HD Map Errors Detection using Smoothing and Multiple Drives

Anthony Welte, Philippe Xu, Philippe Bonnifait, Clément Zinoune

HD Map Errors Detection using Smoothing and Multiple Drives

Anthony Welte[1], Philippe Xu[1], Philippe Bonnifait[1], and Clément Zinoune[2]
[1] Heudiasyc, Université de Technologie de Compiègne, CNRS UMR 7253, France
[2] Renault S.A.S

***Abstract*— High Definition (HD) maps enable autonomous vehicles to not only navigate roads but also localize. Using perception sensors such as cameras or lidars, map features can be detected and used for localization. The accuracy of vehicle localization is directly influenced by the accuracy of the features. It is therefore essential for the localization system to be able to detect erroneous map features. In this paper, an approach using Kalman smoothing with observation residuals is presented to address this issue. A covariance intersection of the residuals is proposed to manage their unknown correlation. The method also leverages the information of multiple runs to improve the detection of small errors. The performance of the method is evaluated using experimental data recorded on public roads with erroneous road signs. Our results allow to evaluate the gain of detection brought during successive drives.**

I. Introduction

High accuracy maps offer the possibility to greatly improve the accuracy of localization systems [1], [2]. By detecting features of the environment that are accurately referenced in a HD map, a localization system can correct the vehicle pose. However, this assumes that the map can be trusted. Although road networks do not change quickly, maps can still become outdated. Errors can be made during the mapping process, features can be damaged or moved because of traffic accidents and roadworks can change an entire section of road for instance. It is therefore essential for a reliable localization system to be able to detect changes in the map.

In this work, the problem of detecting road signs with inaccurate positions is addressed. While missing features lead to fewer measurements, thus worse localization, incorrect referencing of existing features lead to wrong measurements. This induces an erroneous localization solution and overconfidence in the estimated solution. In this work, road signs whose positions have an error smaller than one meter are considered. Detecting large faults is not a difficult problem. It is the small and medium faults that are the most challenging.

In the next section, the existing literature on map error detection is highlighted. Section III details the localization framework used in this work. The method used to detect map errors is explained in Section IV. Our approach uses a localization system that generates residuals of the detected features stored in the HD map through a smoothing method. Finally, Section V presents experimental results carried out with multiple drives recorded on open roads.

II. Related Work

The widespread use of maps for intelligent vehicles has led to questions about the accuracy of these maps [3] and to the development of methods to detect and correct map errors. There are different types of map errors leading to widely different strategies to detect them.

An important element of the map that needs to be accurate for intelligent vehicles is road geometry. The road geometry changes slowly as there are few ways to affect it besides roadworks. Errors on this aspect of the map would have disastrous effects for control and planning systems.

Hartmann *et al*. [4] have trained a neural network to detect maps errors. To detect errors, the network is fed with the difference between observed road characteristics (distance to lane markings, curvature, etc.) and characteristics obtained from a map. With this information, the network outputs a probability of error. The authors show that this approach is able to detect errors such as missing roundabouts or intersections. It is also important to note that unlike many map error detection methods, this approach is able to detect changes before the vehicle goes through an area using a front-facing camera.

Although some have used the general road geometry for localization purposes (e.g. in [5]), specific accurately referenced features are more useful for localization. Hence, research has also been done to address this problem.

Point clouds are also used as georeferenced data in so-called dense maps. Even features maps used in this work have been produced by cartographer using dense point clouds of the area. Hence, detecting map changes can be made using this format. The authors in [6] compared point cloud maps using a point to point approach. Points from one map version are classified based on their distance to points from another map version. New points are far from any points of the old map and points from the old version no longer existing are too far from any point in the new map. This method requires the original dense map format to be kept which raises scalability issues as those types of map are heavy.

The HD map change detection problem has been addressed in [7] using a dual particle filtering approach. Two particle filters are used, one estimate the vehicle state without using the map, the other uses lane marking measurements. From both sets of particles, descriptors such as the mean lane marking innovation and the mean particle weights are computed. Features are classified based on these descriptors using thresholding techniques.

Nguyen *et al*. [8] have also used an estimation with and without map features to figure out if the map is reliable. The authors compare estimates obtained using map features for localization to dead-reckoning estimates. A random forest

classifier trained to classify features as reliable or not is then used to infer the features reliability.

