2014 IEEE Intelligent Vehicles Symposium (IV)
June 8-11, 2014. Dearborn, Michigan, USA

Video Based Localization for BERTHA

Julius Ziegler[1], Henning Lategahn[2], Markus Schreiber[1], Christoph G. Keller[3],
Carsten Knöppel[3], Jochen Hipp[3], Martin Haueis[3] and Christoph Stiller[2]

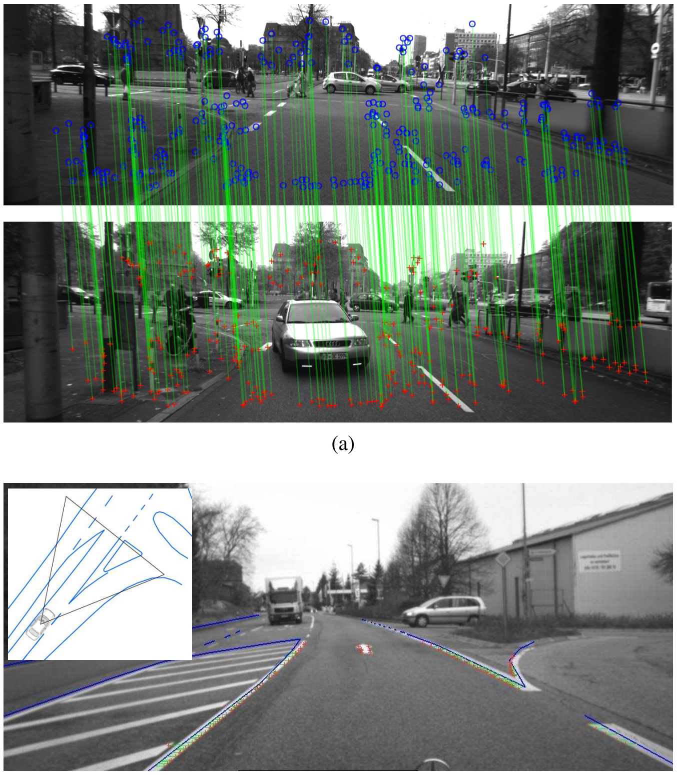
***Abstract*—In August 2013, the modified Mercedes-Benz S-Class S500 INTELLIGENT DRIVE ("BERTHA") completed the historic Bertha-Benz-Memorial-Route fully autonomously. The self-driving 103 km journey passed through urban and rural areas. The system used detailed geometric maps to supplement its online perception systems. A map based approach is only feasible if a precise, map relative localization is provided. The purpose of this paper is to give a survey on this corner stone of the system architecture. Two supplementary vision based localization methods have been developed. One of them is based on the detection of lane markings and similar road elements, the other exploits descriptors for point shaped features. A final filter step combines both estimates while handling out-of-sequence measurements correctly.**

I. INTRODUCTION

This paper gives a survey on the localization technologies that have been used for autonomous driving on the Bertha Benz Memorial Route (BBMR). BBMR is a historic, 103 km long route connecting the cities of Mannheim and Pforzheim, and passing through 25 towns.

For this challenge, a map based strategy was adopted. Maps allow to store all those static properties of the environment that the vehicle cannot detect reliably through its sensors. This applies, *e.g.,* to the geometric layout of lanes, especially where lane boundaries are not clearly marked, *e.g.* within intersections. Even more difficult is the detection of relations between geometries, *e.g.* the priority relation between lanes. To make use of map data, it is required to have precise *map-relative* localization. In the BBMR project, a trajectory for the vehicle is planned in a map-relative coordinate system. The vehicle is guided along this trajectory by a feedback controller, which tries to minimize the offset between planned trajectory and the current vehicle pose. Thus, the single quantity that is fed back is actually the estimated map-relative pose of the vehicle.

The map based approach called for accurate labeling of lane geometries. As a side effect, this enabled us to perform an analysis of the width of the driving corridor along BBMR. Fig. 2 shows the relative frequency of lateral clearance along the road, *i.e.* half the width of the driving corridor minus half the width of the car (without mirrors and door handles). Approximately 6% of the route have a lateral clearance of less than 0.35 m. This analysis only covers lane boundaries which are expected to be static. Note that the driving corridor can be smaller when blocked by non-permanent obstacles, like parked cars, however, these obstacles must be detected by sensors, and hence, in relative position to the car, anyway. Thus, the static analysis gives a good hint towards how precise localization has to be. Given a combined maximum error of controller and localization of < 0.2 m, we can expect to stay within the static driving corridor 99.9% of the time. Global navigation satellite systems (GNSS) - even those with real time kinematic (RTK) capabilities - are far away from achieving this precision under all conditions encountered along BBMR.


(a)

(b)

Figure 1: Two complementary methods for video based localization. Map data is in blue, observations on online video data in red and associations between both in green. (a) Point feature based localization (PFL). (b) Lane feature based localization (LFL).

