Efficient Global Localization Using Vision and Digital Offline Map

Martin Buczko[1] and Volker Willert[1]

***Abstract*— We present an efficient method that combines visual odometry and offline map data for global localization. In contrast to other approaches, no further sensor as GPS or precise information on the initial pose is needed. Rough localization information like the current city or region can be used to speed up the initialization, but is not required. The proposed method uses the observation of street signs in a twofold manner: To roughly pre-localize the vehicle and to subsequently transform the three-dimensional vehicle-to-map-alignment-task into a one-dimensional search within a closed interval. This leads to a fast matching method, that finds the proper shift and rotation angle between the planar trajectory of the visual odometry and the offline map data. The method is evaluated on a real trajectory that is computed with a state-of-the-art visual odometry system to demonstrate its performance.**

I. Introduction

According to the German accident statistics, that are provided by the government, 77 % of all traffic accidents with injured persons in 2015 were caused by the driver. For detailed statistics, we refer to [25]. A major part of these accidents could be prevented by assistance systems, that monitor turning maneuvers, the right of way, overtaking and the distance to vehicles that are driving ahead of the ego-vehicle. In case of a potentially critical scenario, these systems can avoid crashes by warning the driver or by intervening. The prerequisite for this is the availability of the position of surrounding traffic participants relative to the ego-vehicle. This can be achieved by determining each vehicle's position independently and providing it to other nearby vehicles. What comes as a support function for human drivers, evolves into a backbone at autonomously driving vehicles. Here, knowledge on the ego-position and that of nearby vehicles is essential to navigate and prevent accidents. Currently, global localization is mostly determined with GPS sensors. Advantages as described in [1] are the potential accuracy up to centimeters. On the other hand, effectiveness can suffer heavily in scenarios as urban driving, see [7]: Here, the availability reaches only 30 % in Hong Kong. The authors of [17] determine about the same percentage for the city of Calgary. Evaluation on a construction site, described as comparable to urban canyons, even lead to only 10 % availability in [15]. Due to the importance of reliable localization, numerous other methods were developed, such as using the signals of DVB-T emitters with known position. This approach turned out to be not sufficiently reliable due to the low number of emitters as stated in [1].

[1]Martin Buczko and Volker Willert are with
Control Methods & Robotics Lab,
TU Darmstadt, Germany.
www.rmr.tu-darmstadt.de/{buczko, willert}

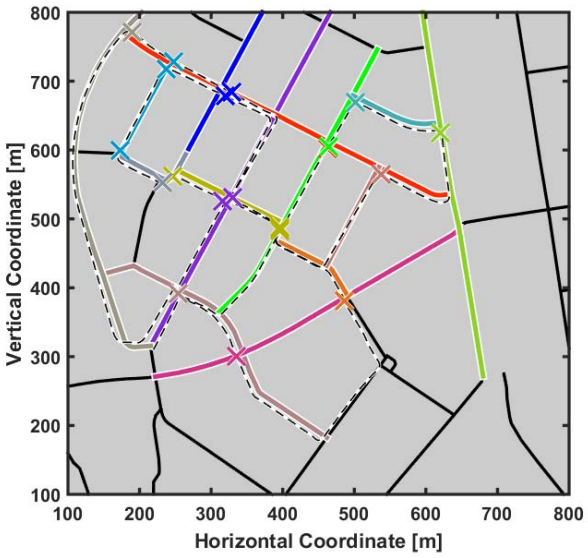

Fig. 1. Visualization of the principle idea of our approach: The dashed black line shows the trajectory, which is reconstructed by visual odometry. The colored crosses mark the observations of well visible street signs at a certain point of the trajectory, where each color stands for one street name on a sign. The colored lines show the corresponding streets from the offline map. We use the observed street signs to globally register the local, relative trajectory to the map. Furthermore we speed up this process by substituting the three degrees of freedom by one parameter, which is limited to a closed interval. The method requires only camera and offline map data.

According to [1], the same accounts for wireless local area networks and bluetooth networks, which potentially achieve an accuracy within several meters. Using RFID reference points for global localization is presented in [11]. Though enabling precise global localization, the infrastructure consisting of passive RFID transponders, placed at known positions must be available. This is given in special scenarios, but does not meet the requirements of automotive application.
In this work, we present an approach to recover the ego-position, using a stereo camera as only sensor and offline map data for global registration. It is organized as follows: We give an overview of existing camera-based global localization approaches and landmark extraction in Sec. II. Next, we briefly introduce the visual odometry system, that is used to reconstruct the vehicle's trajectory relative to the starting point[2] in Sec. III. Subsequently, we describe how the approximate absolute position of the relative trajectory's starting point is determined in Sec. IV and efficiently refined in Sec. V. We conclude with the evaluation of our concept in Sec. VI and an outlook on the next steps in Sec. VII.