Evaluating map features is not only limited to evaluating the accuracy of their positions. Some have chosen the strategy of trying to verify that a feature exists at a known position. A feature persistence model has been developed in [9] to account for the possibility of features disappearing. The authors in [10] have proposed an approach using Dempster-Shafer theory to deal with this issue. In this framework, features that have moved will be classified as *deleted* and a new feature will be added at the correct position.

In contrast to the existing body of research, our method aims at detecting position changes, using states estimated using post-smoothing residuals. The proposed method can be applied by simply storing the states and observations used in the widely used Kalman filtering scheme.

III. Estimation Framework

A. General Architecture

To perform road signs error detection, the proposed method relies on localization estimates. The real time localization solution is used in a post-processing fashion to evaluate the mapped features.

The localization framework used in this work relies on three components as shown in Figure 1. The main component is the estimation layer that performs the actual estimation of the vehicle states. This layer relies on a Kalman filtering scheme to provide high frequency estimates. It fuses information obtained from the vehicle proprioceptive sensors, the GNSS receiver, lane marking and road sign measurements. The vehicle is modeled using a Constant Turn Rate and Velocity (CTRV) model. Both the estimated states (denoted as $\boldsymbol{x}_k$) and observations (denoted as $\boldsymbol{z}_k$) are buffered such that delays can be accounted for. The observations are related to the states through the observation model ($\boldsymbol{z} = h(\boldsymbol{x})$). This component does not perform data association of measurements to map features. The association is instead performed in parallel in a matching layer. The matching layer periodically acquires the state and observation buffers of the estimation layer and performs a global association of detection to map features. Once the matches have been found, they are shared to the estimation layer to be used to estimate the states. The matching method is detailed more in [11]. Finally, a third component performs post processing tasks. It records the measurements and states estimated during a drive. At the end of a drive this data is used to perform calibration tasks [12] and evaluate the map using the method presented in this paper.

B. Road Sign Measurements

In this work, errors on road signs coordinates are being detected. Unlike some methods that rely on independent estimates and measurements to detect map errors, this work instead aims at detecting errors on features used during the estimation. It is therefore essential to detail the observation model used to include road sign measurements in the estimation.

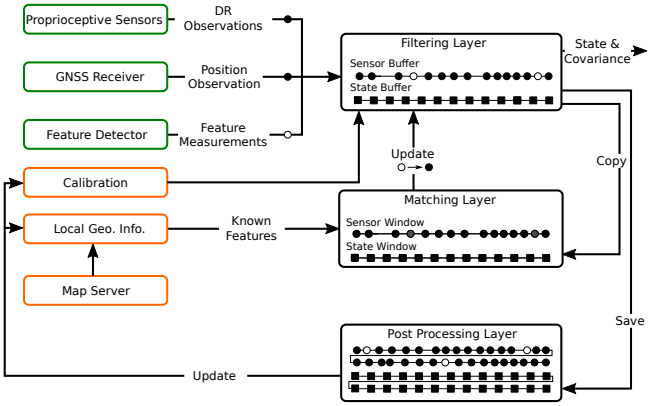


Fig. 1: Diagram of the localization architecture. The Filtering Layer performs the actual state estimation at high frequency, the Matching Layer associate measurements to map features using the state and observation buffer from the Filtering Layer, the Post Processing Layer records the observations and states to detect map errors.

The road sign detector used in this work relies on lidar intensity measurements to detect road signs. Indeed, road signs are designed to be retro-reflective to easily be seen at night. This also makes them easily detectable using the lidar intensity measurements. The road signs are detected by filtering the lidar point cloud using a high-intensity threshold. The remaining points are then clustered using Euclidean clustering. The centroid $\begin{bmatrix} {}^M x_k^j & {}^M y_k^j \end{bmatrix}^\top$ of each cluster j is a measurement that can be used for state estimation (^{M} indicates that the coordinates are expressed in the mobile frame attached to the vehicle).

As will be explained in Section IV the error detection method relies on observation residuals (the difference between the actual observation and the expected observation: $\boldsymbol{z}_k - h(\boldsymbol{x}_k)$). Even after a single drive, multiple residuals will be obtained for a map feature (one for each observation). A method to merge those residuals is proposed in Section IV. However, not every type of residual can be merged. The residuals need to be expressed in the same observation frame. Indeed if the measured road sign positions (in the sensor frame) were used as measurements, the residuals will correspond to errors in the sensor frame. This frame changes depending on the pose of the vehicle at every instant. An erroneous road sign might have positive residuals when it is observed in one direction but negative residuals when it is observed in the opposite direction. The residuals would then cancel each other make the error hard to detect.