Two complementary, video based methods have been developed to provide map relative localization at the required precision. To give an idea of their working principles, we anticipate some results in Fig. 1. Fig. 1a shows the principle

[1] J. Ziegler and M. Schreiber are with the FZI Research Center for Information Technology, Dep. on Mobile Perception Systems, 76131 Karlsruhe, Germany {ziegler|schreiber}@fzi.de

[2] C. Stiller and H. Lategahn are with Karlsruhe Institute of Technology (KIT), Department of Measurement and Control Systems (MRT), 76131 Karlsruhe, Germany {stiller|lategahn}@kit.edu

[3] C. G. Keller, C. Knöppel, J. Hipp and M. Haueis are with Daimler AG, Research and Technology, 71063 Sindelfingen, Germany firstname.lastname@daimler.com

978-1-4799-3637-3/14/$31.00 ©2014 IEEE

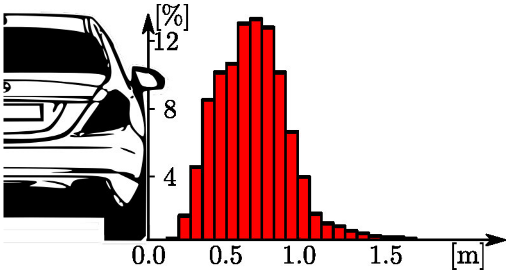


Figure 2: Relative frequency of lateral clearance along BBMR. Vehicle rear is drawn to scale.

of point feature based localization (PFL). The video data from a rear facing vehicle camera (bottom) is compared with a video sequence that has been acquired previously in a mapping run (top). It can be seen that the two images are shot from approximately the same position and angle, but at a different time of the year. The two images are registered spatially by means of descriptor based point feature association. The point features of the map sequence (blue) are associated to those detected on the current camera image (red), and a 6d rigid-body transformation between both camera poses is computed that would bring associated features in agreement. By combining this transformation with the global reference pose of the map image, a global position estimate is recovered.

In lane feature based localization (LFL), the reference map is a geometric, global representation of linear features which are located in the ground plane. A local excerpt of this map is shown in the top left corner of Fig. 1b. This map is brought into agreement with the video data from a forward facing stereo camera by detecting and associating lane features. Special detectors for solid lines, broken lines and curbstones have been implemented. The map of lane features is projected onto the camera image (blue). The back projection error (green) between detected features and the map geometry is minimized by means of a Kalman filter.

II. Related work

A rotating scanner with 64 single lasers has been used by Levinson and co-workers in [1]. The current laser scan is matched to a previously recorded 3D point cloud map. The laser beams additionally measure surface remittance, and this is adopted as an additional feature for matching. Based on a similar sensor setup, Moosmann and colleagues present a method for point cloud matching [2], reporting an improvement in localization accuracy over high precision GNSS systems. Both laser based methods achieve very good results and are inherently robust against illumination changes. However, the effort required in terms of sensor hardware is immense in comparison to our approach, which relies on cameras and inertial measurement only.

Cameras have been previously used by Badino in [3], [4]. Imagery was recorded for an urban area and holistic image features describing each single pose of the mapping trajectory are extracted from the images and stored as the map. During online localization, current image features are matched to the map and position estimates are smoothed by incorporating odometric information. The final position estimate will always correspond to exactly one pose of the mapping trajectory. Hence the precision of the method is limited, and in particular, because it will “lock” to the mapping trajectory, it is not suitable to determine a position error with respect to a reference trajectory.

PFL draws heavily on concepts which have been introduced earlier in the context of simultaneous localization and mapping (SLAM) [5], [6], [7], [8]. In SLAM, the state to be estimated consists of both the pose of the agent, and the global positions of landmarks. Most recent approaches to SLAM solve a large non-linear least squares problem whereas earlier methods are based on recursive filtering. To achieve real time performance, we decoupled map generation from vehicle localization. Preliminary versions of the PFL system have been presented earlier in [9], [10].

Lane feature based localization has been proposed previously by Pink [11]. Here, a single pose estimate was derived from an iterative closest point (ICP) match of image- to map features. The single pose was then used as an observation inside a Kalman filter framework. In our approach, we also use a Kalman filter, but our observations are the matching errors between single features. Thus, we achieve a tighter coupling between observations and the model state. A preliminary version of LFL has been presented in [12].

III. Methodology

In this section, we will present the two complementary methods that were developed for video based localization, namely point feature based localization (PFL) and lane feature based localization (LFL). The methods use separate camera systems as illustrated in Fig. 3. PFL is essentially monoscopic, however, for creation of the point feature map the camera system is extended to a stereo setup. Both methods use 6D motion sensors to support pose estimation. A 6D vector of velocities and angular rates are provided by accelerometers, yaw rate- and wheel speed sensors of an ego motion cluster (EMC). Satellite navigation (GNSS) is used to provide an initial pose estimate and reference positions during mapping runs. We assume the relative poses of all sensors are precisely known, which has been achieved by the automatic calibration method [13]. Results of PFL and LFL will be combined in a downstream data fusion stage (Sec. III-D).

A. Generating consistent mapping data and a reference trajectory

Both methods localize relative to a base map which is created from sensor data that has been recorded during a manual mapping run along BBMR. Let the mapping run contain N video frames. The set of reference positions $\hat{q}_i \in SE(3)$ attached to each frame of map data - the *reference trajectory* - defines the global coordinate system for localization. It is important to note that global *accuracy* is much less important

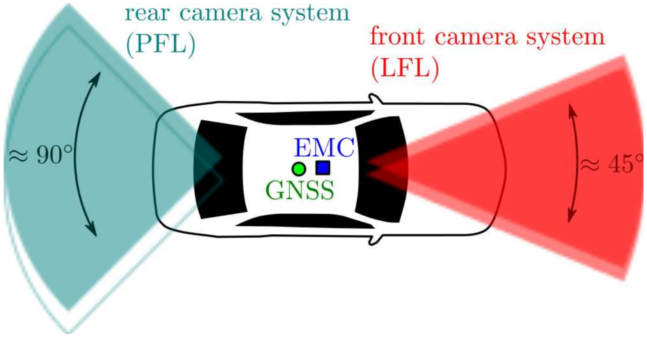


Figure 3: Sensor setup. The rear mono camera is extended to a stereo system only during the mapping stage.During mapping, GNSS provides a global reference position.

than map-relative *precision* - i.e. *repeatability*. Global localization accuracy mainly depends on the global accuracy of the map.

