

Precise Localization in High-Definition Road Maps for Urban Regions

Fabian Poggenhans¹, Niels Ole Salscheider¹ and Christoph Stiller²

Abstract—The future of automated driving in urban areas will most probably rely on highly accurate road maps. However, the necessary precision of a localization in such maps has so far only been reached using extra, sensor specific feature layers for localization. In this paper we want to show that it is possible to achieve sufficient accuracy without a separate localization layer. Instead, elements are used that are already contained in high-resolution road maps, such as markings and road borders. For this, we introduce a modular approach in which detections from different detection algorithms are associated with elements in the map and then fused to an absolute pose using an Unscented Kalman Filter. We evaluate our approach using a sensor setup that employs a stereo camera, vehicle odometry and a low-cost GNSS module on a 5 km test route covering both narrow urban roads and multi-lane main roads under varying weather conditions. The results show that this approach is capable to be used for highly automated driving, showing an accuracy of 0.08 m in typical road scenarios and a is available 98% of the time.

I. INTRODUCTION

The use of high-definition maps is of unbroken importance for highly automated driving. Especially in urban environments, such maps are indispensable as they need to provide information that is very hard to infer during driving because of occlusion, limited sensor range, because they are computationally expensive or simply because the available algorithms are not sufficient. Maps must therefore provide the location of road borders, lane markings and neighbouring lanes for navigation, valuable information for pedestrian prediction and include traffic rules such as traffic lights or speed limits for each individual lane. Such maps (often referred to as high definition maps) are to a small degree already available from commercial map providers, such as [1] or [2].

With a higher level of detail of the maps, the demands on localization accuracy increase as well. While localization only by means of a GNSS receiver is possible for simple navigation maps, this is no longer sufficient for lane-level accurate maps. Firstly, because the accuracy of using only GNSS is insufficient [3], and secondly, the geo-annotation of the maps themselves is inaccurate. To solve this problem, many algorithms for many sensors have been proposed and shown their value in practical usage (e.g. [4]–[6]). These approaches originate from classical simultaneous localization and mapping (SLAM) approaches in robotics, where an agent is supposed to localize in an unknown indoor environment.



Fig. 1: Map projected into camera image based on current pose determined by our algorithm

Even if these approaches can achieve high precision within a few centimetres [7], these algorithms have disadvantageous properties when used in automated vehicles. Because the maps are generated with the same sensor that is used for localization, they are inherently sensor-bound. This is very unfavourable for persistent road maps, which need to be usable for as many current and future vehicles as possible. In addition, the features are part of a separate map layer which needs to exactly coincide with the other map layers. The features are also not maintainable or comprehensible by humans, which makes subsequent correction of the map much more difficult. This is all the more problematic because these maps are known for rapid aging. Changes in vegetation due to seasonal changes may require a map update every few months in vegetation-rich sections.

In contrast to SLAM-based approaches, a second class of approaches exists where structures are used that have a semantic meaning for driving such as road markings and guardrails (often referred to as road features). The advantage of these structures is that, since they have a direct significance for road traffic and are therefore very concise, they can be detected comparatively easily and by various sensors. The functionality to detect them is often already part of today’s production vehicles. Since these elements are also used for high-definition maps, there is no need for a further layer in the map. Instead, the localization information can be derived directly from the map, which makes this approach much more suitable for highly automated driving than SLAM approaches. And because these features are very persistent, such maps only need to be updated if structural changes have been made, which requires an update of the map anyway.

Most of such approaches rely on the detection of road markings (e.g. [8]–[12]). Compared to SLAM approaches, these are characterized by a lower – but for most applications still tolerable – accuracy of typically about 0.2 m in lateral and 0.5 m in longitudinal direction on main roads. The reduced accuracy is explained by the lower density of the features used. The maps are mostly created by hand, so that the maps itself are inaccurate as well. The higher

¹FZI Research Center for Information Technology, 76131 Karlsruhe, Germany {poggenhans, salscheider}@fzi.de

²Institute of Measurement and Control Systems, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany stiller@kit.edu

error in the longitudinal direction can be explained by the fact that markings are usually extended in the direction of travel, which makes the position in longitudinal direction more difficult to observe. The mentioned approaches also do not distinguish between different types of road markings. Instead, a parametric model is often fitted to the detections in the camera image. This means that e.g. for individual dashes, the exact position of each dash is lost. Furthermore, when few or no road markings are available (as on most small roads in European cities), the performance of these approaches quickly decreases so that they become unusable. Therefore, some approaches combine the road marking detection with more features to increase robustness. As an example, for the Bertha-Benz drive [13], curbstones were used in addition for localization [14]. As a result, it was possible to achieve satisfying accuracy even off the main roads.