To avoid this the observation model needs to be chosen such that the residuals it produces do not depend on the vantage point. For that reason, the observation used for the state estimation is not the local measured road sign position but is instead the global map reference. The observation model h is built using the measured local position $\begin{bmatrix} {}^M x_k^j & {}^M y_k^j \end{bmatrix}^\top$ moved to the global frame using the vehicle position and

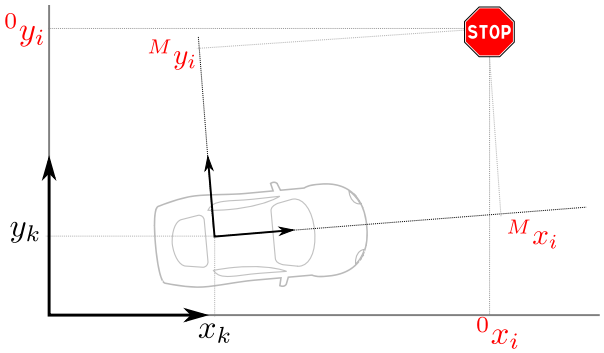


Fig. 2: Graphic of the road sign observation model. The road sign measurement in the vehicle frame M can be transformed to the global frame 0 using the vehicle pose.

orientation, as described in equation 2.

$$\boldsymbol{z}_k^j = \begin{bmatrix} x_i \\ y_i \end{bmatrix} = h\left(\boldsymbol{x}_k\right) \tag{1}$$

$$= \begin{bmatrix} \cos\theta_k & -\sin\theta_k \\ \sin\theta_k & \cos\theta_k \end{bmatrix} \begin{bmatrix} {}^Mx_k^j \\ {}^My_k^j \end{bmatrix} + \begin{bmatrix} x_k \\ y_k \end{bmatrix} \tag{2}$$

It is important to note that to simplify the equation, the measurements are assumed to be obtained directly in the vehicle frame. In practice the transformation between the sensor and vehicle frame needs to be well calibrated.

The covariance matrix $\boldsymbol{R}$ of this observation model can be evaluated by comparing the actual measurements with the predicted measurements from the observation model using a ground truth states.

IV. Map Error Detection

A. Single Drive

1) Kalman Smoothing: The error detection method relies on a Kalman filtering scheme for the state estimation. Kalman filtering enables high frequency estimation but, unlike graph based methods [13], it only estimates the state $\boldsymbol{x}_k$ using the observations $\boldsymbol{z}_1$ to $\boldsymbol{z}_k$. Hence, a state estimate does not benefit from future measurements. This can be mitigated using Kalman smoothing. Kalman smoothing, also known as Rauch–Tung–Striebel (RTS) smoothing, enables to propagate the new information from future measurements to past states. This is not done using the future observations but rather using only the states. Since future states have been estimated using future observations, the information is contained in the state estimates. With this method, the buffer of filtered $\left\{\hat{\boldsymbol{x}}_{k|k}, \boldsymbol{P}_{k|k} \,|k \in [\![0, N]\!]\right\}$ and predicted $\left\{\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{P}_{k|k-1} \,|k \in [\![1, N]\!]\right\}$ states estimates can be smoothed sequentially in a backward pass from time N to time 0. This is done using

$$\hat{\boldsymbol{x}}_{k|N} = \hat{\boldsymbol{x}}_{k|k} + \boldsymbol{J}_k\left(\hat{\boldsymbol{x}}_{k+1|N} - \hat{\boldsymbol{x}}_{k+1|k}\right) \tag{3}$$

$$\boldsymbol{P}_{k|N} = \boldsymbol{P}_{k|k} + \boldsymbol{J}_k\left(\boldsymbol{P}_{k+1|N} - \boldsymbol{P}_{k+1|k}\right)\boldsymbol{J}_k^\top \tag{4}$$

with $\boldsymbol{J}_k$ expressed with the evolution model Jacobian matrix $\boldsymbol{F}_{k+1}$ as

$$\boldsymbol{J}_k = \boldsymbol{P}_{k|k}\boldsymbol{F}_{k+1}^\top \boldsymbol{P}_{k+1|k}^{-1} \tag{5}$$