To provide a reference position during mapping, a GNSS system is used. Poor satellite visibility and undetected multipath scattering, however, cause step and impulse shaped disturbances of several meters. This is undesirable, especially in inner city scenarios, as these disturbances would reappear during online localization. Therefore, we combine GNSS readings and video based odometry [14] to largely mitigate these unwanted effects. This allows us to trade off global accuracy for smoothness and local consistency of the reference trajectory is assured.

Let g_i denote the purely GNSS based observations of the true mapping trajectory q_i, where $i \in [0, \ldots, N-1]$ is the discrete time index of a measurement frame. Poses and rigid body transformations are represented by $SE(3)$ e.g. homogeneous matrices consisting of a rotation matrix and translation vector. In addition to GNSS observations, the motion between two consecutive poses q_{i-1} and q_i is estimated by visual odometry [14] and denoted as o_i. It is parameterized by a vector $o_i \in \mathbb{R}^6$. Bringing the noisy pose observations g_i in line with the noisy motion observations o_i to yield an estimate $\hat{q}_i$ of the true mapping pose q_i can be posed as a least squares (LS) problem. In the following, we will use the notation of factor graphs [15] to illustrate construction of the LS objective function. The graph of the LS problem at hand is depicted in Fig. 4. Triangular nodes denote camera poses and squared nodes represent GNSS observations. Solid nodes always denote a constant, while hollow nodes represent alterable quantities and correspond to the free variables of the LS problem.

The graph of Fig. 4 can be summarized by the error function

$$E_{\text{map1}}(q_0, \ldots, q_{N-1}) = \sum_{i=2}^{N} ||\,(q_i \ominus q_{i-1}) - o_i||^2 + \sum_{i=1}^{N} ||g_i \ominus q_i||^2 \quad (1)$$

where $\ominus$ yields a minimally parameterized motion difference between two $SE(3)$ poses and $||\cdot||$ is a suitable Mahalonobis norm. Now, the reference trajectory is the minimizing argument of E_{map1},

$$\hat{q}_0, \ldots, \hat{q}_{N-1} = \underset{q_1,\ldots,q_N}{\arg\min} \{E_{\text{map1}}(q_0, \ldots, q_{N-1})\}, \quad (2)$$

and can be found by standard least squares methods [16]. We utilize the g2o library [7] which is specially suited for problems that are formalized as factor graphs.

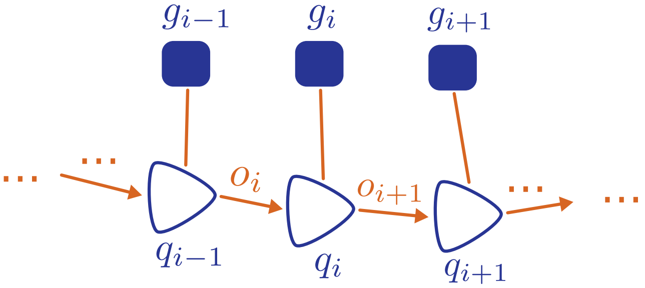


Figure 4: First step of mapping pipeline: Triangles denote camera poses (backward facing in this case) and squares represent GNSS readings. Solid variables are fixed during optimization. Edges between variables constitute constraints. Consecutive poses are e.g. constrained by visual odometry o_i.

B. *Point feature based localization*

1) Point feature map: For PFL, a map of 3D landmarks and their visual descriptions must be computed from the recorded imagery. To this end, we detect salient image points and associate these across all images. We refer to a set of pixel positions tracking a single point in 3D as a *tracklet*. Every landmark that is stored in the final map is computed from exactly one tracklet. Hence we obtain one pixel position and disparity (from stereoscopy) for a landmark l_j and a camera reference pose $\hat{q}_i$ (*cf.* Sec. III-A) and combine it in the measurement vector $z_{ij} = (u_{ij}, v_{ij}, d_{ij})^T$. The 3D landmark position l_j can then be estimated by minimizing the sum of squared back projection errors. More concisely, we define the landmark error function for one landmark l_j to be

$$E_{\text{map2}}(l_j) = \sum_i ||\pi(l_j, \hat{q}_i) - z_{ij}||^2 \quad (3)$$

with a camera projection function $\pi(l, p)$ that computes a pixel position and disparity from pose p and landmark l [17]. The sum extends over all pose indexes i that the landmark l_j was observed from. Note that poses are kept fixed and are not optimized. The minimizing argument of (3) is adopted as the landmark position estimate.

Finally, landmark positions are stored with their respective image descriptors to form the map. For high robustness to illumination changes we use our novel DIRD descriptor presented in [18], [19].

2) Online localization: PFL is a two step approach and yields a six degrees of freedom vehicle pose estimate. Firstly, we query the map for landmarks close to the current pose, associate these landmarks with the current (mono) image and

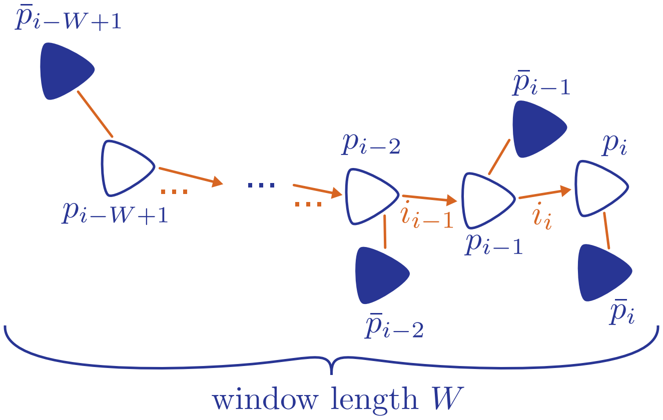


Figure 5: Second step of localization (pose adjustment): Single shot estimates $\bar{p}_i$ are balanced with odometry readings i_i to yield the final pose estimate. $\bar{p}_i$s serve as a prior in this step.

recover an ego position estimate. Since this estimate is based on exactly one image frame, we will refer to it as a *single shot estimate*. Secondly, a history of single shot estimates is fused with motion information from the EMC to yield a smoothed estimate of the vehicle pose. We will refer to this step as *pose adjustment*. We now address both steps in detail.