Still, there are many situations left, where curbstones are hard to detect, not present or occluded by parking vehicles. For these situations, the approaches mentioned above are not generic enough because they have either been developed around a particular detector or for a specific scenario. At the same time, modern maps can provide significantly more information than just the position of lane boundaries, such as guardrails, traffic lights, traffic signs – and more information might be added in the future. Similar progress can be observed in the detection of such structures. Developments in the field of semantic labeling with artificial neural networks have made it possible to detect more and more relevant objects in road traffic. A localization approach must therefore be modular and generic in terms of the primitives used for localization to keep pace with these developments instead of using hand-selected features.

With this paper, we aim to present such a modular localization procedure that is built around using information already present in high-definition road maps. It is designed to provide sufficient accuracy, possibly even on small urban roads. If our approach is combined with a SLAM approach to obtain accurate vehicle positions for mapping, we can also demonstrate that it is possible to automate the process of map generation to avoid the lower accuracy of manually created maps and to speed up the map generation procedure.

II. OUR APPROACH

A. Association

Compared to the information contained in map layers for SLAM approaches, the information provided by road maps for localization has some properties that make the design of localization algorithms more difficult. The main difference is that unique descriptors are used in SLAM-based approaches so that the association of detected and mapped elements can be made directly. In addition, point features are usually used, so that each feature can always be observed unambiguously. However, this is not the case for the elements of a road map. Instead, a map can contain an infinite number of elements of the same type, and the elements can have any shape, not just a point. Therefore, such an approach always requires an intermediate step in

which associations are formed between the detection and elements of the map. We call this step *Map Matching*. To limit the search radius in the map, it is advisable to use a GNSS receiver. This effectively reduces the search radius to a few meters, depending on the accuracy of the receiver and the method used for evaluating the signals. Nevertheless, ambiguities cannot be resolved completely because, for example, markings from adjacent lanes have a distance of about 3 m to each other [15], which is less than the accuracy provided by low-cost GNSS receivers. In literature, a nearest-neighbour association relative to the currently estimated pose is often used as a solution (e.g. [12], [14], [16]). It is obvious that mistakes in the association are impossible to correct. Furthermore this approach ignores the dynamic properties of a vehicle, because it is not checked if the estimated position is even reachable from the last known position given the current vehicle motion. Other publications (e.g. [11]) therefore use a particle-based approach, which allows several possible positions to be tracked and compared with the observed detections. However, particle-based approaches have the disadvantage that an oscillation of the most likely position between several equal positions can occur, which is highly undesirable for automated driving. We therefore suggest that ambiguities in the determination of the position should instead be passed on to a separate *fusion* step in which a selection can be made based on the history of vehicle poses and current movement reported by odometry sensors.

B. Representation of detections

Since elements in the map do not only consist of points, a representation form must be chosen which can be used for all primitives (and thus also for all detectors). This is also important for the extensibility of the localization system. Structures along the map are generally flat and long, so that the representation by polylines (or an individual points as a special case of a very short line) offers itself. Nearly all three-dimensional structures along the road are vertical (such as poles, walls and buildings), so that they can be completely described by their outline by a polyline. In the literature splines are also used as an alternative, especially on highways, because they are characterised by a more compact representation. However, it is difficult to describe corners and sharp curves that dominate in urban areas.

C. Fusion

As mentioned above, a fusion step is required in which the vehicle position is estimated from the chronological order of observations of elements from the map and the motion state of the vehicle. In SLAM-based approaches (e.g. [4], [7]), a bundle adjustment procedure is used for this. Here, the vehicle position is determined, which optimally explains all point-shaped landmark observations. In principle, this adjustment can be made over the whole temporal sequence of observations. Since each individual landmark is thus taken into account when determining the position, a high degree of accuracy can be achieved. This is not common in road feature-based approaches, however, because the formulation

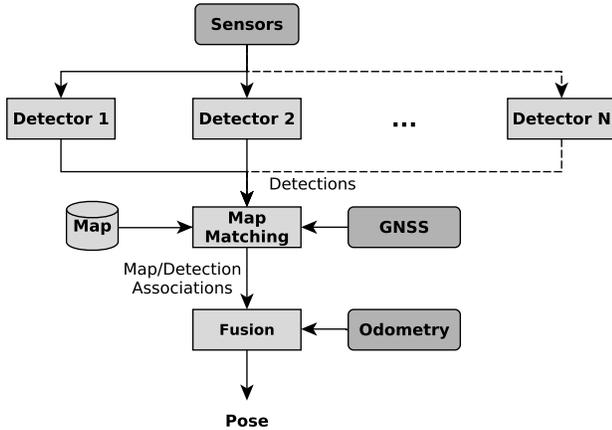


Fig. 2: Structure of the localization procedure

of a line-based bundle adjustment problem is characterized by significantly higher non-linearities and thus a worse convergence behavior. Instead, probabilistic methods such as Kalman filters are used here (see [8], [14]). The observed features are here combined into a single normal distribution around an estimated position. The influence of the individual observed lines on localization is lost. This is particularly unfavourable because the probability distribution of possible positions for an observed line is generally also line-shaped, which cannot be expressed by a normal distribution. In this paper we therefore present a fusion approach based on an Unscented Kalman Filter (UKF), in whose measurement model the influence of the observed elements can be considered individually. The filter is continuously updated by high-frequency odometry and steering angle measurements from inside the vehicle to keep the prediction up to date. We do not recommend to use the GNSS as additional measurement in the filter, because they are often offset relative to the map.