2) Covariance Intersection: To evaluate map features, residuals are used. Map features can be observed multiple times. For a single drive, multiple residuals are computed. After having smoothed the state estimates, the residuals $\boldsymbol{y}_{k|N}^j$ (and corresponding covariance matrices $\boldsymbol{S}_{k|N}^j$) of each observation are computed as follows [14],

$$\boldsymbol{y}_{k|N}^j = \boldsymbol{z}_k^j - h\left(\hat{\boldsymbol{x}}_{k|N}\right) \tag{6}$$

$$\boldsymbol{S}_{k|N}^j = \boldsymbol{R}_k - \boldsymbol{H}_k\boldsymbol{P}_{k|N}\boldsymbol{H}_k^\top \tag{7}$$

with $\boldsymbol{R}_k$ being the covariance matrix of the observation, $\boldsymbol{H}_k$ the Jacobian of the observation model and $\boldsymbol{P}_{k|N}$ the covariance matrix of the smoothed state estimate.

It is important to note some observation models are not suited to be used with this method. This method combines multiple residuals from multiple observations and multiple drives. This can be done only if the residuals can be compared. In Section III, the observation model chosen for road sign measurement was chosen such that the observation (and thus the residuals) are expressed in the global reference frame. With this observation model, all residuals can be interpreted as the road sign error in the global frame. It is for that reason that this method is not applied to lane markings. The lane marking observation model as stated in [15] would result in residuals depending on the point of view (e.g. positive residuals when observing a marking driving East to West but negative residuals when driving West to East).

At the end of a drive, a subset $\mathcal{M} = \{\boldsymbol{m}_i\}_{i\in[\![0,M]\!]}$ of map features has been observed. For every map feature $\boldsymbol{m}_i$, residuals have been obtained $\left\{\boldsymbol{y}_{k_0}, \boldsymbol{y}_{k_1}, \ldots, \boldsymbol{y}_{N_i}\right\}$ (where N_i is the number of residuals for feature $\boldsymbol{m}_i$). These residuals are correlated. Indeed, they are all computed using smoothed state estimates that are themselves correlated. Therefore, the residuals of features of the same trajectory are correlated.

To decide whether a feature should be used or not, an aggregate residual is used. Because the residuals are correlated standard fusion methods cannot be used. Instead, the aggregated residual $\boldsymbol{y}_i$ is obtained using a covariance intersection strategy. Covariance intersection enables fusion of multiple information with unknown correlation. It consists in a linear interpolation between the different sources of information weighted to minimize the resulting covariance matrix. In this application, a feature $\boldsymbol{m}_i$ has the N_i following residuals

$$\boldsymbol{y}_{k_0}, \boldsymbol{y}_{k_1}, \ldots, \boldsymbol{y}_{N_i} \tag{8}$$

with corresponding covariance matrices

$$\boldsymbol{S}_{k_0}, \boldsymbol{S}_{k_1}, \ldots, \boldsymbol{S}_{N_i} \tag{9}$$

The aggregate residuals are expressed as

$$\begin{aligned} \boldsymbol{S}_i^{-1} &= \omega_{k_0}\boldsymbol{S}_{k_0}^{-1} + \omega_{k_1}\boldsymbol{S}_{k_1}^{-1}, \ldots, +\omega_{N_i}\boldsymbol{S}_{N_i}^{-1} \\ \boldsymbol{y}_i &= \boldsymbol{S}_i\left(\omega_{k_0}\boldsymbol{S}_{k_0}^{-1}\boldsymbol{y}_{k_0} + \omega_{k_1}\boldsymbol{S}_{k_1}^{-1}\boldsymbol{y}_{k_1}, \ldots, +\omega_{N_i}\boldsymbol{S}_{N_i}^{-1}\boldsymbol{y}_{k_{N_i}}\right) \end{aligned} \tag{10}$$

where $\sum_{k\in\{k_0,\ldots,N_i\}}\omega_k = 1$.

In the general case, the parameters ω_k have to be found using iterative estimation methods. For small value of the

dimension (five dimensions when minimizing $det(\boldsymbol{S}_i)$ and four dimensions when minimizing $trace(\boldsymbol{S}_i)$) analytic methods exist [16] to find the best parameter ω when intersecting two residuals $\boldsymbol{y}_{k_0}$ and $\boldsymbol{y}_{k_1}$. For that reason the covariance intersection is not performed globally using all residuals at once. Instead the residuals are intersected two by two until all residuals associated to a map feature have been processed.

Hence the covariance intersection is solved sequentially for each feature as shown in Algorithm 1.

