LIDAR-Based road signs detection For Vehicle Localization in an HD Map

Farouk Ghallabi, Ghayath El-Haj-Shhade, Marie-Anne Mittet, Fawzi Nashashibi

LIDAR-Based road signs detection For Vehicle Localization in an HD Map

Farouk Ghallabi*[†], Ghayath EL-HAJ-SHHADE[†], Marie-Anne MITTET[†]
and Fawzi Nashashibi*

*INRIA Paris-Rocquencourt
Paris, France
{farouk.ghallabi, fawzi.nashashibi}@inria.fr,

[†]Renault s.a.s
Guyancourt, France
{ghayath.el-haj-shhade, marie-anne.n.mittet}@renault.com

Abstract—Self-vehicle localization is one of the fundamental tasks for autonomous driving. Most of current techniques for global positioning are based on the use of GNSS (Global Navigation Satellite Systems). However, these solutions do not provide a localization accuracy that is better than 2-3 m in open sky environments [1]. Alternatively, the use of maps has been widely investigated for localization since maps can be pre-built very accurately. State of the art approaches often use dense maps or feature maps for localization. In this paper, we propose a road sign perception system for vehicle localization within a third party map. This is challenging since third party maps are usually provided with sparse geometric features which make the localization task more difficult in comparison to dense maps. The proposed approach extends the work in [2] where a localization system based on lane markings has been developed. Experiments have been conducted on a Highway-like test track using GNSS/INS with RTK corrections as ground truth (GT). Error evaluations are given as cross-track and along-track errors defined in the curvilinear coordinates [3] related to the map.

I. INTRODUCTION

Vehicle localization is a crucial task for autonomous driving and advanced driver assistance systems (ADAS). This task is usually achieved using Global Navigation Satellite Systems (GNSS), but GNSS does not provide better than 2-3 m in open sky environments [1]. Fusing low cost solutions based on GNSS with inertial measurement units improves the accuracy but is still not sufficient for autonomous driving level 4.

The use of prior maps becomes an interesting source for localization since they can be pre-built very accurately. The localization task in a prior map or Map-based localization is the process of associating sensor data with information available in the map. In state of the art approaches, feature maps [4], [5] and dense maps [6] are utilized. Localization using dense maps is usually more accurate but suffers from high computational time. The use of feature maps decreases the computational time and makes the localization task more suitable for real time application. In addition, since third party maps are often provided with sparse geometric features (lane markings, road signs, poles), raw sensor data cannot be directly associated with the map and a perception system is therefore required. LIDAR sensors have been widely used for environment perception thanks to their accurate range measurements, performance in low light conditions and large field of view. However, processing LIDAR points cloud is time consuming a reason why 3D points are usually projected into grid representations to speed up the process.

This paper is an extension of [2] where a LIDAR-based lane markings detection for vehicle localization was proposed. In addition to lane markings, the proposed approach detects road signs using a multi-layer LIDAR and evaluates their impact on the localization accuracy. As shown in [2], lane markings are stable and robust features to obtain good localization accuracy in the lateral direction (cross-track), nevertheless the longitudinal error (along-track) is not affected. In this approach, we investigate the impact of road signs detection on the along-track error. Experiments were conducted on a highway-like test track at different ego vehicle speeds. The laser range finder used in our experiments is a Velodyne VLP-32C. Moreover, a highly accurate GPS with RTK correction signals (ixblue: ATLANS-C) was used as ground truth for evaluation purposes. The third party map [1] is built with centimetric precision ($\approx$ 5 cm) and contains high level geometric features such as lane markings, road signs, road intersections, etc.

The remainder of this paper is structured as follows: section II is a brief survey on map-based localization methods, section III describes the proposed road signs detection approach. Section IV illustrates the developed map-matching algorithm. An experimental evaluation is presented in section V and we conclude with perspectives and future work in section VI.