D. Structure

A generalized structure of the proposed localization procedure described in the previous lines is shown in Fig. 2. For each type of element in the map, a separate detector can be used, which is specialized in the detection of this element. In this paper we will limit ourselves to the detection of markings and road borders, more and different detectors would be conceivable here. The detections are then passed to a map matching step, where associations between detection and the closest element in the map are created using GNSS and the last known pose as initialization. These associations are then passed to the final fusion step, where the actual pose is determined.

III. ALGORITHM DETAILS

The details of each processing step are explained in the following sections.

A. Road Marking Detection

Road marking detection on accumulated topview images instead of the original images. Topview images have the

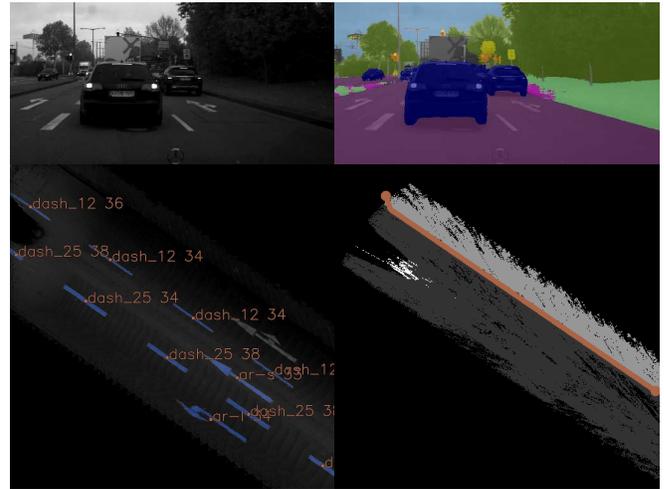


Fig. 3: Processing steps of the detection algorithms (top left to bottom right): Original image, semantic labelling (road pixels in violet), marking detection is performed on accumulated topview (detected markings in blue), road border detections on accumulated semantic topview (detections in red).

advantage of providing images which are sensor and perspective independent and the accumulation over multiple time steps increases robustness and detection range. Accumulated topviews are created by mapping a stereo pointcloud [17] with intensity values from the image to a two-dimensional grid map that will then be interpreted as the topview image. Points in the pointcloud that are too far away from the ground plane are discarded to avoid that trees or bridges are mapped into the grid map. Between two pointclouds from different time steps, the gridmap is shifted by the vehicle motion reported by the odometry sensors. The actual detection of road markings is performed by the detection approach introduced in [18]. Detected markings are characterized by their centerline (i.e. a polyline) and a probability distribution over the most likely classes. Results are shown in Fig. 3 (bottom left), with the marking candidates drawn in dark blue and the most probable classifications as text on each marking. The topview image was created using a resolution of 50 px/m at a 20 m×20 m window.

B. Road Border Detection

For road border detection, Semantic Labelling is used so that each pixel in the input image is assigned to a class. The classification is done by a Convolutional Neural Network (CNN) based on ResNet-38 [19]. This neural network is trained on the Cityscapes dataset [20] that contains 5000 images with fine annotations and 20000 images with coarse annotations. The network distinguishes between 19 different labels. An example is shown in Fig. 3 top right. The image includes the following labels: road (violet), vegetation (light green), trees (dark green), vehicles (blue), sidewalk (magenta) and sky (turquoise). However, only a small portion of these labels are used for the recognition of the road border:

We define road border as the transition from road to any other label. This also includes vehicles, as the road is often bounded by parked vehicles (see Fig. 5 bottom for examples).

Similar to the road marking detection described above, the labelled image is transformed to an accumulated topview. As a result, the image only contains three classes: Unknown (black), road (dark grey) and non-road (light grey). From here, road borders can be easily detected by finding edges between road and non-road regions. We solved this by applying morphological closing to the image for reducing noise and filling gaps, then using Canny edge detection [21] on road to non-road edges to detect the road borders and finally applying a post-processing step to reject too short edges and simplify the shape as a polyline.

C. Map Matching

The map matching step serves to find sets of associations between elements in the map and detections reported by the individual detectors. Compared to the mentioned SLAM-based approaches, additional challenges must be considered:

Global ambiguities: The features in the map are not unique. Even with a wide detection range around the vehicle, there are still endless possible poses in a map. To reduce the search area the algorithm needs one or multiple search areas in the map. Because the algorithm is allowed to report multiple sets of associations, every search area can be considered separately. The associations are then collected and passed on to the next step. We use the GNSS signal and the predicted pose from the fusion algorithm (if available) in parallel to find valid associations. Thus, the algorithm does not have to distinguish between an initialization phase (use of GNSS) and normal operation (use of prediction).