Algorithm 1 Covariance Intersection

$\boldsymbol{y} \leftarrow \boldsymbol{y}_{k_0}$
$\boldsymbol{S} \leftarrow \boldsymbol{S}_{k_0}$
for $k \in [\![k_1, k_{N_i}]\!]$ **do**
  $\omega \leftarrow \arg\min_{\omega \in [0,1]} \det\left(\omega \boldsymbol{S}^{-1} + (1-\omega)\boldsymbol{S}_k^{-1}\right)^{-1}$
  $\boldsymbol{S}' \leftarrow \left(\omega \boldsymbol{S}^{-1} + (1-\omega)\boldsymbol{S}_k^{-1}\right)^{-1}$
  $\boldsymbol{y} \leftarrow \boldsymbol{S}'\left(\omega \boldsymbol{S}^{-1}\boldsymbol{y} + (1-\omega)\boldsymbol{S}_k^{-1}\boldsymbol{y}_k\right)$
  $\boldsymbol{S} \leftarrow \boldsymbol{S}'$
end for
return $\boldsymbol{y}, \boldsymbol{S}$

To find the optimal value of ω, the algorithm presented in [16] is used.

B. Multiple Drives

While residuals of the same trajectory are correlated, residuals from different trajectories are not. Indeed, the residuals from the same trajectory are correlated because all states are correlated through the evolution model. The states from different trajectories should be uncorrelated leading to uncorrelated residuals. Therefore, multiple trajectories can be used to further reduce the uncertainty of residuals.

A residual $\bar{\boldsymbol{y}}$ with corresponding covariance matrix $\bar{\boldsymbol{S}}$ known from previous drives can be updated using the new residual $\boldsymbol{y}$ (with associated covariance matrix $\boldsymbol{S}$) obtained as described in algorithm 1 using the following equations,

$$\bar{\boldsymbol{y}} \leftarrow \left(\bar{\boldsymbol{S}}^{-1} + \boldsymbol{S}^{-1}\right)^{-1}\left(\bar{\boldsymbol{S}}^{-1}\bar{\boldsymbol{y}} + \boldsymbol{S}^{-1}\boldsymbol{y}\right) \tag{11}$$

$$\bar{\boldsymbol{S}} \leftarrow \left(\bar{\boldsymbol{S}}^{-1} + \boldsymbol{S}^{-1}\right)^{-1} \tag{12}$$

The updated residual can finally be used to evaluate the feature.

C. Eliminating features

If the road sign is correctly referenced, its residual should be centered. Therefore, as new observations are obtained and new drives are performed, the residual should tend to 0 as the uncertainty of the residual decreases. However, erroneous features will result in residuals not converging toward 0 but covariance matrices still getting smaller. Hence, at some point, the residual will be too high with respect to the accuracy that would be expected from the covariance matrix.

To detect these cases, a χ^2 test is used. After the residuals are computed, the inequality from equation (13) is performed. Features failing this test are flagged as erroneous to be ignored in future estimations.

$$\bar{\boldsymbol{y}}\bar{\boldsymbol{S}}^{-1}\bar{\boldsymbol{y}}^T < F_{\chi_2^2}^{-1}(1-\alpha) \tag{13}$$

where $F_{\chi_2^2}$ is the cumulative distribution function of a χ^2 distribution with 2 degrees of freedom and α is a tuning parameter controlling the number of correct features rejected.

Fig. 3: Experimental vehicle of the Heudiasyc laboratory used in these experiments. It is equipped with a low-cost GNSS receiver, a Mobileye camera, and a Velodyne VLP-32C lidar. Ground truth is provided by a post-processed IMU with PPK GNSS.

TABLE I: Length and duration of the Compiègne datasets.

Name	Length	Duration
Between Roundabouts	9.50 km	17 min 30 s
Laboratory to BF 1	7.03 km	11 min 42 s
Laboratory to BF 2	7.03 km	12 min 54 s
Large Loop 1	12.85 km	14 min 43 s
Large Loop 2	12.85 km	14 min 37 s
Large Loop 3	12.84 km	14 min 17 s

V. Experimental Results

A. Experimental Setup

The method presented in the previous section has been tested using experimental data. The data was recorded in Compiègne (France) using the experimental vehicle of the Heudiasyc laboratory (see Figure 3). The sensor measurements were obtained using the ROS framework. The method has been tested on an Intel© Core™ i7-7820HQ processor (2.9 GhZ, 4 cores, 8 threads) with 16 GB of RAM. With this hardware the localization was performed in real time (50 Hz) and the post processing takes only a few seconds to run. Multiple drives were recorded using the same experimental vehicle. It can be noted that, although in those experiments the same vehicle was used, the method can be applied to different vehicles. On the whole, six drives totaling 55 km have been used in these experiments (detailed in Table I). The drives followed three different trajectories in Compiègne (see Figure 4).