We denote the camera pose to be estimated at the time step i as p_i. Firstly, we derive the preliminary single shot estimate, $\bar{p}_i$. All landmarks in the immediate vicinity of the vehicle are found and associated with pixel positions of the current camera image. Let $(u_j, v_j)^\intercal$ denote the pixel position of the 3D landmark l_j, in the current image. An error function of sums of squared back projection errors is then defined by

$$E_{\text{loc1}}(p_i) \quad = \quad \sum_{ij} ||\pi(l_j, p_i) - (u_j, v_j)^T||^2. \qquad (4)$$

The minimizing argument $\bar{p}_i$ of (4) is computed by the methods of least squares (LS).

To mitigate and largely avoid the influence of incorrectly associated landmarks (outliers) we embed the aforementioned LS estimate into a RANSAC ([20]) scheme. Minimal sets of three landmarks are drawn randomly and the resulting pose hypothesis is searched for support. After a few hundred such iterations the largest inlier set is finally solved jointly yielding $\bar{p}_i$.

During a second pose adjustment step, the previous W single shot estimates $\bar{p}_{i-W+1}, \dots, \bar{p}_i$ are fused with motion information $i_{i-W+2}, \dots, i_i$ acquired from the EMC. The single shot estimate $\bar{p}_i$ is now the prior for the poses and combined with the EMC readings. The past W poses are jointly optimized such that the error function

$$\begin{aligned} E_{\text{loc2}}(p_{i-W+1}, \dots, p_i) \quad &= \sum_{k=i-W+1}^{i} ||\bar{p}_k \ominus p_k||^2 \qquad (5) \\ &+ \sum_{k=i-W+2}^{i} ||\left(p_k \ominus p_{k-1}\right) - i_k||^2 \end{aligned}$$

becomes minimal. Note the great similarity to (1). The factor graph is shown in Fig. 5.

The reason for separating both of these localization steps (single shot, pose adjustment) from each other is twofold. Firstly, in contrast to a combined, joint optimization of poses and landmark positions, the individual optimization problems are of moderate size. Hence, they can be solved efficiently, thereby reducing overall computational complexity. Secondly, any single shot estimate which does not comply with the relative motion within the specified time frame can easily be exposed by examining the residuals $\bar{p}_k \ominus p_k$ and thereafter pruning it from (5) further contributing to the overall robustness of the system.

C. Lane feature based localisation

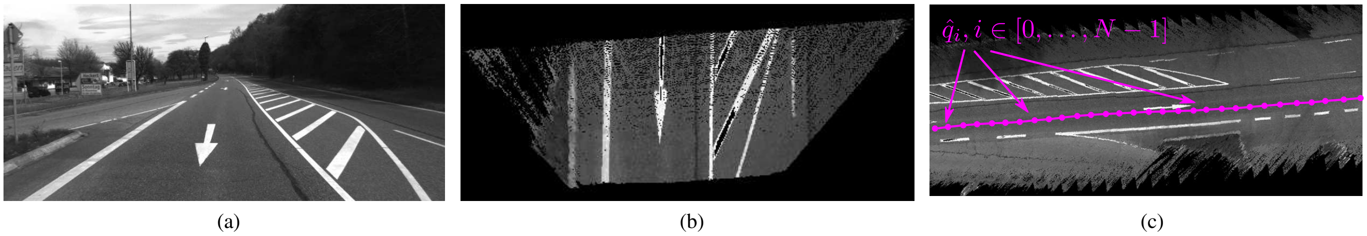
1) Map generation: For lane feature based localization, we must obtain a map that provides a global, geometric representation of lane markings and similar features along BBMR. We decided to use the exact same camera data as the basis for the lane feature map as was used to create the feature point map. This minimizes the possibility to introduce a miss match between LFL and PFL through misalignment of the reference sensors during the mapping stage. A prerequisite to the generation of a lane feature map is an image based top view map. Fig. 6 illustrates how such a top view is created from the rear view stereo camera system. Each single top view frame is referenced by a pose $\hat{q}_i$ of the reference trajectory from Sec. III-A.

The aggregated top view image is used as a backdrop in the JOSM editor from the OpenStreetMap project [21]. Lane markings with good visibility can be obtained from an automatic detector, but manual editing must be used to validate the data and to fill in gaps.

Fig. 7 shows an excerpt of map data during the manual editing process.

Formally, a lane feature map consists of a set of line segments with different attributes. Each line segment l_i is defined by a starting point $p_{s,i}$, an end point $p_{e,i}$ and a describing attribute a_i, where $p_{s,i}, p_{e,i} \in \mathbb{R}^2$ (latitude, longitude) and $a_i \subset \{\text{solid,broken,curb,stopline}\}$. In the case of broken lines, each p_i specifies beginning and end of a marked segment on the road. Stop lines usually occur perpendicular to the driving direction, and hence, a different kind of detector will be used during online localization.