Local ambiguities: Even if the pose is known to within a few meters, many ambiguities arise. For example, with a search radius of 5 m, up to four associations are possible for the detection of one of the dashed lines as shown in Fig. 4: The true marking, the respective predecessor or successor of the detection and the continuous lines on the left and right side of the road. We can partly eliminate these ambiguities by considering the line type. Those ambiguities that cannot be resolved will be passed on to the fusion.

Inaccuracies in classification: Detectors often report a likelihood or probability distribution for a detection or classification result. The marking detector used by us for example reports a probability distribution for the type of each detection (including dashed lines, continuous lines, stop lines, pedestrian crossings, different types of arrows, etc.). That is why we consider not only the best classification result, but also the alternatives if they are classified with a considerably high probability.

Partial occlusion and limited sensor range: Many elements in the map – especially continuous lines and road borders – can be many hundreds of meters long. Because of their length, these lines are important to determine the lateral position in the lane and the vehicle orientation. However, the position along the line cannot be determined because the endpoints are often unobservable due to the limited sensor

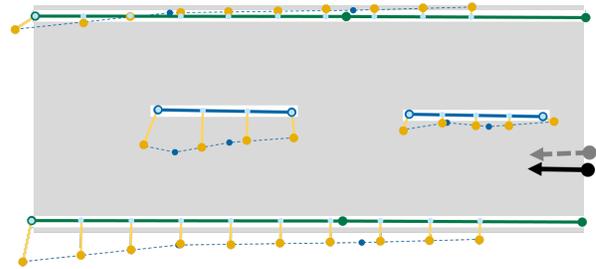


Fig. 4: Schematic representation of the fusion step. Dashed polylines in the map are shown as solid blue lines, continuous lines as solid green lines. Detections are dashed lines of the same color. Subsamped points on these lines are shown together with a connecting line (yellow) to their respective closest point on the element in the map (light blue).

range or occlusion. Therefore, we distinguish between types where the endpoints of a detection are usually observable (e.g. dashed line markings or arrows) and those where they are not observable (e. g. continuous lines and roadsides).

Our proposed solution to these problems consists of three steps: First, to eliminate global ambiguities, we reduce the search space to the elements in the map that are closer than a distance d_{\max} to a prior (either GNSS or predicted pose), where d_{\max} is the sum of the search radius plus the sensors maximum viewing range. Secondly, to reduce local ambiguities, we search for pose hypotheses that explain as many detections as possible with matching elements from the map. Lastly, for each such pose, we calculate the number of inliers as the number of points equally spaced along each detected line that are closer to the matching element than a certain threshold. This inlier rating serves as a measure to select the best sets of matches to be passed on to the fusion while the other, less significant sets are discarded.

The process of generating the pose hypotheses could be described as continuous hough transform [22] in continuous space. We generate votes for an individual position by forming pairs of detections and an element in the map of the same type and computing possible vehicle poses in the map where such a detection could be observed. Similar votes (i.e. poses that are closer than a threshold) are then grouped together. A group with a large number of poses then indicates that a large number of detections support this position. If the type of a detection is not clear because – as mentioned above – the classification result is given as a probability distribution, pairs are generated if one of the classification results matches.

D. Fusion

Our proposed fusion approach is based on an Unscented Kalman Filter (UKF) [23] used during the Bertha Benz Drive [24]. The vehicle state is represented using a dynamic single track model and continuously updated with on-board vehicle odometry data composed of current speed, yaw rate and steering angle.

To filter matches between detections and elements in the map, a new observation model is introduced. Fig. 4 shows a

simplified example. For every set of matches from the map matching step, a number of associations between detections and elements in the map is given. For this processing step, these associations remain unchanged. Instead, any set of associations is interpreted as a possible observation, and then, from all of these sets, the one that best fits to the filter state is selected. In this point, our method differs from other methods such as Iterative Closest Point (ICP). This is because the associations given a suspected pose are very clear (road markings are usually more than 2 m away from each other) and also because better convergence can be achieved if the assignments do not change during the fusion step.

Furthermore, we formulate the problem not on a point-to-point basis (P2P), as is the case with ICP, but on a point-to-polyline basis (P2L), where the point is a point on the detected polyline and the polyline is the matched element from the map. The error is therefore the distance to the closest position on the polyline. This eliminates the need to subsample the elements from the map as well, which increases accuracy and improves convergence. This formulation does not punish if only some parts of a longer line could be observed, since only the distance of a detection point to the polyline in the map is calculated, not vice versa. To achieve an even weighting, we sub-sample the detections with equal spacing. Fig. 4 shows an example of this formulation. Every yellow point is one subsampled point on a detection.