The HD map of the area was provided by a third party map provider. The map does not have enough incorrect feature to evaluate the detection method. For that reason, errors were

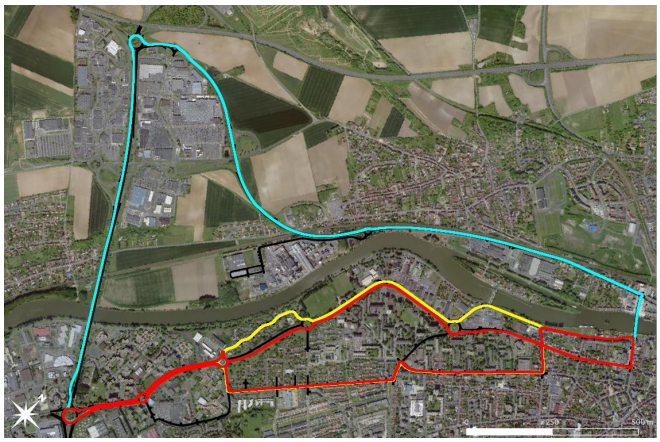


Fig. 4: Trajectories used in the experiments. In red is the first trajectory "Between Roundabouts", in yellow is the trajectory used for the drives "Laboratory to BF" and in cyan is the trajectory corresponding to the drives "Large Loop".

simulated in these experiments. Since the roads of the first trajectory are also used for the second and third trajectory, errors were added on road signs observed during the first drive. In this first drive, 207 road signs are observed. On 20 randomly chosen of those signs, an error was artificially added. The error was taken uniformly distributed within $[-1\text{m}\ , 1\text{m}]$ for both dimensions. The distribution of error is shown in Figure 6. The parameter α was chosen at 5%.

B. Error Detection

From Table II, it can be seen from the first column that after a single drive few erroneous features, 2 out of 20, have been found. However, as more drives are done, most matched erroneous features are correctly flagged (detected erroneous): 8 out of 11 (Flagged faulty in Table II).

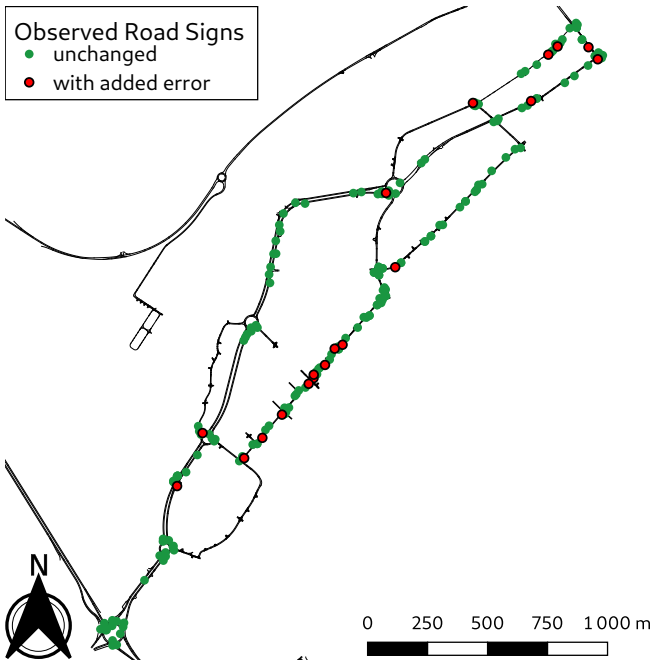
Some correct road signs are being flagged, which is unavoidable. On average 7% of correct road signs are flagged, which is well in accordance with the expected value α.

The number of unmatched (measurements that could not be matched with any map feature) correct road signs decreases as more drives are done which is a normal behavior. Meanwhile the number of unmatched faulty road signs remains constant as they might not be matched because of their errors. These unmatched features are actually detected and measurements are available since the list of features was chosen among the detected features of the first drive. However, they are not used in the estimation or the feature evaluation (since only matched features can be evaluated).

