers were excluded from the calculation of the aggregated values.

Table II states the repeatability for three test routes, the Monte Carlo-based algorithm was run 50 times to calculate the results. As the particle filter feeds the Kalman filter, Kalman inputs underlie a random influence, thus Kalman

results are averaged as well. All tests have a repeatability below 0.2 m. Shorter courses might ease the constraint of identical driving, resulting in a slightly better result for the Englerallee. The Kalman Filter does not change the result considerably, showing a correct filter design, as its primary goal is to add robustness and to deliver a smooth signal with a high update rate. A plot of the resulting trajectories can be seen in Fig. 8. The Applanix reference system shows a considerably higher error in a range from 0.5 m to 1.2 m. Even in very similar scenarios, the error can vary drastically, as can be seen on the Campus routes.

TABLE II: Repeatability measure results - Results for Particle Filter and Kalman Filter are the mean of 50 MC runs.

Name	PF	KF	Applanix	Laps
FUB Campus Short	0.177 m	0.175 m	1.162 m	7
FUB Campus Long	0.186 m	0.188 m	0.737 m	6
Englerallee	0.140 m	0.139 m	0.507 m	8

C. Resource Usage

In the field tests, a processing rate of 13.5 Hz was reached on average. Table III shows that the stereo preprocessing dominates the runtime consumption. rSGM was performed completely on the CPU, taking a big share of the processing load. The localization modules only need 20% of the overall runtime, therefore increasing the load only slightly in a system already using stereo data for other purposes such as object detection. As the system relies on a KD-tree to store the poles in the map, the approach is scalable to larger scenarios.

TABLE III: Processing times of the full localization solution on a mobile Intel i7-4960HQ CPU laptop

rSGM Stereo Matching	56 ms
Landmark Detection & Tracking	8 ms
Particle Filter	5 ms
Output Kalman Filter	$<$ 1 ms
Sum	$\sim$70 ms

The average latency for the pose estimation is 110 ms.

D. Number of Re-Initializations

The particle filter can get lost without a realistic probability to recover on its own. A threshold on the position uncertainty of the filter is used as indicator. If the geometric mean of the position standard deviation is above 15 m, the filter is treated as lost and a re-initialization using GPS is triggered. These re-initializations can be counted and used to compare parameter settings with regard to robustness.

E. Influences on accuracy

The impact of a reduced frame rate was tested on a trajectory of several laps recorded at the test route Campus short. The full frame rate of 22 Hz was reduced to 11 Hz leading to only a slight but steady increase in the repeatability error. 50 Monte Carlo runs were performed with each setting. Fig. 6 shows as well that reliability suffers at lower frame rates, at 11 Hz one re-initialization happened.

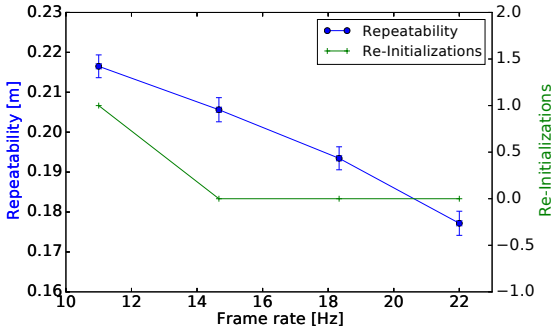


Fig. 6: Localization quality in relation to processing frame rate. As expected, the repeatability degrades with lower frame rates and the probability to get lost increases.

The number of particles is usually a key parameter for a particle filter, as its runtime scales approximately linearly with this number (if the likelihood evaluation dominates the processing, as it is in our case). Varying the number from 50 to 2000 (Fig. 7) shows that repeatability is nearly constant until the number of particles is reduced to 500, while the number of re-initializations is only constant till 1000, indicating reduced robustness from thereon. 1000 particles seem to be a good compromise between runtime consumption and reliability.