An observation model for an Unscented Kalman Filter consists of three things: An observation vector \vec{z} , a function $h(\vec{x})$, which transforms the state \vec{x} or a sigma point χ into the domain of the observation, and a covariance matrix Q for the observation. In our case, the sub-sampled points of the detection correspond to the observation, i.e. $\vec{z} = [p_1, p_2, \dots, p_n]$, where n is the number of points. To define the function $\vec{z} = h(\vec{x})$, we proceed as follows: For each point p_n in \vec{z} , the corresponding polyline of the map is transformed into the vehicle coordinate system using the pose, which is part of \vec{x} . Now the point $\vec{p}_{next,i}$ on the polyline closest to the detected point is determined (this corresponds to the light blue points in Fig. 4). Thus $\vec{z} = [\vec{p}_{next,1}, \vec{p}_{next,2}, \dots, \vec{p}_{next,n}]$. Finally, the covariance matrix Q indicates the uncertainty of the individual points observed. We assume an equal covariance matrix for all points, but more complex formulations (for example, based on the distance to the vehicle) would be possible. The described formulation also applies to the case where an element is just described by a single point, because this is just a special case of a very short polyline.

To select the most suitable set of associations from the associations generated in the map matching step, we can make use of the state of the UKF itself to find the likelihood of an observation: When filtering n new sets of matches, for each observation \vec{z}_k with $k = 1 \dots n$ and dimension d_k , we compute the innovation covariance S_k and the predicted observation $\hat{\vec{z}}$ of the UKF. These properties represent the state of the filter as well as its covariance in the domain of the observation. From this, we compute the innovation residual

$\vec{r}_k = \vec{z}_k - \hat{\vec{z}}_k$ and obtain the observation likelihood p_k as:

$$p_k = \frac{1}{\sqrt{(2\pi)^{d_k} \det(S_k)}} \exp\left(-\frac{1}{2} \vec{r}_k^T S_k^{-1} \vec{r}_k\right).$$

We then select the pose with the highest observation likelihood, update the filter with it and discard the others.

IV. MAP GENERATION

As mentioned in Section I, our method is intended to be applied to existing high-definition maps with lane-level accuracy. However, it is comparatively easy to use the methods described above to generate a map for localization from a previous drive. A precondition is that geo-referenced vehicle positions are known with great accuracy. Since the method described here is not suitable as a SLAM method (the accuracy would not be sufficient for mapping), we use a different method for this, which we describe in Section IV-A. Once the vehicle positions are known, we can combine them with the results from the detectors during the drive to determine geo-referenced positions of all detections. To filter out multiple detections, fragmented markings and false detections, we use a procedure described in Section IV-B.

A disadvantage of automatic mapping is that not all elements of the map can be detected. This is partly due to errors by the detectors, but also because elements are occluded in the mapping drive. One example for this are other road users, but especially parked vehicles that occlude the road border (see Fig. 5 bottom). The map is therefore designed to be editable by humans so that missing (or wrong) elements can be corrected. For that, we used the OpenStreetMap (OSM) file format that is already used for mapping and has publicly available editors.

A. Pose Bundle Adjustment

Determining the poses from the driven route is the most important part for the mapping as it directly impacts map accuracy. A complex, exact geo-annotation is however not necessary, because GNSS is only used as a hint for the required search area. Therefore it must only be ensured that the search area is large enough so that the correct position is guaranteed to be covered. It is however important that the map is consistent and smooth. For that we used an algorithm that generates a pose-graph from the observation of visual landmarks and optimises them in a bundle adjustment problem [25]. We extended the bundle adjustment problem by adding the GNSS measurements obtained during the mapping drive as constraints to obtain geo-referenced poses.

B. Grouping of Detections and Outlier Rejection

The grouping step serves to distinguish correct detections from false detections, multiple detections or misclassifications. This is done by making use of the relations between the individual detections to validate them with each other. The algorithm used is described in [26]. It validates detections by testing if they can be clustered with other detections in the proximity to form continuous lines (e.g. from individual dashes) or clustered multiple detections from the element

type on the same position. If this is not possible, they are rejected.

V. EVALUATION

For evaluating the performance of our approach, we reuse the mapping poses from the landmark-based posegraph described in Section IV-A as a reference. Because the whole map is based on these poses, an optimal localization would be able to exactly reproduce them. High-precision GNSS sensors, which are often used as a reference at this point, have been found to be unsuitable for urban environments with a provided accuracy of more than 20 cm. We therefore used the data from the mapping drive again for localization with our approach and compared the poses obtained to the poses used to create the map. This gives us the opportunity to test the maximum performance of our approach under the most favourable conditions because the detections should fit particularly well to the elements of the map, whereas the influence of the detectors on the localization result is reduced. Under these conditions, we can examine the effects of certain aspects of our approach (considering different types of markings (lines, dashed lines, arrows, stoplines, etc.) for matching, multiple hypotheses, using road border as additional feature) on the localization result.