[2]The absolute position of the starting point is not known, since we assume no further sensor to be involved.

978-1-5386-1526-3/17/$31.00 ©2017 IEEE

II. Related Work

A. Methods for Camera-Based Global Localization

In 2013, Daimler presented the 100 km autonomous travel of a modified S-Class, refered to as *Bertha Benz* project in [9]. They hypothesize, that detailed maps will be an essential element of future autonomous vehicles. In their work, they manually mark features in images along the track and thereby create a visual map. When driving, matching current images to these features is used to get an estimate of the current vehicle position. This is then fused with GPS data and vehicle sensors for maximum availability and precision of the localization. The authors of [27] combine the landmark-based localization with visual odometry. Visual odometry provides locally precise and high frequent relative motion updates. This allows a sparse use of landmark-based localization to limit the inevitable drift of visual odometry. They propose to build a map with 3D coordinates, 2D positions in the image and the HoG descriptor of distinct features. The map is then matched to features from the current position for global localization. Also in [12], [23], the authors use a previously generated map for localization. They use SIFT features to compare the current image to a previously built map. The fact, that specialized offline maps are required for all these approaches causes serious effort: Realizing the methods for wide areas leads to large required map data. Furthermore, time horizons in the scale of months or years violate the assumption of a static world, which reduces the usable features in the map and makes remapping necessary. The costs of this may be a big obstacle for market introduction. In this paper, we want to help closing this gap for urban scenarios by exploiting the information of OpenStreetMap[3] data that does not contain visual landmarks and is therefore much more persistent over time.

B. Street Name Recognition

In our approach, we manually label frames with well visible street signs. We do so, because the focus of this paper is the concept of how to fuse this information with the trajectory from visual odometry and map data to achieve global localization, not the detection itself. In the following, we discuss why the manual labeling is a reasonable simplification without severe limitations to applicability.

Publications as e.g. [18] deal with the recognition of text in urban scenarios. In contrast to the assumed scenario, where only street signs are relevant, they additionally try to detect more complicated kinds of texts as e.g. store names. This data contains much more variance, considering contrast and font style. As they point out, one has to differ between optical character recognition (OCR) and optical text detection: Current OCR methods are shown to achieve bad performance, when working on full images. Much better results can be reached, when previously detecting areas that contain text. In [18], this is done by image segmentation, character filtering, character grouping and text region filtering. Downsampling to lower resolutions is used to deal with varying font sizes and styles. In the evaluation of [18], urban text recognition reaches 80 % of the OCR score of a hand-cropped reference evaluation. In approaches as [24] more than 99 % accuracy is reached at detecting traffic signs in grayscale images which even outperforms the results of human classification. We hypothesize, that transferring those methods to street sign detection should lead to very good results in this application.

[3]Map data copyrighted OpenStreetMap contributors and available from http://www.openstreetmap.org

C. Localization Initialization

Our proposed method estimates the vehicle's global location without depending on global sensors. It does not even require a rough initial estimate. Therewith it differs from methods as that described in [20]. There, the authors present a method to fuse the trajectory from a visual odometry system with map data to back up GPS navigation assistance. In [13], a low-cost GPS is combined with vehicle odometry, which provides global information on the current vehicle location. In [3], the authors hypothesize, that their system is independent of external vehicle localization initialization. Nevertheless, they have to initialize their method with the assumption, that the probability of each point within this map to be the starting point of the trajectory is equal. This leads to high computational costs for large maps and induces high ambiguity. The method was tested for maximum areas of about 18 km^2, which we consider as rather local map patches.

III. Local Trajectory from Visual Odometry

Visual odometry describes the reconstruction of rotation $\mathbf{R}_V$ and translation $\mathbf{T}_V$ of a camera system attached to a mobile platform using the visual data, see [22]. Based on the difference between the optical flow from all features f_n^{t-1} indexed by n at time $t-1$ to subsequent image's corresponding features f_n^t and the modeled flows according to the motion hypothesis as:

$$(\hat{\mathbf{R}}_V^t, \hat{\mathbf{T}}_V^t) = \text{argmin}_{\mathbf{R}_V,\mathbf{T}_V} \sum_{n=1}^{N} \|\mathbf{x}_n^{t-1} - \pi(\mathbf{R}_V Z_n^t \hat{\mathbf{x}}_n^t + \mathbf{T}_V)\|_2^2 \tag{1}$$