II. RELATED WORK

Different map-based localization approaches have been proposed in the literature. In most cases, two sensors have been studied: vision-based and laser-based. Vision-based techniques [7], [8], [9], [10], [11] match visual features or landmarks to map attributes. Visual landmarks can be of different types: lane markings and curbs [9], traffic signs painted on the road such as arrows, pedestrian crossings and speed limits [11], traffic signs (vertical poles) [12], or feature points such as SIFT [10]. Despite promising results, vision-based techniques depend heavily on illumination conditions and suffer from scale estimation for the monocular case.

---

[1]The name of the map provider is not mentioned for confidential reasons

On the other hand, LIDAR-based techniques have proven to be more accurate since LIDAR range measurements are precise. In addition, LIDARs do not depend on external illumination, they can cover a large field of view and provide 3D representation of the environment. Processing the whole points cloud is time consuming and is not suitable for real time applications. Therefore, they are usually projected into 2D representations such as 2D orthographic reflectivity grid [13], precise height grid or elevation map [14], [15], or a combination with other information such as colors, curvatures and normals [16]. Finding correspondences between features and map attributes often relies on measuring similarities by defining a specific metric. In [17] a maximization of normalized mutual information (NMI) is implemented. Other methods minimize a cost function such as re-projection error [18], the distance between points [19], etc.

III. DETECTION OF ROAD SIGNS

In the literature, vision-based road sign detection methods are largely more addressed than LIDAR-based techniques. Since 3D LIDAR point cloud is dense, prior information on the location of the road sign is of key importance. For example, in [20], a vision-based system is used to detect the road limits. Indeed, since road signs are assumed to be located outside road boundaries, road points are removed from the cloud and only LIDAR points that are outside the road boundaries are used for the road sign detection process. In [21], an intensity map is created. Then, pixels with high reflectivity values are clustered to indicate prior road sign locations. Our approach is similar to [22] and is a three-step algorithm. First, a front polar grid is constructed from the whole lidar 3D points. Second, reflectivity data are used to construct a binary front image on which a morphological closing operation is used to identify ROIs (Region Of Interests). Later, ROI points are fed into a RANSAC plane estimation algorithm to estimate the centroid c and the normal n of the plane. Finally, ROI candidates are further refined by setting constraints on the number of inliers and the plane normal n. Each step will be described in the following paragraphs.

A. Front polar grid

A rotating LIDAR measures three parameters: the range r, the azimuth angle ϕ and the vertical beam angle θ. The front polar grid is constructed in polar coordinates with a horizontal (azimuth) field of view: $\pm\phi_{max}$ and N rows, where N is the number of LIDAR beams (see Fig.1). In addition, the number of columns of the grid is defined by setting an azimuth angular resolution ϕ_{res}. For each grid cell $m_{i,j}$, the set of 3D points:

$$\mathcal{P}_{i,j} = \{p_{i,j}^1, p_{i,j}^2, ..., p_{i,j}^N\} \qquad (1)$$

that are within $m_{i,j}$ and the average intensity value are kept.

B. Detection of Region Of Interest (ROI)

In order to locate the positions of potential road signs, a front reflectivity image is created by taking into account the maximum intensity value of each grid cell. Since road signs are very reflective, the latter is thresholded in order to obtain a 2D binary image. The selected threshold is chosen according to the specifications of the sensor provided by the lidar manufacturer. A morphological closing operation is then implemented to extract ROIs as a set of bounding boxes Bx_i.
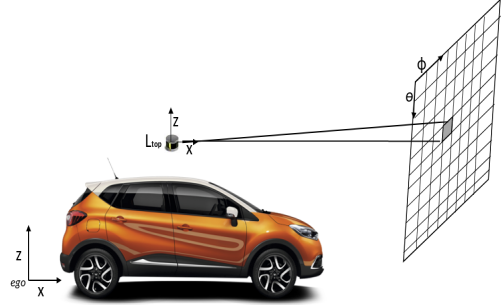