In order to assess the quality, besides the average error in the x-direction (longitudinal), y-direction (lateral) and the yaw angle we also use the availability and the reliability as quality measures. We define availability as the proportion of the total time, in which two global updates by our approach are not more than one second apart. Reliability is the proportion of the total time, in which the localization error is smaller than 0.5 m. Vehicles typically have a width of 2 m while the width of a typical lane is 3 m. Therefore, we chose 0.5 m as a just tolerable error.

To ensure that localization quality is also satisfactory in other drives, we calculate the lateral distance of the poses from other drives to the trajectory of the mapping drive. For that, we recorded the same route under different weather conditions (sunny, cloudy, rain) to test the robustness of the approach. Here, a higher deviation is to be expected because, of course, the trajectories of the test drive and the other drives are not identical. However, the expected deviations should be at around 10 cm, except at locations where lane changes were made.

A. Data Set

The data was acquired in four drives of the same route in dense urban area in Karlsruhe, Germany of approximately 5 km length under different weather conditions. The route was selected because it featured a lot of different road types from narrow single-lane roads to multi-lane major roads under varying road conditions and included many intersections. Fig. 5 shows some images from the dataset. For the evaluation, we divided the drive into two sections: Narrow and normal roads. Narrow roads are areas with virtually no road markings, so that the localization is only based on road border detections, which is very challenging.



Fig. 5: Scenes from our data set with the map projected into the image based on the estimated pose.



Fig. 6: Weather comparison during measurement drives.



Fig. 7: Results from the mapping process. Detected dashed markings are drawn in blue, road border detections in red.

We have included these areas in our dataset because they are very common in inner cities, but have been almost completely ignored in existing literature. In our data set, the proportion of the total distance is about 20%.

The data was recorded with two BlackFly PGE-50S5M cameras for stereo vision at 5 Mpx and 8 Hz, Ublox C94-M8P receiver for the GNSS data and the on-board vehicle odometry by the research vehicle *BerthaONE* [27].

We also evaluated the average online processing times of our implementation using two 2.6 GHz Intel Xeon CPUs, 64 GB memory and a Nvidia Geforce Titan X for semantic labelling and stereo processing. The total cycle time for one image is 255 ms. However, as the whole process chain is pipelined, it is able to process data in the rate of the slowest element (semantic labelling with 120 ms per image) so that the average processing rate is 8 Hz. Map matching processing time was 5 ms and the UKF required 15 ms.

B. Evaluation of Map Generation

Some example images of the map are shown in Fig. 7. It can be seen that almost all line markings on the road were detected, most of the arrows on the ego lane and some of

TABLE I: Localization performance compared to mapping poses: Availability, average error in x (longitudinal), y (lateral) and yaw angle and reliability when using the full approach compared to when using only the first hypothesis, when not using road border, and when not considering any classes for matching.

Scenario		Availability [%]	Avg. e_x [m]	Avg. e_y [m]	Avg. e_ϕ [°]	Reliability [%]
Normal	Full	97.7	0.19	0.08	2.17	97.1
	No multi hypotheses	97.7	0.21	0.09	2.11	96.7
	No road border	90.48	0.17	0.08	1.78	98.9
	No classes	95.57	1.54	0.16	4.36	89.1
Narrow	Full	53.5	0.58	0.37	1.71	75.4
	No multi hypotheses	60.7	0.71	0.34	1.45	78.0
	No road border	38	0.5	0.46	3.03	60.7
	No classes	64	1.6	0.49	13.5	50.7
Overall	Full	85.24	0.26	0.14	2.10	93.4
	No multi hypotheses	87.27	0.35	0.16	1.93	91.5
	No road border	81.77	0.26	0.18	2.12	88.5
	No classes	86.05	1.57	0.25	6.36	82.7

the arrows from the neighbouring lane. In the normal road regions, almost all of the road border was correctly detected, however most of the road border detections in the narrow regions of the drive had to be manually corrected in the map because long rows of parking vehicles occluded the road border. It is also visible, that the satellite images drawn in the background do not perfectly align with our map. This is partly because satellite data is often not correctly aligned but also because the geo-annotation of our map is only accurate within 3 m due to GNSS receiver inaccuracy.

C. Evaluation of Localization Performance

Table I summarizes the collected results. It is noticeable that the lateral accuracy in narrow streets with an average error of 0.37m differs very much from the accuracy on normal roads with an average error of 0.08m. Obviously, localizing alone with road border in the presence of parked vehicles is not accurate enough. Nevertheless, the accuracy is considerably higher compared to the scenario where road borders are not used at all (0.46 m). The same is shown by the availability measure. This is 53% for narrow roads in the full case and drops to 38% if no road border detection is used. The same holds true for normal roads, where availability is 97.7% in the full case and falls to 90.5% without road border detection. The use of the road borders for localization therefore increases the availability, but not necessarily the accuracy. On the one hand, this has to do with the fact that road border detection in our dataset was often influenced by parked vehicles, on the other hand with the fact that for the detection of road borders images at a resolution of 512x256 pixels are used for performance reasons. Therefore the detection of road borders is very coarse.