2) Online localization: In principle, lane feature based localization is derived from a conventional lane marking detector [22]. In conventional lane marking detectors, a parametric model curve is iteratively fitted to measurements, *e.g.* a polynomial or a clothoid. For localization, the model curve is replaced by the rigid geometry of the lane map, which is parameterized by the pose of this model relative to the camera. The model fit is performed iteratively by a Kalman filter. This allows to incorporate a motion model into the estimator. LFL performs a 2D localization only and the map geometry is assumed to be flat. The camera pitch angle towards the ground



(a)

(b)

(c)

Figure 6: Creation of a stereo based topview map: (a) rear camera image. (b) topview from stereo reconstruction. (c) aggregation of single topview frames based on reference trajectory.

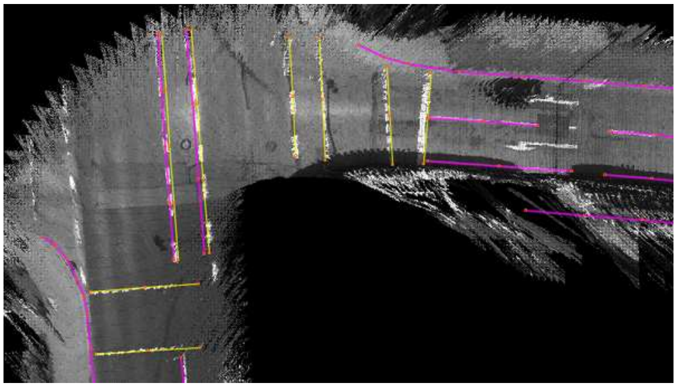


Figure 7: Map data on top view imagery

plane is estimated independently by analyzing v-disparity [23] [24].

a) System model: The ego vehicle state to be estimated is described by the state vector $\mathbf{x} = (x, y, \varphi)^\intercal$ where $(x, y)^\intercal$ is the vehicles 2D position and φ its orientation. The discrete system model is defined by the state transition function

$$f(\mathbf{x}, \mathbf{u}) = \begin{pmatrix} x + v \cdot \Delta t \cdot \cos\left(\varphi + \Delta t \cdot \dot{\psi}\right) \\ y + v \cdot \Delta t \cdot \sin\left(\varphi + \Delta t \cdot \dot{\psi}\right) \\ \varphi + \Delta t \cdot \dot{\psi} \end{pmatrix},$$

with the control input $\mathbf{u} = (v, \dot{\varphi})^\intercal$ and v the velocity of the vehicle, and $\dot{\varphi}$ its yaw rate. Velocity and yaw rate can be measured very accurately by the EMC. The discrete time step Δt is the cycle time between two consecutive innovation steps. The system covariance matrix $\mathbf{P_{xx}}$ is chosen to match the accuracy of velocity and yaw rate measurement.

b) Observation model: Assume that a match between a global 2D-point $\mathbf{p}_{\mathrm{map},i} = (x_{\mathrm{map},i}, y_{\mathrm{map},i})^\intercal$ on the lane feature map and a 2D-point $\mathbf{p}_{\mathrm{obs},i} = (x_{\mathrm{obs},i}, y_{\mathrm{obs},i})^\intercal$ observed in a coordinate system relative to the vehicle by the stereo camera system has been established (matching will be addressed in the next paragraph), and that this is the ith of N matches. The functional observation model $h_i(\mathbf{x})$ for the single match consists of transforming the map point to the vehicle coordinate system, *i.e.*

$$h_i(\mathbf{x}) = \begin{pmatrix} \cos\varphi & \sin\varphi & x \\ -\sin\varphi & \cos\varphi & y \end{pmatrix} \begin{pmatrix} x_{\mathrm{map},i} \\ y_{\mathrm{map},i} \\ 1 \end{pmatrix}.$$

The complete observation vector $\mathbf{z}$ is created by stacking all $\mathbf{p}_{\mathrm{obs},i}$, thus $\mathbf{z} = \begin{pmatrix} \mathbf{p}_{\mathrm{obs},0}^\intercal & \ldots & \mathbf{p}_{\mathrm{obs},N-1}^\intercal \end{pmatrix}^\intercal$, and analogously, the full observation model becomes

$$h(\mathbf{x}) = \begin{pmatrix} h_i(\mathbf{x})^\intercal & \ldots & h_{N-1}(\mathbf{x})^\intercal \end{pmatrix}^\intercal$$

The observation covariance matrix $\mathbf{P_{zz}}$ has been chosen to accommodate both map- and stereo reconstruction errors. In stereo imaging, the reconstruction error depends on the distance of the reconstructed point from the camera, *i.e.* on $x_{\mathrm{obs},i}$. It rises linearly for the y-coordinate, and quadratically for the x-coordinate, hence

$$\mathrm{Cov}(\mathbf{p}_{\mathrm{obs},i}) = \begin{pmatrix} (\sigma_{\mathrm{disp}} a x_{\mathrm{obs},i}^2)^2 & 0 \\ 0 & (\sigma_{\mathrm{disp}} b x_{\mathrm{obs},i})^2 \end{pmatrix},$$

where σ_{disp} is the standard deviation of disparity matching (in pixel) and a, b are constants which depend on the stereo imaging geometry.