Fig. 1. Lidar projection to a front grid

C. De-skewing of ROI 3D points

The detected region of interests on the binary image are now used in parallel with the front grid to extract 3D points corresponding to each bounding box Bx_i. These 3D points are used to estimate 3D parameters of a road sign candidate. However, a typical problem of mobile spinning LIDARs is the distortion of the point cloud. Indeed, the time difference (spinning rate) between the start and the end of a complete rotation of the LIDAR is approximately $0.1s$. For an ego vehicle speed of $25m/s$, the spinning rate induces a translation gap of $2.5m$. To de-skew this distortion, we used the timestamp t_k of each point p_k. Indeed, our LIDAR is synchronized to a GNSS clock and all points are consequently timestamped. This correction is implemented in two steps. First, the minimum timestamp t_{min} is computed for each ROI set of points. Then, by using inertial data (velocity v and yaw rate w), a 3D point $p_k = (x_k, y_k, z_k, t_k)$ is corrected as follows:

$$x_k = x_k + dt \times v \times cos(dt \times w) \qquad (2)$$
$$y_k = x_k + dt \times v \times sin(dt \times w) \qquad (3)$$
$$dt = t_k - t_{min} \qquad (4)$$

The above equations are computed by assuming a constant velocity and yaw rate motion model.

D. Road sign parameter estimation

Now that we have a de-skewed 3D points of potential road sign candidates, we implemented a RANSAC [23] plane

fitting algorithm to further refine the detection. The result of the estimation is a centroid $c = (c_x, c_y, c_z)$ and a normal vector $n = (n_x, n_y, n_z)$. A candidate (c, n) is retained if the following requirements are met:

- The normal n is nearly parallel to the driving direction (direction of x axis).
- The number of inliers of the RANSAC estimation over the total number of points is greater than 0.5.
- The distance of the centroid c to the vehicle is less than $d_{max} = 30m$.

(a) Lidar raw data   (b) Projection to a front view

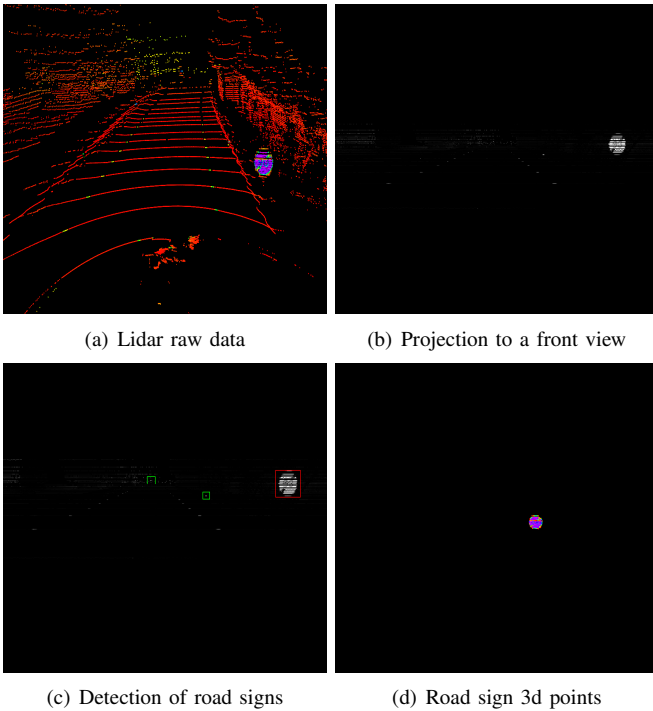(c) Detection of road signs   (d) Road sign 3d points

Fig. 2. Detection of road signs from LIDAR points

The projection of lidar 3d points (Fig 2.a) into a front reflectivity image is depicted in figure 2.b . In this case, we have one road sign to be detected. The application of the detection process gives the result of three bounding boxes (Fig 2.c), only one of which (the red bounding box) is considered as a good detection after the application of the three selection criteria. Finally, in figure 2.d, the corresponding 3D points of the red bounding box is illustrated.