We denote $\{\mathbf{x}_n^{t-1}, \mathbf{x}_n^t\} \in \mathbb{R}^3$ as the correspondent pair of coordinates $\mathbf{x}_n^t = [x_n^t, y_n^t]^T$ for all 3D points $\mathbf{X}_n^t = [X_n^t, Y_n^t, Z_n^t]^T$. The respective image coordinates with known focal length f and principle point $\mathbf{o} = [o_x, o_y]^T$ at time t can be written as $\hat{\mathbf{x}}_n^t = [\hat{x}_n^t, \hat{y}_n^t, 1]^T = \left[\frac{x_n^t - o_x}{f}, \frac{y_n^t - o_y}{f}, 1\right]^T$. The planar projection π maps 3D points to pixel coordinates as $[X,Y,Z]^T \mapsto [fX/Z + o_x, fY/Z + o_y]^T$. Recently published visual odometry systems provide the vehicle's trajectory with high reconstruction quality as presented e.g. in [4], [5], [8]. In this work, we base on [4], where rotation $\hat{\mathbf{R}}_V^t$ and translation $\hat{\mathbf{T}}_V^t$ are calculated in dedicated processes. As shown, this allows the incorporation of an effective outlier rejection scheme and leads to precise reconstruction. We refer to the publication for details of the method. The resulting system allows reconstruction with less than 0.9 % mean translation error and less than 0.0025 °/m mean rotation error, according to the metrics, introduced in [10]. From the outcome of the visual odometry module, we use the planar trajectory, $\tau \in \mathbb{R}^{2\times T}$ with T locations.

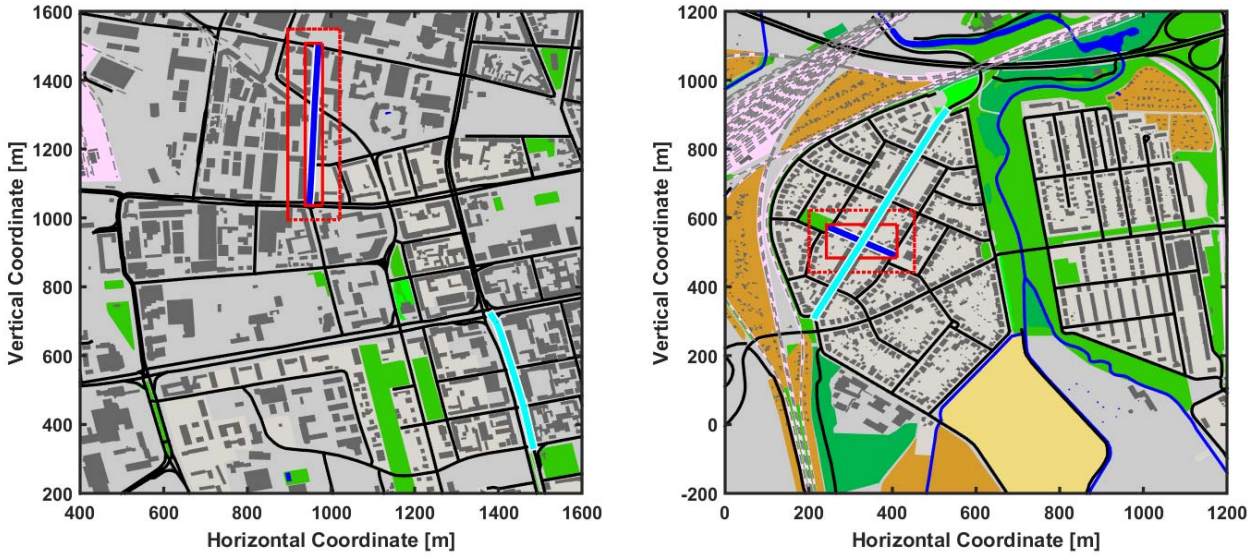

Fig. 2. Example for initialization with camera and map information only. The left map patch shows an extract from a map of Darmstadt and the right patch one of Karlsruhe. *Feldbergstrasse* (dark blue line in solid red bounding box) and *Neckarstrasse* (cyan line), corresponding to the first and second observed street signs are marked. Evaluating the trajectory from visual odometry, the distance between the relative positions at the observation-times of the first and second street are known. Extending the solid bounding box with this distance, a second bounding box (dotted red line) is created. At least one element of the second street must be inside this box, if the map topology fits to the measured scenario. As can be seen, the left scenario does not fulfill this requirement and is therefore not a possible map patch. In the right map patch the second street is contained in the dotted bounding box. It therefore is a possible map patch. This evaluation is carried out for all possible candidates. If only one map patch remains, the initialization is found. If not, further observations of street signs can be evaluated in the same way, to get a unique initialization.