To yield robustness against outliers, we employ a method proposed by Hartley and Zissermann [25]. With it, the influence of measurements that are far away from their predicted positions gets limited through a Cauchy function.

c) Map matching: The lane feature map represents markings or curbs as line segments, while our measurements are obtained as points. The goal of map matching is to achieve the best possible match of a point measurement to a point on a line segment. Ideally, each point detection (red) is associated to the true correspondent in the map (blue), as is illustrated schematically in Fig. 8a. In practice, this cannot be achieved. Alternatively, each measurement can be assigned the closest point on an associated line segment, as illustrated in Fig. 8b. In practice, we achieve an approximation to this behavior by densely sampling points from the line segment and associating each measurement to the closest of these points. This is depicted in Fig. 8c. An example of map-to-measurement association can be seen in Fig. 1b.

d) Measurement Extraction: Specialized detectors for both lane markings and curbstones have been developed. For lane marking detection, an oriented matched filter (dark-light-dark) is applied to the gray scale image to detect lane marking candidates, as is standard in conventional lane marking trackers [22]. To reduce false detections, only free space areas are considered. Free space is extracted by the stereo based method described in [26]. We only search for lane marking candidates in the vicinity of the lane marking map, which is projected

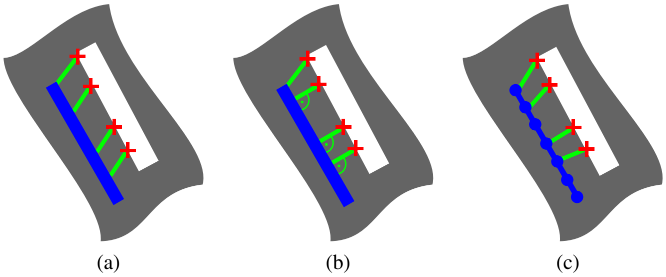


Figure 8: Matching point detections to map segments.

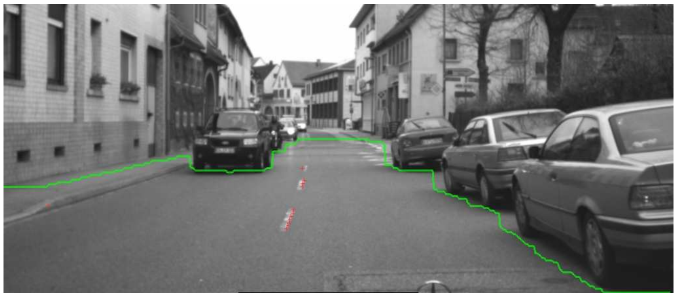


Figure 9: Measurement of lane markings (red crosses) under consideration of freespace (green).

into the gray scale images based on the current, predicted pose- and pitch estimate. Fig. 9 shows an example of lane marking detection under consideration of free space. Through stereo processing, a point in the vehicle coordinate system can be reconstructed for the detected image points.

We adpoted the robust approach to curb recognition that was proposed in [27]. Both gray value and disparity are used to classify and reconstruct curb stones. Fig. 10 shows an example of curb stone detection in urban area.

D. Data fusion

Before serving as input to the trajectory controller, the pose estimates obtained by PFL and LFL are combined in a model based fusion stage. This serves multiple purposes:

- PFL and LFL update rates are bound by the video rate and the required processing time. However, the trajectory controller feedback loop runs at a constant rate of 100 Hz, hence, an interpolated position estimate must be provided at this rate.
- Because of processing delays, especially of PFL, pose updates can arrive out-of-sequence (OOS). Data fusion re-orders and re-processes OOS-measurements.
- Integrating two redundant pose estimators enables additional validation and outlier rejection.
- The fusion method is based on an accurate dynamic vehicle model and has predictive capabilities that allow to bridge moderate outage times even of both pose estimators.

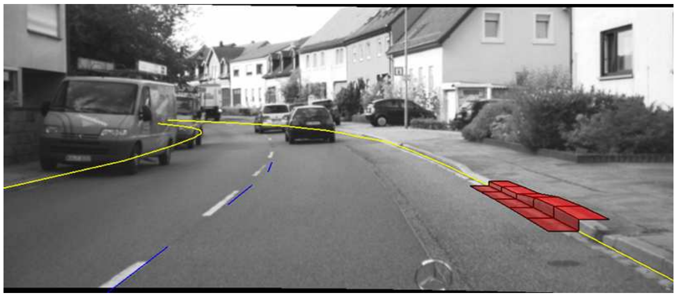


Figure 10: Curb classified patches on the right side.

Data fusion is performed by using a sigma-point Kalman filter that estimates the state of a full dynamic one track model of the vehicle. The state vector to be estimated is $\mathbf{x} = (x, y, \varphi, \dot{\varphi}, v, a, \beta, \delta)^{\intercal}$, where x and y are the position of the vehicle's center of gravity (COG), φ its orientation, v its velocity, a longitudinal acceleration, β the slip angle at COG and δ the deflection of the front wheels. The system model differential equation is

$$\dot{\mathbf{x}} = \begin{pmatrix} v\cos(\varphi + \beta) \\ v\sin(\varphi + \beta) \\ \dot{\varphi} \\ A\beta + B\frac{\dot{\varphi}}{v} + C\tan\delta \\ a \\ 0 \\ D\frac{\beta}{v} + (\frac{E}{v^2} - 1)\dot{\varphi} - F\frac{\tan\delta}{v} \\ 0 \end{pmatrix}$$

where A, B, C, D, E, F are constants which depend on the steering geometry, mass, moment of inertia and tire characteristics of the car [28]. We assume that the model is driven by additive Gaussian noise on both the longitudinal acceleration and the steering angle.