Considering several hypotheses for localization did not bring any noticeable improvements. If only the first hypothesis (the one with the most inliers) is taken into account, the average accuracy decreases by only 0.02 m. The reason for this is probably that road markings are usually at least 3 m away from each other. The localization uncertainty must therefore be at least 1.5 m, so that mismatches are possible. This uncertainty was never reached during the entire test drive. We therefore conclude that the consideration of several hypotheses improves the localization result only during the

initialization, or re-initialization after an error. If the vehicle position is already known with sufficient accuracy, the effect is negligible.

In contrast, the consideration of the different types of markings in the matching step was very important. Without this, the localization error in y direction increases by more than 1 m. This was mainly caused by small deviations in the estimated orientation that caused dashed lines to be assigned to straight lines at the side of the road. If this causes the vehicle to be mistakenly matched to the adjacent lane, correction of that error is no longer possible without knowledge of the road marking types.

D. Evaluation of Performance in Different Drives

The comparison of the results with other test drives (see Fig. 8) generally shows very similar results to those of the mapping drive. It should be mentioned again that no exact results can be given, because only the distance to the trajectory of the localization drive was calculated. When comparing the different weather scenarios, it is noticeable that sun and shadows casted by it had no effect on the localization result. Average deviation from the reference trajectory on normal roads is 0.20 m in cloudy weather and 0.18 m during sunshine. Thus, the difference to the measurement drive with 0.08 m is about 0.1 m which seems plausible. During rain, however, the deviation increases to 0.28 m. This can be explained by the lower detection rate of the road markings: The wet road reflects so bright that markings can no longer be properly segmented. Moreover, the windshield wiper in the image causes artifacts.

VI. CONCLUSION AND FUTURE WORK

The results show that a localization method based solely on high definition map data, is accurate enough under typical circumstances to allow highly automated driving. This is demonstrated by the availability of 97.7%, the reliability of 97.1% on normal roads and the insensitivity to weather conditions. The integration effort is low, since most detectors are already part of existing and near future series vehicles and the additional computational effort for the matching and fusion steps is very low. Our algorithm significantly reduces the mapping effort because existing detectors can be used

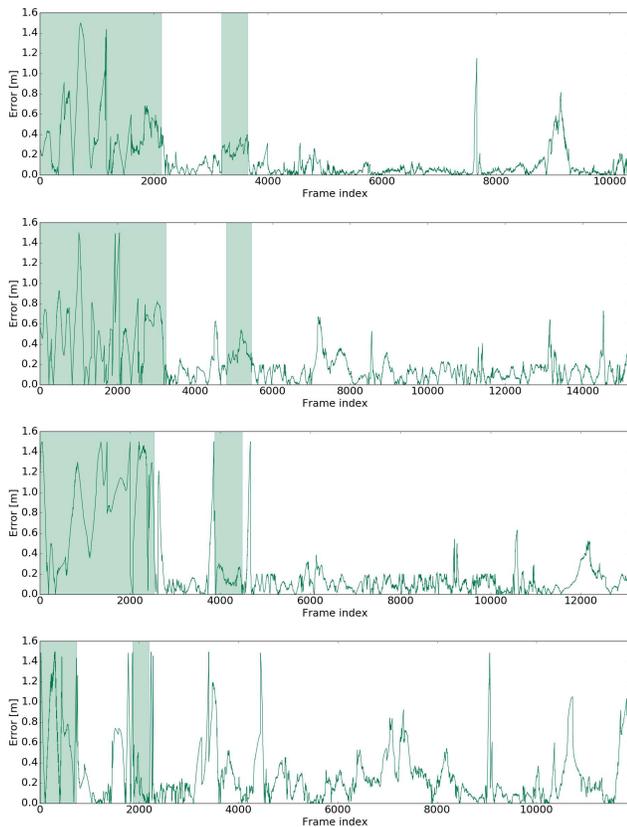


Fig. 8: Distance to reference trajectory in (from top to bottom) the mapping drive, cloudy weather, sunny weather and rain. Sections with narrow roads are highlighted in green.

to create new maps. The effort required for the 5 km long track was about 10 min, in which elements obscured by other vehicles were corrected.

Because our approach is modular in terms of the detectors used, it is easy to add more detectors to improve the localization result in specific situations. The quality of localization can thus benefit directly from future advances in high definition maps and progress in detection algorithms. Our results suggest that this makes sense especially in areas with lots of parked vehicles at the side of the road. Here, additional detectors, e.g. for house facades or trees could provide an improvement, since the detection of road borders alone is not sufficient to achieve the required accuracy.

ACKNOWLEDGEMENT

This work was funded by Daimler AG as part of the Tech Center a-drive initiative. The authors would like to thank Daimler AG for their support.