IV. Map Patch Preselection

Most other visual localization systems assume knowledge on the initial position of the vehicle. By contrast, our approach is not restricted to that. Especially in Manhattan-like topologies, where ICP-based approaches tend to fail, our proposed method does not suffer from the inherent ambiguities of the streets' arrangement. Of course, an initial position can nevertheless be integrated to reduce computation time. However, if no such information is available, the initial position can be determined by fusing the trajectory from visual odometry, offline map data and the information on recognized street signs. The offline map Ω consists of J streets, $\Omega = [\omega_1, \cdots, \omega_J]$, where each street ω_j consists of P_j nodes, $\omega_j = [\omega_j^1, \cdots, \omega_j^{P_j}]$, that each contain a vertical and horizontal coordinate $\omega_j^p = [\omega_{j,x}^p, \omega_{j,y}^p]^T$. We denote $\omega_{A,c_A}/\omega_{B,c_B}$ as the lists of streets with the names of the first/second observed street sign at time t_A/t_B and τ_A/τ_B as the respective location on the VO-trajectory. We investigate the combination of places on the streets $\omega_{A,c_A}(\lambda_A)$ and $\omega_{B,c_B}(\lambda_B)$, where $\lambda_{A/B}$, $0 \leq \lambda_{A/B} \leq 1$ uniquely parameterizes the position on the street. This is performed by iterating over the candidate indices c_a for the first and c_b for the second observed street until the following condition is fulfilled:

$$\exists(\lambda_A, \lambda_B): \ \|\omega_{A,c_A}(\lambda_A) - \omega_{B,c_B}(\lambda_B)\|_2 = \|\tau_A - \tau_B\|_2. \quad (2)$$

In practice, (2) is approximated representing the first trajectory by a bounding box. This bounding box is extended by $\|\tau_A - \tau_B\|_2$ in $\pm x$ and $\pm y$ direction. If an element of the second street is in this box, (2) is fulfilled, as shown in Fig. 2.

V. Parameter-Substituting Global Vehicle Self-Localization

The trajectory from visual odometry is measured relative to the vehicle pose at the start of the measurement. In this section we present a scheme that employs this information together with the times and names of the observed street signs, that were detected. Thereby, the global position that corresponds to each point of the trajectory τ_t is recovered. In the following, we write λ instead of λ_A for better readability.

A. General Plane ICP with Three Optimization Parameters

The registration of one point cloud with respect to a second one can be solved with ICP, which was introduced in [2], [26]. For an overview on advanced ICP methods, we refer to [21]. Since our map is generated from OSM-data (see [19]), where the altitude of map points is rarely recorded, we perform plane matching only[4]. With this, the rotation matrix $\mathbf{R}$ with rotation α around the axis through $[0,0]^T$ and normal to the map plane emerges as

$$\mathbf{R} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix}. \quad (3)$$

Shifting the trajectory within the map is performed with the addition of $\mathbf{T} = [t_x, t_y]^T$. This means, that rotating and shifting the trajectory τ onto the map leads to three parameters $\{t_x, t_y, \alpha\}$ that have to be estimated.

[4]Our approach is not restricted to be performed in $\mathbb{R}^2$, but can be extended to process map points and trajectories in $\mathbb{R}^3$ straightforwardly.

The resulting optimization problem can be formulated as:

$$(\hat{\alpha},\hat{t}_x,\hat{t}_y) = \text{argmin}_{\mathbf{R},\mathbf{T}} \sum_{t=0}^{t_{max}} d_t(\mathbf{R},\mathbf{T})^2 \tag{4}$$

$$\text{with } d_t(\mathbf{R},\mathbf{T}) = \min_{j,p} \left\| \mathbf{R}\tau_t + \mathbf{T} - \omega_j^p \right\|_2 . \tag{5}$$

B. Optimization Parameter Space Reduction

As shown in Sec. V-A, for a plane ICP problem, three parameters have to be optimized. Combining the reconstruction of the vehicle's trajectory in the visual odometry pipeline with simultaneously detecting street signs allows to reduce the optimization parameter space dimension from $\mathbb{R}^3$ to $\mathbb{R}^1$. Jointly, the constraint that arises from a street sign being seen at a certain time is integrated into the cost function. How this is performed is in the scope of this section.