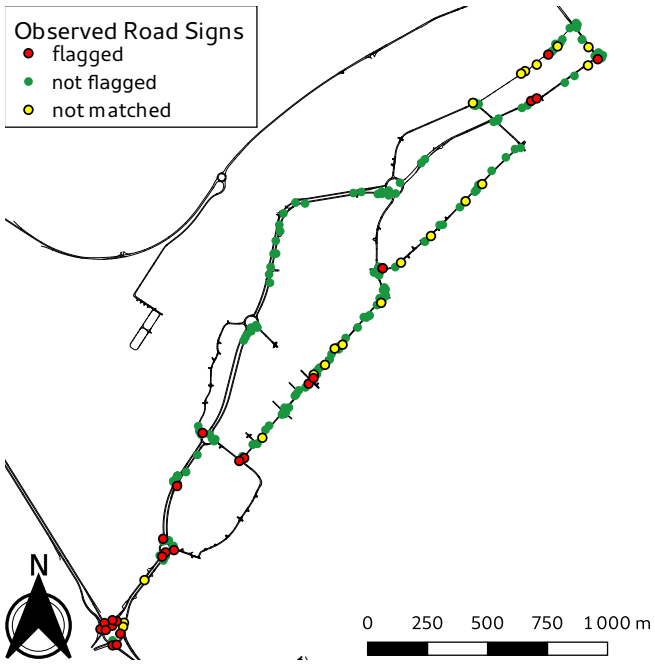
Most of the gain of using multiple drives is obtained after the first three drives, 75% of flagged faulty signs are detected. The last three drives only provide marginal improvements with only two additional faulty road signs flagged.

C. Influence of the Error Magnitude

The error added to the road signs being chosen uniformly between -1 and 1 meter in both dimensions, some road signs have small errors. On Figure 6, it is shown that road signs with errors higher than 0.5 m are either not observed (not matched) or flagged. The three faulty road signs that



(a)



(b)

Fig. 5: (a) shows the map with the 207 road signs studied in these experiments. The green dots indicate unaffected road signs and the red dots road signs with an error. (b) shows the result of the error detection after the last drive. The green dots show that no error has been found the red dots are classified as erroneous, the yellow dots have not been observed (the matching layer did not match these measurements to any feature).

TABLE II: Evolution of the categorization of road signs. The flagged road signs have been matched and detected as faulty. The first three lines correspond to road signs with no added error (green dots in Figure 5 (a)) the last three lines are the 20 road signs with added errors.

		first	second	third	fourth	fifth	sixth
Not matched	(ok)	28	16	15	14	14	11
Not flagged	(ok)	147	162	161	161	161	159
Flagged	(ok)	12	9	13	12	12	17
Not matched	(faulty)	10	10	10	9	9	9
Not flagged	(faulty)	8	7	4	4	4	3
Flagged	(faulty)	2	3	6	7	7	8

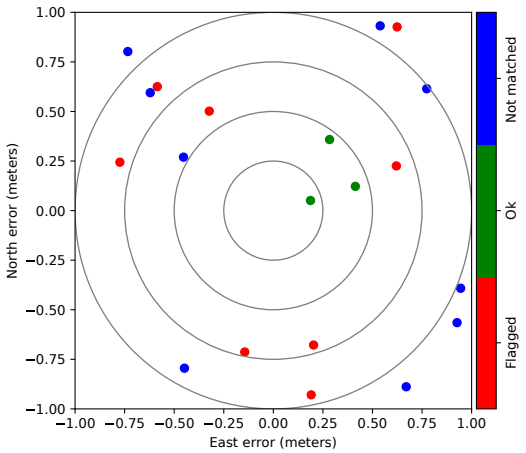


Fig. 6: Distribution of the errors added to the road signs. The colors indicate whether the error was detected (red), it was never matched (blue) or the error was missed (green).

remain not flagged are those with smaller errors. Errors too small would require more drives to be detected. Also, the number of road signs not observed when the errors are added, even though they were observed before, shows that the matching is able to discard some erroneous road signs during localization.

VI. Conclusion

In this paper, a map feature evaluation method has been presented. The method uses information from the localization system to evaluate the quality of map features. It uses a standard Kalman filtering scheme and can therefore be applied without significant modifications to different estimation frameworks. It only requires the localization system to have enough redundancy as to not rely on single (possibly faulty) observations. The method uses multiple residuals from a drive combined using Covariance Intersection. Different drives can also be leveraged to further refine the estimation of the residuals, thus improving the error detection. The features are then flagged using a χ^2 test. The method has been evaluated using experimental data with map errors. After several drives, the method is able to correctly detect over 70 % of erroneous road signs used in the estimation. Errors large enough are either detected or the observation was already removed in the matching step. Undetected errors only remain for road signs with errors below 0.5 m that would require more drives to identify.