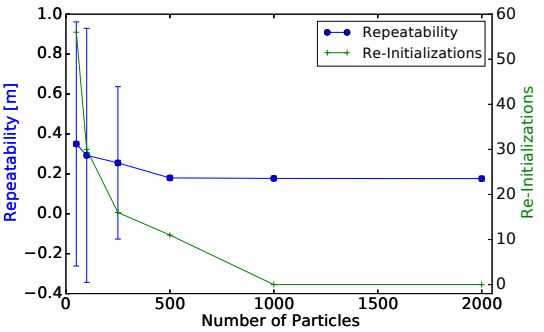


Fig. 7: Varying the number of particles - lowering the number of particles first reduces robustness, then lateral accuracy.

VI. CONCLUSIONS

We demonstrated that pole-like structures can serve as reliable landmarks in urban scenarios for autonomous driving. The realized accuracies were sufficient in various scenarios and allowed for smooth autonomous control behavior. Increasing the availability of the localization to a comprehensive approach by the integration of further landmark classes is a future goal. Lane and pavement markers or other distinct three-dimensional landmarks as guard-rails, walls or bigger structures as tunnels close the gap in areas without poles. The integration of the map building into a large-scale SLAM method is another extension, which is directly related to the question of efficient map updates to keep them accurate.

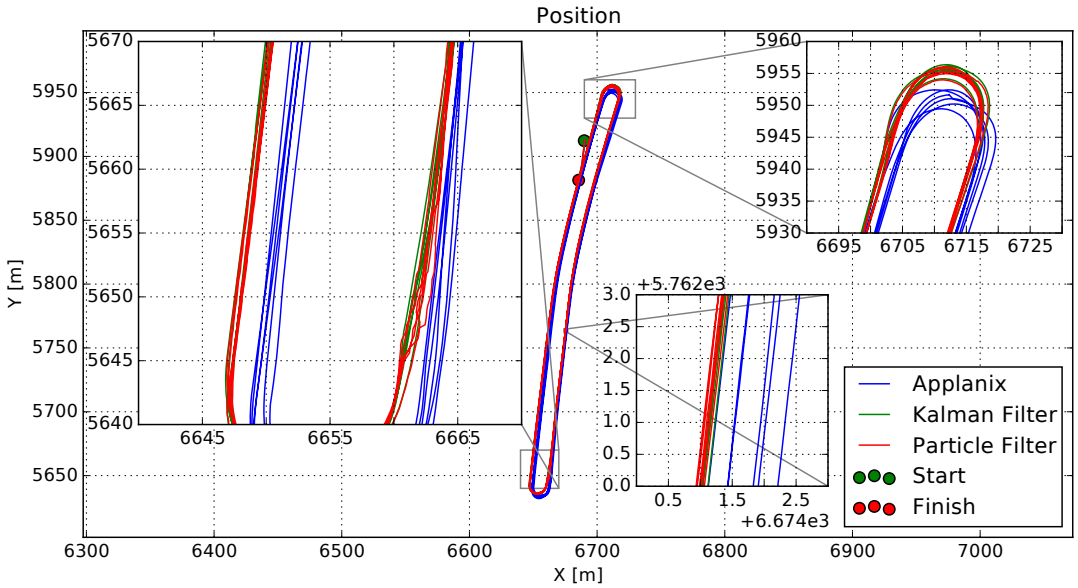


Fig. 8: Example trajectory at Englerallee, arbitrary UTM coordinates: Trajectory parts at the U-turns were excluded, as the driver had to evade other vehicles several times. Repeatability of the particle filter is good most of the time, after the south U-turn the particle filter shows a slight jitter (left in left zoomed area). The Kalman filter stays on a smooth course. The Applanix trajectories show a significant drift over time, resulting in a bad repeatability.

References

[1] A. Geiger, M. Lauer, F. Moosmann, B. Ranft, H. Rapp, C. Stiller, and J. Ziegler, "Team annieway's entry to the 2011 grand cooperative driving challenge," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, no. 3, pp. 1008–1017, Sept 2012.