First, the optimization parameter $0 \leq \lambda \leq 1$, with $\lambda \in \mathbb{R}$ is introduced. It describes an arbitrary point on the first street, where $\lambda = 0$ stands for the beginning[5] of the street and $\lambda = 1$ for the end of the respective street. We use this point to shift the VO-trajectory on the map, so that the trajectory at the time t_A at which the first street was observed lays at the origin of its coordinate system and thereby is the center of rotation:

$$\mathbf{T}(\lambda) = \omega_A(\lambda) - \tau_A. \tag{6}$$

As introduced in Sec. IV, streets are stored as collections of ordered nodes. Connecting these ordered nodes with straight lines from one node ω_j^p to the next node ω_j^{p+1} recovers the shape of the street. This leads to a sparse representation of the information but needs to be considered in the algorithm: For a given parameter λ, the corresponding continuous coordinate on street j, $\omega_j(\lambda)$ is calculated as the interpolation between node number p_0 and p_0+1 as

$$\omega_j(\lambda) = \omega_j^{p_0} + \frac{\lambda L(\omega_j^{P_j}) - L(\omega_j^{p_0})}{L(\omega_j^{p_0+1}) - L(\omega_j^{p_0})}(\omega_j^{p_0+1} - \omega_j^{p_0}). \tag{7}$$

Index p_0 is determined with the accumulated street length $L(\omega_j^{p_k}) = \sum_{p=2}^{p_k} \|\omega_j^{p-1} - \omega_j^p\|_2$ and the total length $L(\omega_j^{P_j})$ of street j, as follows: p_0 is the highest index to fulfill $\frac{L(\omega_j^p)}{L(\omega_j^{P_j})} \leq \lambda$. Therewith, it is ensured, that the point of the trajectory at observation time of the first street t_A can be shifted continuously and always lays on the first street within the map: $\tau_A \in \omega_A$. Now, (6) and (7) can be used to substitute the shift $[t_x,t_y]^T$ by $\mathbf{T}(\lambda)$, leading to an optimization parameter reduction of one. The rotation angle α can as well be derived from λ as follows:

Since the observation times of the first street t_A and second street t_B are known, also the metric distance between those points in the map $d(\omega_A(\lambda),\omega_B(\lambda_B))$ can be calculated as

$$d(\omega_A(\lambda),\omega_B(\lambda)) = \|\tau_A - \tau_B\|_2. \tag{8}$$

[5]The determination of start and end is irrelevant, as long as it is defined.

Next, the crossing point between a circle with radius $r = d(\omega_A(\lambda),\omega_B(\lambda))$ around the λ-determined point on the first street $\omega_A(\lambda)$ and the second observed street is calculated as

$$\omega_B(\lambda_B) = \{\mathbf{x} \in \mathbb{R} : \ \|\mathbf{x} - \omega_A(\lambda)\|_2 = r\} \cap \omega_B. \tag{9}$$

Here, due to the sparse representation of the map data, $\omega_B(\lambda_B)$ has to be found by iterating through all lines, formed by ω_B^p to ω_B^{p+1} until the crossing point is found. If no unique solution can be identified, both resulting solutions are used and evaluated in parallel. The configuration with the lower error according to (4) is then selected. For simplification, we therefore assume uniqueness in the rest of the section without loss of generality. With the determined crossing point $\omega_B(\lambda_B)$ and vertical and horizontal street components $\omega_{A,x}(\lambda)$ and $\omega_{A,y}(\lambda)$, the rotation angle α can be determined in dependency of λ as follows.

$$\alpha = \underbrace{\arctan\left(\frac{\omega_{B,y}(\lambda_B) - \omega_{A,y}(\lambda)}{\omega_{B,x}(\lambda_B) - \omega_{A,y}(\lambda)}\right)}_{\alpha_{Map}} - \underbrace{\arctan\left(\frac{\tau_{B,y} - \tau_{A,y}}{\tau_{B,x} - \tau_{A,x}}\right)}_{\alpha_{VO}} \tag{10}$$

The principle of the rotation substitution is depicted in Fig. 3.