In this work, the method has been applied to maps expected to be perfectly accurate but the method could be adapted to account for map uncertainties. This approach can also be used with multiple vehicles in order to crowdsource the detection of map errors.

Acknowledgment

This work was carried out within SIVALab, a shared laboratory between Renault and Heudiasyc (UTC/CNRS).

References

[1] M. Schreiber, C. Knoppel, and U. Franke, "LaneLoc: Lane marking based localization using highly accurate maps," in *IEEE Intelligent Vehicles Symp. (IV)*, Australia, Jun. 2013, pp. 449–454.

[2] 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 Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 582–597, Feb. 2018.

[3] J.-H. Pauls, T. Strauss, C. Hasberg, M. Lauer, and C. Stiller, "Can We Trust Our Maps? An Evaluation of Road Changes and a Dataset for Map Validation," in *IEEE 21st International Conference on Intelligent Transportation Systems (ITSC)*, Maui, HI, Nov. 2018, pp. 2639–2644.

[4] O. Hartmann, M. Gabb, R. Schweiger, and K. Dietmayer, "Towards autonomous self-assessment of digital maps," in *IEEE Intelligent Vehicles Symposium (IV)*, MI, USA, Jun. 2014, pp. 89–95.

[5] C. Fouque, P. Bonnifait, and D. Betaille, "Enhancement of global vehicle localization using navigable road maps and dead-reckoning," in *IEEE/ION Position, Location and Navigation Symposium*, Monterey, CA, USA, 2008, pp. 1286–1291.

[6] J. Hyyppa, A. Jaakkola, H. Hyyppa, H. Kaartinen, A. Kukko, M. Holopainen, L. Zhu, M. Vastaranta, S. Kaasalainen, A. Krooks, P. Litkey, P. Lyytikainen-Saarenmaa, L. Matikainen, P. Ronnholm, R. Chen, Y. Chen, A. Kivilahti, and I. Kosonen, "Map updating and change detection using vehicle-based laser scanning," in *Joint Urban Remote Sensing Event*, Shanghai, China, May 2009, pp. 1269–1274.

[7] D. Pannen, M. Liebner, and W. Burgard, "HD Map Change Detection with a Boosted Particle Filter," in *IEEE International Conference on Robotics and Automation (ICRA)*, Canada, May 2019, pp. 2561–2567.

[8] T. T. Nguyen, J. Spehr, M. Uhlemann, S. Zug, and R. Kruse, "Learning of lane information reliability for intelligent vehicles," in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, Germany, Sep. 2016, pp. 142–147.

[9] D. M. Rosen, J. Mason, and J. J. Leonard, "Towards lifelong feature-based mapping in semi-static environments," in *IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, Sweden, May 2016, pp. 1063–1070.

[10] K. Jo, C. Kim, and M. Sunwoo, "Simultaneous Localization and Map Change Update for the High Definition Map-Based Autonomous Driving Car," *Sensors*, vol. 18, no. 9, pp. 3145–3160, Sep. 2018.

[11] A. Welte, P. Xu, P. Bonnifait, and C. Zinoune, "Improved data association using buffered pose adjustment for map-aided localization," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6334–6341, 2020.

[12] A. Welte, P. Xu, and P. Bonnifait, "Four-Wheeled Dead-Reckoning Model Calibration using RTS Smoothing," in *IEEE International Conference on Robotics and Automation (ICRA)*, Montreal, QC, Canada, May 2019, pp. 312–318.

[13] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216–235, Feb. 2012.

[14] R. G. Gibbs, "New Kalman filter and smoother consistency tests," *Automatica*, vol. 49, no. 10, pp. 3141–3144, Oct. 2013.

[15] A. Welte, P. Xu, P. Bonnifait, and C. Zinoune, "Estimating the reliability of georeferenced lane markings for map-aided localization," in *IEEE Intelligent Vehicles Symposium (IV)*, Paris, France, Jun. 2019, pp. 1225–1231.

[16] M. Reinhardt, B. Noack, and U. D. Hanebeck, "Closed-form optimization of covariance intersection for low-dimensional matrices," in *15th International Conference on Information Fusion*, Singapore, Jul. 2012, pp. 1891–1896.