[2] J. Carlson, "Mapping large, urban environments with gps-aided slam," Ph.D. dissertation, Carnegie Mellon University, 2010.

[3] H. Dodel and D. Häupler, *Satellitennavigation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[4] J. Levinson and S. Thrun, "Robust vehicle localization in urban environments using probabilistic maps," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, May 2010, pp. 4372–4378.

[5] E. Olson, "Real-time correlative scan matching," in *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, 2009, pp. 4387–4393.

[6] C. Brenner, "Global localization of vehicles using local pole patterns," in *Pattern Recognition*, ser. Lecture Notes in Computer Science, J. Denzler, G. Notni, and H. Süße, Eds. Springer Berlin Heidelberg, 2009, vol. 5748, pp. 61–70.

[7] A. Schlichting and C. Brenner, "Localization using automotive laser scanners and local pattern matching," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, June 2014, pp. 414–419.

[8] O. Pink, "Bildbasierte selbstlokalisierung von straßenfahrzeugen," Ph.D. dissertation, Institut für Mess- und Regelungstechnik mit Maschinenlaboratorium (MRT) , KIT, 2011.

[9] A. Schindler, "Vehicle self-localization using high-precision digital maps," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*, June 2013, pp. 141–146.

[10] M. Lundgren, E. Stenborg, L. Svensson, and L. Hammarstrand, "Vehicle self-localization using off-the-shelf sensors and a detailed map," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, June 2014, pp. 522–528.

[11] J. Ziegler, H. Lategahn, M. Schreiber, C. Keller, C. Knoppel, J. Hipp, M. Haueis, and C. Stiller, "Video based localization for bertha," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, June 2014, pp. 1231–1238.

[12] D. Gruyer, R. Belaroussi, and M. Revilloud, "Map-aided localization with lateral perception," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, June 2014, pp. 674–680.

[13] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image Vision Comput.*, vol. 22, no. 10, pp. 761–767, 2004.

[14] M. Stubler, J. Wiest, and K. Dietmayer, "Feature-based mapping and self-localization for road vehicles using a single grayscale camera," in *Intelligent Vehicles Symposium (IV), 2015 IEEE*, June 2015, pp. 267–272.

[15] J. Wiest, H. Deutsch, D. Nuss, S. Reuter, M. Fritzsche, and K. Dietmayer, "Multi-sensor self-localization based on maximally stable extremal regions," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, June 2014, pp. 555–560.

[16] R. Spangenberg, T. Langner, S. Adfeldt, and R. Rojas, "Large scale semi-global matching on the cpu," in *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, June 2014, pp. 195–201.

[17] A. F. Bobick and S. S. Intille, "Large occlusion stereo," *International Journal of Computer Vision*, vol. 33, no. 3, pp. 181–200, 1999.

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

[19] R. Jonker and A. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing*, vol. 38, no. 4, pp. 325–340, 1987.

[20] J. Munkres, "Algorithms for the assignment and transportation problems," *Journal of the Society of Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 32–38, March 1957.

[21] B.-T. Vo and B.-N. Vo, "Labeled random finite sets and multi-object conjugate priors," *Signal Processing, IEEE Transactions on*, vol. 61, no. 13, pp. 3460–3475, July 2013.

[22] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995. [Online]. Available: http://dx.doi.org/10.1109/34.400568

[23] R. Schubert, E. Richter, and G. Wanielik, "Comparison and evaluation of advanced motion models for vehicle tracking," in *Information Fusion, 2008 11th International Conference on*, June 2008, pp. 1–6.

[24] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: John Wiley And Sons, Inc., 2002.

[25] J. Zhou, H. Peng, and T. Gordon, "Characterization of the lateral control performance by human drivers on highways." *SAE Int. J. Passeng. Cars - Mech. Syst.*, vol. 1, no. 1, pp. 450–458, 4 2009.