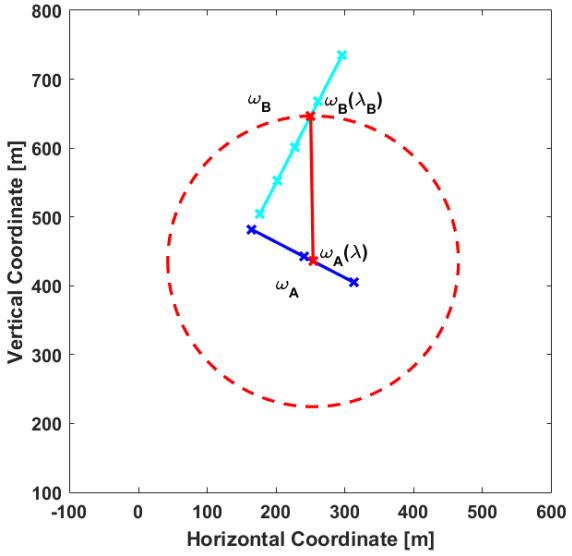

Fig. 3. The dark blue line, formed by the three dark blue crosses from the map data is the first observed street ω_A. The cyan line, formed by the five cyan crosses from the map data is the second observed street ω_B. With $\lambda = 0.6$, the position $\omega_A(\lambda)$ of the red cross on this street is determined, applying (7). The circle with the radius, calculated in (8) is drawn in dashed red. Now, the intersection point with the second observed street ω_B is determined according to (9), which is visualized as the red cross on ω_B. The red line, connecting the two red crosses can now be used to calculate the map-based rotation angle α_{Map} with (10).

Now, the new optimization parameter λ determines all three standard plane ICP parameters $\lambda \mapsto (t_x,t_y,\alpha)$, which reduces the search space and thereby speeds up the optimization. Additionally, it is ensured, that the vehicle traverses the observed streets at the observation times.

C. Overall Global Localization Process

At the beginning of our scheme, an initial map patch has to be selected. This patch can either be gained based on a very rough GPS position with an error dimension of tens of kilometers or with method from Sec. IV. Based on this initial map patch, a precise initial localization of the vehicle is performed. Therefore at least two traveled streets ω_A, ω_B and corresponding coordinates of the local trajectory τ_A, τ_B must be available. With this data, the method, described in Sec. V-B is employed to precisely and globally localize the vehicle. Further re-localization is then operated on parts of the trajectory that come from the visual odometry system. This is due to the fact, that these trajectories suffer from the accumulation of measurement errors, which leads to drift. When considering parts of the trajectory, these relative errors can be neglected. At this point, there are two possible ways for localizing the vehicle: If new information on traveled streets is gathered, the method from Sec. V-B is used again. If this is not the case, we fall back to a standard ICP, with three parameters (t_x, t_y, α) and take the end of the last subtrajectory as the initial starting point of the current one. Furthermore we restrict the map to those streets, that can be reached from the starting point with a traveled way up to the total distance of the current subtrajectory. Since the trajectory is already pre-rotated by the previous localization step, good initial estimates are available, which still leads to fast convergence of the re-localization step.

VI. Evaluation

For evaluating the proposed method, we use the visual input of track 00 of Kitti benchmark, see [10]. The local trajectory is estimated with our visual odometry system, described in Sec. III and [4], [5]. The automatic extraction of street signs is simulated by manual annotation. The used map covers an area of $\approx 1000\,km^2$ and has a size of 11 MB.

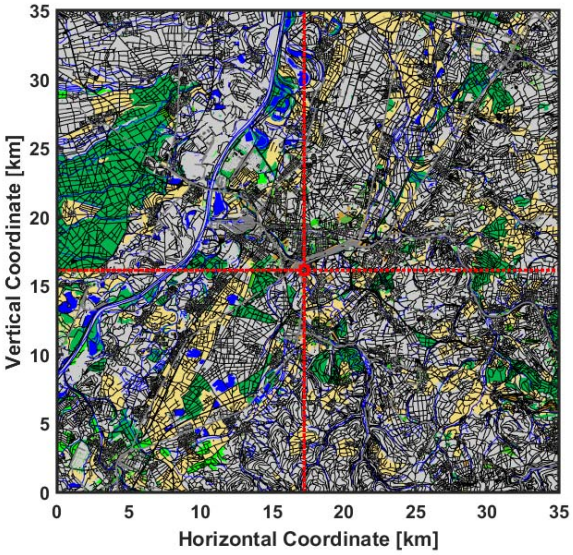

Fig. 4. In our example, we start with a map size of about $1000\,km^2$. Determining the initial map patch (red rectangle) takes <0.5 s in Matlab.

First, the initial location is determined as described in Sec. IV. The map is shown in Fig. 4 together with the resulting initial map patch, which is shown as red rectangle. After initializing the localization, the trajectory has to be registered to the map patch in the red rectangle. As explained, this is done based on the nonlinear cost function (4), using the proposed parameter substitution from Sec. V-B. Since we use Levenberg-Marquardt algorithm for the optimization, only local minima can be found, see [14], [16]. Therefore, the initial value of the optimized parameter plays a significant role, when searching for the global minimum.