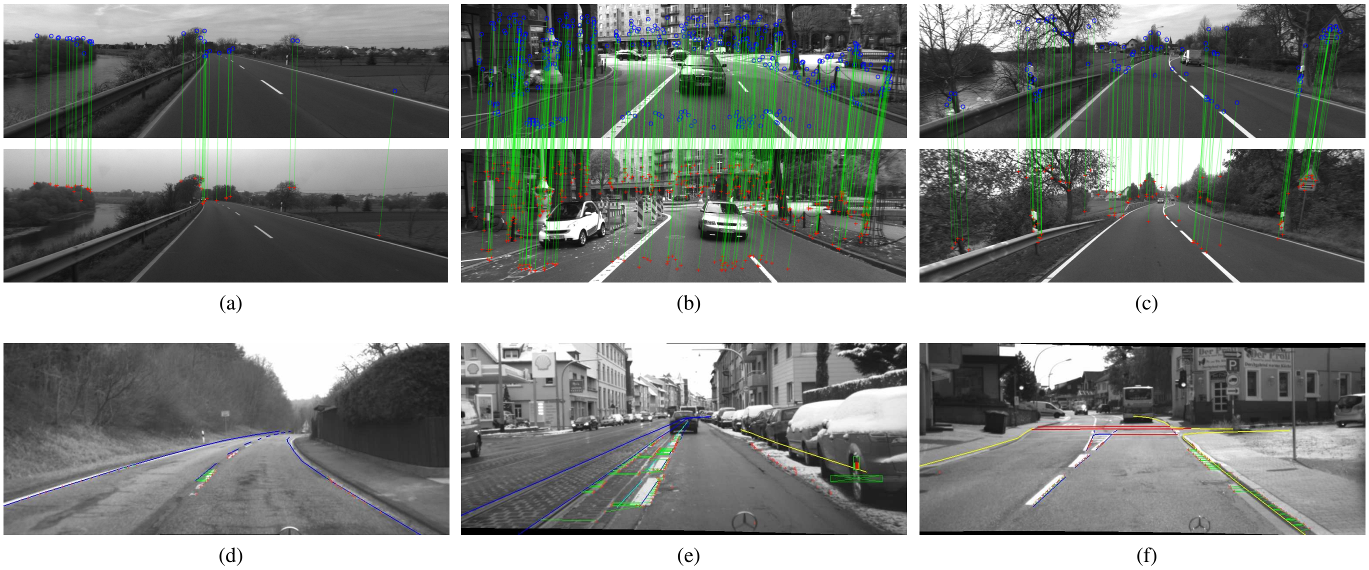
Four observe models have been implemented to incorporate measurements. PFL and LFL both observe x, y and φ directly. The quantities v, a and $\dot{\varphi}$ are observed by the EMC. The steering angle δ is observed by an encoder in the steering column.

Observations are rejected as outliers in the observe step if they violate a 3σ criterion with respect to the predicted state covariance.

For treating OOS-measurements, a finite history of observations and filter states are buffered and kept sorted by time. If a measurement taken at t_{meas} arrives out of sequence, the filter state is re-initialized to the newest stored filter state which has a time-stamp $t < t_{\text{meas}}$. The new measurement and all buffered measurements with a younger time-stamp a re-processed in order. Estimates between sensor measurements are generated by a prediction step (without a subsequent observe step).

IV. Experiments and discussion

In August 2013, the BBMR was completed in autonomous mode, in multiple sections. A quantitative evaluation of localization performance is difficult because a precise reference sensor to compare against does not exist. However, the successful journey along BBMR gives a strong indication that the requirement which we presented initially could be met. Some quantitative evaluation of a preliminary version of PFL has



(a) (b) (c)

(d) (e) (f)

Figure 11: Strengths and weaknesses of PFL and LFL. Top row: PFL. Online observations red, map features blue. Bottom row: LFL. Red crosses are online observations. Map features are blue (lane markings and rails) yellow (curbstones) and dark red (stop lines).

been presented in [10]. Here, an average precision of 10 cm was reported, under variable illumination conditions.

An important finding is that the two core methods presented, PFL and LFL, complement each other well. PFL works best where man-made structures are abundant, *i.e.* in inner city scenarios. Complementary, LFL excels in rural areas, where lanes are typically clearly marked. Fig. 11 contrasts some strengths and weaknesses of both methods. Fig. 11a and 11d show rural scenes which are almost void of man-made structures. PFL (top) finds only 32 associations to map features. Moreover, these associations are with structures which are far away from the car, and hence, give bad support for its position. Below, a similar scene is displayed from the perspective of LFL. As can be seen, lane markings are correctly associated to map features. Fig. 11b and 11e show inner city scenes. PFL (top) finds approximately 400 map matches. This is allows to even compensate for considerable deviation between the map- and online image which is induced by a temporal construction site (visible in the online image). Fig. 11e shows an inner city scene which poses some difficulties for LFL. The row of parking cars to the right occludes a curbstone that is in the map (yellow). Snow deposits induce deceptive observations near the right lane border. Further deceptive observations are created by tarred fissures in the tram's track bed. The high number of collateral features in the left half of the image causes some erroneous associations (green).

We would like to point out that dropouts like in Fig. 11a and 11e do occur, but are not the rule. In general, PFL does work on rural roads, because some reliable features, like sign- and reflector posts, fissures in the road surface and tree trunks can be found almost anywhere (*cf.* Fig. 11c). Likewise, most inner city scenarios are abundant with lane marking- or curbstone features which provide good support for LFL (*cf.* Fig. 11f). Nevertheless, none of both methods was capable of individually providing a correct position estimate all of the time. However, they complement each other perfectly, so that they work very reliable in combination.

V. Conclusions and Outlook

The BBMR experiment showed that video based localization can not only replace, but outperform expensive real time kinematic satellite navigation systems. The two different methods presented here - PFL and LFL - complemented each other in a natural way. In combination, they are competitive with laser scanner based systems in terms of precision, but surpass them in terms of economy. Nevertheless, we would like to point out some directions for improvement.