REFERENCES

[1] “Here hd live map - on the road towards autonomous driving.” <https://here.com/en/products-services/products/here-hd-live-map>, 2018. Accessed: 2018-02-20.

[2] “Tomtom hd map with roaddna.” automotive.tomtom.com/automotive-solutions/automated-driving/hd-map-roaddna, 2018. Accessed: 2018-02-24.

[3] M. Schreiber, H. Königshof, A. M. Hellmund, and C. Stiller, “Vehicle localization with tightly coupled gnss and visual odometry,” in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 858–863, 2016.

[4] H. Lategahn and C. Stiller, “Vision-only localization,” *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 3, pp. 1246–1257, 2014.

[5] M. Lundgren, E. Stenborg, L. Svensson, and L. Hammarstrand, “Vehicle self-localization using off-the-shelf sensors and a detailed map,” in *IEEE Trans. Intell. Veh.*, pp. 522–528, 2014.

[6] J. Levinson and S. Thrun, “Robust vehicle localization in urban environments using probabilistic maps,” in *IEEE Trans. Robot. Autom.*, pp. 4372–4378, 2010.

[7] M. Sons, M. Lauer, C. G. Keller, and C. Stiller, “Mapping and localization using surround view,” in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1158–1163, 2017.

[8] Z. Tao, P. Bonnifait, V. Frémont, and J. Ibañez-Guzman, “Mapping and localization using gps, lane markings and proprioceptive sensors,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 406–412, 2013.

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

[10] T. Heidenreich, J. Spehr, and C. Stiller, “Laneslam – simultaneous pose and lane estimation using maps with lane-level accuracy,” in *IEEE Trans. Intell. Transp. Syst.*, pp. 2512–2517, 2015.

[11] A. Y. Hata and D. F. Wolf, “Feature detection for vehicle localization in urban environments using a multilayer lidar,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 420–429, 2016.

[12] R. P. D. Vivacqua, M. Bertozzi, P. Cerri, F. N. Martins, and R. F. Vassallo, “Self-localization based on visual lane marking maps: An accurate low-cost approach for autonomous driving,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 582–597, 2018.

[13] J. Ziegler, P. Bender, and M. Schreiber, et al., “Making bertha drive – an autonomous journey on a historic route,” *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 2, pp. 8–20, 2014.

[14] M. Schreiber, C. Knöppel, and U. Franke, “Laneloc: Lane marking based localization using highly accurate maps,” in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 449–454, 2013.

[15] *Richtlinie für die Markierung von Straßen (Guidelines for Road Markings)*. German Department of Transport, 1980.

[16] M. Schreiber, A. M. Hellmund, and C. Stiller, “Multi-drive feature association for automated map generation using low-cost sensor data,” in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1140–1147, 2015.

[17] B. Ranft and T. Strauß, “Modeling arbitrarily oriented slanted planes for efficient stereo vision based on block matching,” in *IEEE Trans. Intell. Transp. Syst.*, pp. 1941–1947, 2014.

[18] F. Poggenhans, M. Schreiber, and C. Stiller, “A universal approach to detect and classify road surface markings,” in *IEEE Trans. Intell. Transp. Syst.*, pp. 1915–1921, 2015.

[19] Z. Wu, C. Shen, and A. van den Hengel, “Wider or deeper: Revisiting the resnet model for visual recognition,” *CoRR*, 2016.

[20] M. Cordts, M. Omran, and S. Ramos et al., “The cityscapes dataset for semantic urban scene understanding,” in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3213–3223, 2016.

[21] J. Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 679–698, Nov 1986.

[22] R. O. Duda and P. E. Hart, “Use of the hough transformation to detect lines and curves in pictures,” *Commun. ACM*, vol. 15, no. 1, 1972.

[23] S. J. Julier and J. K. Uhlmann, “A new extension of the kalman filter to nonlinear systems,” in *Signal Processing, Sensor Fusion, and Target Recognition*, vol. 6, 1997.

[24] J. Ziegler, H. Lategahn, M. Schreiber, C. G. Keller, C. Knöppel, J. Hipp, M. Hauéis, and C. Stiller, “Video based localization for bertha,” in *IEEE Trans. Intell. Veh.*, pp. 1231–1238, 2014.

[25] M. Sons, H. Lategahn, C. G. Keller, and C. Stiller, “Multi trajectory pose adjustment for life-long mapping,” in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 901–906, 2015.

[26] F. Poggenhans, A. M. Hellmund, and C. Stiller, “Application of line clustering algorithms for improving road feature detection,” in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2456–2461, 2016.

[27] Ö. Ş. Taş, N. O. Salscheider, F. Poggenhans, S. Wirges, C. Bandera, M. R. Zofka, T. Strauß, J. M. Zöllner, and C. Stiller, “Making bertha cooperate – team annieway’s entry to the 2016 grand cooperative driving challenge,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–15, 2017.