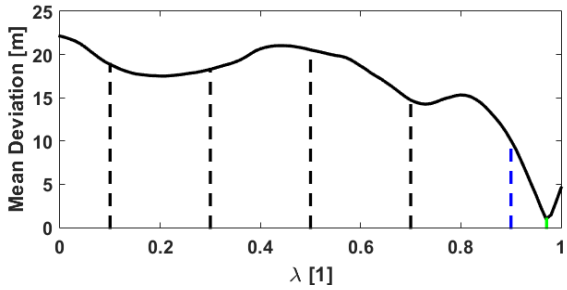

Fig. 5. The ICP cost function in dependency of λ is plotted in sold black. The rastered points from the initialization phase are visualized in dashed black. The initial value for the subsequent refinement is marked blue. The result from the refining optimization is marked green.

The mean value of all residuals d_i from (5) as a function of λ is shown in Fig. 5. As can be seen, the cost function of the considered scenario has three local minima. Until about $0.2\,\lambda$ away from the correct point on the first street $\omega_A(\lambda_{ideal})$, the function decreases monotonically to the global optimum. For robustness, we propose initial rastering with $\Delta\lambda$. The rastering result for $\lambda = 0.7$ is visualized in Fig. 6.

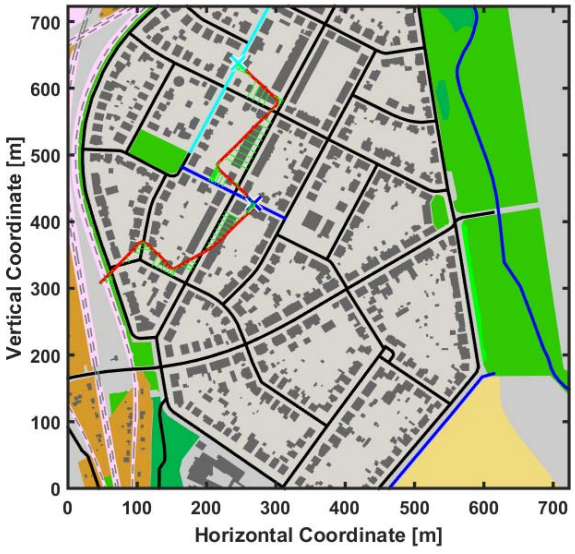

Fig. 6. Intermediate result during the rastering process. The blue solid line cross mark the first observed street ω_A and the local trajectory at the time of the street sign observation τ_A. The second street and corresponding assumed trajectory point are marked in cyan. The trajectory is plotted in red. The residuals according to (5) are plotted as green lines.

The raster point $\lambda = m\Delta\lambda$ with minimum resulting mean absolute costs is then selected as the initial value of the subsequent Levenberg-Marquardt-algorithm-based optimization. In this example, the best initial value after the raster-search is $\lambda = 0.9$, which is marked in blue.

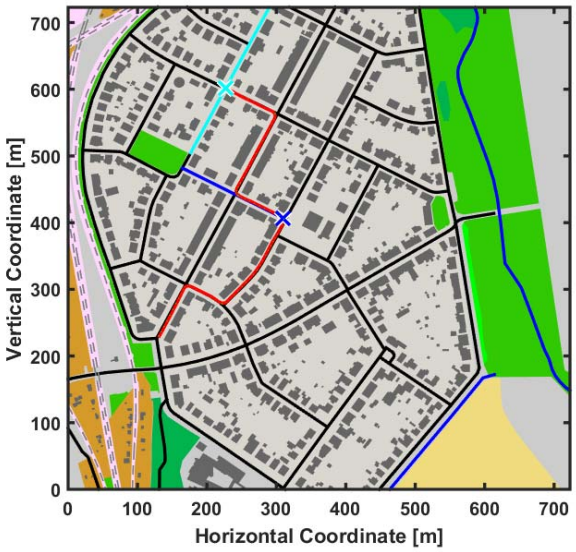

Fig. 7. Result of the overall localization: The blue solid line and cross mark the first observed street ω_A and the local trajectory τ_A at the time of observation. The second street and corresponding assumed trajectory point are marked in cyan. The trajectory is plotted in red. As can be seen, the optimization converges to the optimum. This means, that the trajectory is properly registered to the map. The mean residual distance between trajectory and the closest drivable street is about 1 m.