The decision to strongly decouple the two localization methods and to combine them only in a separate fusion stage was mainly made to reflect the hardware architecture, with the two camera systems being connected to separate computer systems. However, from a scientific point of view, it is more desirable to tightly couple both methods. This could be achieved by deciding for one estimator framework - either a recursive filter as employed in both LFL and the fusion stage, or the regression method used for PFL - and re-phrasing the respective measurement equations in the context of that framework. *E.g.*, (3) could be augmented by an error term that sums the squared point-to-line errors that were used as the measurement residuals in the LFL method.

While most parts of BBMR were fairly flat, some passages showed significant vertical variation. In these areas, the flat-world assumption made in LFL prevented a good fit of the model data to the image data. This is especially true at increasing distance from the camera. Thus, we believe that augmenting the lane feature map with a vertical elevation profile will enable to exploit a larger local neighborhood of the map and hence, improve robustness and accuracy of fit.

Our method critically depends on map information. Currently, map creation for LFL is partially a manual process. Since this will not easily scale to an industrial application, an automated mapping process is sought. Our aim is to completely eliminate manual annotation and to replace it by automatic processing of imagery and other data.

As described in Sec. III-A, to guarantee consistency, all data used for mapping ideally is acquired during one single mapping run. In practice, this is not always possible even for a constrained scenario like the BBMR, e.g. if temporal detours are present. Thus, some ad-hoc methods had to be developed to patch-up incomplete mapping data. Currently, we are developing methods to automatically register multiple mapping runs.

REFERENCES

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

[2] F. Moosmann and C. Stiller, "Velodyne SLAM," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, (Baden-Baden, Germany), pp. 393–398, June 2011.

[3] H. Badino, D. Huber, and T. Kanade, "Visual topometric localization," in *Intelligent Vehicles Symposium (IV)*, (Baden-Baden, Germany), June 2011.

[4] H. Badino, D. Huber, Y. Park, and T. Kanade, "Real-time topometric localization," in *International Conference on Robotics and Automation (ICRA)*, (St Paul, Minnesota, USA), May 2012.

[5] A. Davison, I. Reid, N. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 1052–1067, 2007.

[6] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part ii," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 3, pp. 108–117, 2006.

[7] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 3607–3613, IEEE, 2011.

[8] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous robots*, vol. 4, no. 4, pp. 333–349, 1997.

[9] H. Lategahn and C. Stiller, "Vision only localization (to appear)," *IEEE Transactions Intelligent Transportation Systems*, 2014.

[10] H. Lategahn, H. Schreiber, J. Ziegler, and C. Stiller, "Urban localization with camera and inertial measurement unit," in *Intelligent Vehicles Symposium*, (Gold Coast, Australia), IEEE, 2013.

[11] O. Pink, "Visual map matching and localization using a global feature map," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*, pp. 1–7, IEEE, 2008.

[12] M. Schreiber, C. Knöppel, and U. Franke, "LaneLoc: Lane marking based localization using highly accurate maps," in *Intelligent Vehicles Symposium*, (Gold Coast, Australia), IEEE, 2013.

[13] A. Geiger, F. Moosmann, O. Car, and B. Schuster, "Automatic camera and range sensor calibration using a single shot," in *International Conference on Robotics and Automation (ICRA)*, (St. Paul, USA), May 2012.

[14] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3d reconstruction in real-time," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pp. 963–968, IEEE, 2011.

[15] F. Kschischang, B. Frey, and H. Loeliger, "Factor graphs and the sum-product algorithm," *Information Theory, IEEE Transactions on*, vol. 47, no. 2, pp. 498–519, 2001.

[16] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge University Press, 2007.

[17] R. Hartley and A. Zisserman, *Multiple view geometry*, vol. 642. Cambridge university press Cambridge, UK, 2000.

[18] H. Lategahn, J. Beck, B. Kitt, and C. Stiller, "How to learn an illumination robust image feature for place recognition," in *Intelligent Vehicles Symposium*, (Gold Coast, Australia), IEEE, 2013.

[19] H. Lategahn, J. Beck, and C. Stiller, "Dird is an illumination robust descriptor," in *Intelligent Vehicles Symposium*, (Dearborn, Michigan, USA), IEEE, 2014.

[20] M. Fischler and R. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[21] F. Ramm, J. Topf, and S. Chilton, *OpenStreetMap*. UIT Cambridge, 2007.

[22] E. Dickmanns and A. Zapp, "A curvature-based scheme for improving road vehicle guidance by computer vision," in *Mobile robots: proceedings of SPIE, vol. 727 (1987)*, 1987.

[23] R. Labayrade, D. Aubert, and J. P. Tarel, "Real time obstacle detection in stereovision on non flat road geometry through "v-disparity" representation," in *Intelligent Vehicle Symposium, 2002. IEEE*, vol. 2, pp. 646–651 vol.2, June 2002.

[24] A. Wedel, U. Franke, H. Badino, and D. Cremers, "B-spline modeling of road surfaces for freespace estimation," in *Intelligent Vehicles Symposium, 2008 IEEE*, pp. 828–833, IEEE, 2008.

[25] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, vol. 2. Cambridge Univ Press, 2000.

[26] D. Pfeiffer and U. Franke, "Towards a global optimal multi-layer stixel representation of dense 3D data," *BMVC, Dundee, Scotland. BMVA Press (August 2011)*, 2011.

[27] P. Greiner, M. Enzweiler, C. Knoeppel, and U. Franke, "Towards multi-cue urban curb recognition," in *IEEE Intelligent Vehicles Symposium*, (Gold Coast, Australia), 2013.

[28] H. Pacejka, *Tyre and Vehicle Dynamics*. Automotive engineering, Butterworth-Heinemann, 2006.