Taking this as a starting point, the refinement results in $\lambda \approx 0.97$. The registered position of the visual odometry trajectory relative to the map for this λ is shown in Fig. 7. The mean residual distance between the trajectory and the nearest drivable street is $\overline{d} \approx 1\,\text{m}$. This remaining error is caused by simplification of the road structure in the map data as e.g. multi-lane streets mostly being represented by a single center line. Now, the alignment between the current trajectory and map is known. The optimization step is repeated for all following trajectory parts that are calculated by the visual odometry system – optionally using the knowledge on new recognized street signs, see Sec. V-C.

VII. Conclusion and Future Work

We presented a method for efficient vehicle global self-localization and proved the successful application to real visual odometry trajectories and map data. Until now, the automatic detection of street signs was simulated, since our scope is to present the concept of this method. In a subsequent project, this detection has to be implemented. Also the extraction of further landmarks as the name of shops and the integration of surround view cameras are in that scope. A further topic to be investigated, is the model-based map-refinement as performed in [3], [6], [13] and the resulting effects on self-localization with lane-level precision.

Acknowledgments

We kindly thank Continental AG for funding this work within a cooperation.

References

[1] A. Basiri et al. Overview of positioning technologies from fitness-to-purpose point of view. In *2014 IEEE International Conference on Localization and GNSS*, 2014.
[2] P. Besl and N. McKay. Method for registration of 3-d shapes. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992.
[3] M. Brubaker et al. Probabilistic map localization through visual odometry. In *Proceedings of Scene Understanding Workshop at IEEE Conference on Computer Vision and Pattern Recognition*, 2013.
[4] M. Buczko and V. Willert. Flow-decoupled normalized reprojection error for visual odometry. In *19th International IEEE Conference on Intelligent Transportation Systems*, 2016.
[5] M. Buczko and V. Willert. How to distinguish inliers from outliers in visual odometry high-speed automotive applications. In *IEEE Intelligent Vehicles Symposium*, 2016.
[6] G. Cao et al. Camera to map alignment for accurate low-cost lane-level scene interpretation. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems*, 2016.
[7] J. Chao et al. An experimental investigation into the performance of gps-based vehicle positioning in very dense urban areas. 2001.
[8] I. Cvišić and I. Petrović. Stereo odometry based on careful feature selection and tracking. In *European Conference on Mobile Robots*, 2015.
[9] U. Franke et al. Making bertha see. In *IEEE International Conference on Computer Vision Workshops*, 2013.
[10] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.
[11] D. Hahnel et al. Mapping and localization with rfid technology. In *2004 IEEE International Conference on.Robotics and Automation*, 2004.
[12] J. Košecká et al. Global localization and relative positioning based on scale-invariant keypoints. In *Robotics and Autonomous Systems 52.1*, 2005.
[13] F. Kuhnt et al. Particle filter map matching and trajectory prediction using a spline based intersection model. In *2014 IEEE 17th International Conference on Intelligent Transportation Systems and corresponding FZI-Report*, 2014.
[14] K. Levenberg. A method for the solution of certain problems in least squares, 1944.
[15] M. Lu et al. Positioning and tracking construction vehicles in highly dense urban areas and building construction sites. In *Automation in Construction 16.5*, 2007.
[16] D. Marquardt. An algorithm for the least-squares estimation of nonlinear parameters, 1963.
[17] O. Mezentsev et al. Vehicular navigation in urban canyons using a high sensitivity gps receiver augmented with a low cost rate gyro. In *ION GPS*, 2002.
[18] R. Minetto et al. Snoopertext: A text detection system for automatic indexing of urban scenes. In *Computer Vision and Image Understanding 122*, 2014.
[19] OpenStreetMap contributors. Planet dump [data file from [2/2017]. retrieved from http://planet.openstreetmap.org, 2015.
[20] I. Parra et al. Visual odometry and map fusion for gps navigation assistance. In *2011 IEEE International Symposium on Industrial Electronics*, 2011.
[21] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Third International Conference on 3-D Digital Imaging and Modeling*, 2001.
[22] D. Scaramuzza and F. Fraundorfer. Visual odometry [tutorial]. In *IEEE robotics and automation magazine 18.4*, 2011.
[23] S. Se et al. Vision-based global localization and mapping for mobile robots. In *IEEE Transactions on robotics 21.3*, 2005.
[24] P. Sermanet and Y. LeCun. Traffic sign recognition with multi-scale convolutional networks. In *2011 International Joint Conference on Neural Networks*, 2011.
[25] Statistisches Bundesamt. Verkehrsunfälle, zeitreihen. 2016.
[26] C. Yang and G. Medioni. Object modelling by registration of multiple range images. In *Image and vision computing 10.3*, 1992.
[27] Z. Zhu et al. Real-time global localization with a pre-built visual landmark database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.