The sole use of the lidar sensor to detect a valid road sign for the localization system can lead to false positives in the presence of other reflective objects: such as a vehicle's license plate, reflective stripes on an ambulance, etc. Interestingly, the combination with the HD map can solve this problem as the HD map only contains static objects. Consequently, a detected road sign that satisfies the three selection criteria is considered to be valid if a correspondent road sign exists in the map.

IV. MAP-BASED LOCALIZATION

A. Coordinate Systems & Map Description

Our prototype map is a third party map with high absolute accuracy. The used map attributes in this paper are road markings and road signs. Road markings are expressed as polylines and road signs are expressed as 2D polygons (triangles) for which we have the centroid position. Map elements are expressed with respect to the global reference frame or the map reference frame:

$$\mathcal{G} = (x_0, y_0, z_0)$$

However, the perception data are expressed with respect to the lidar reference frame for which we know the geometric transformation to the ego vehicle reference frame:

$$\mathcal{V} = (x_{ego}, y_{ego}, z_{ego}).$$

where the origin is the middle of the rear axle, the x-axis points forward and the y-axis points laterally from right to left (Fig 1).

B. Particle Filtering

The localization framework is a map-matching algorithm using particle filtering (PF) [24]. The posterior density distribution of the estimated vehicle position is approximated by a set of weighted samples or particles. Let the set of samples be:

$$\mathcal{X}_k := \langle x_k^{[1]}, w_k^{[1]} \rangle, ..., \langle x_k^{[N]}, w_k^{[N]} \rangle \quad (5)$$

The state space vector

$$\mathbf{x} = [x, y, \gamma] \quad (6)$$

consists of a 2D position $\langle x, y \rangle$ and a heading γ. Suppose also that the reference frame attached to a particle position is $\mathcal{V}_i$. The geometric transformation from $\mathcal{V}_i$ to the global frame is given by

$$\mathcal{T}_{\mathcal{V}_i}^{\mathcal{G}} = [\mathbf{R}_i | \mathbf{t}_i]. \quad (7)$$

The implementation of a particle filter is a three-step algorithm:

C. The prediction step

The adopted motion model for the prediction step is a Constant Velocity and Yaw Rate (CVYR) model [25]. Vehicle velocity v and yaw rate w are measured from an inertial measurement unit and are used in the following equations:.

$$\gamma_k^{[i]} = \gamma_{k-1}^{[i]} + w_k \times dt + \nu_w \quad (8)$$

$$x_k^{[i]} = x_{k-1}^{[i]} + v_k \times dt \times cos(\gamma_k^{[i]}) + \nu_x \quad (9)$$

$$y_k^{[i]} = y_{k-1}^{[i]} + v_k \times dt \times sin(\gamma_k^{[i]}) + \nu_y \quad (10)$$

where $\nu_w = \mathcal{N}(0, \sigma_w)$, $\nu_x = \mathcal{N}(0, \sigma_x)$, $\nu_y = \mathcal{N}(0, \sigma_y)$ are Gaussian noise distributions.

D. The update step

In this step, particle weights are updated according to the perception data and map attributes. In our approach, we computed three different sub-weights related to each perception data: lane markings: w_{lanes}, road signs: w_{signs} and GNSS signals: w_{gnss}.
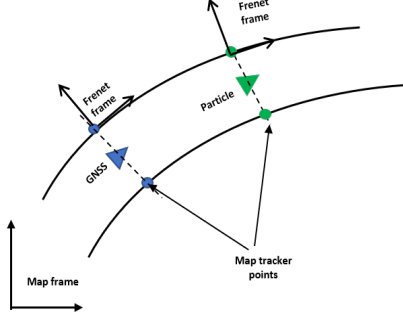
Fig. 3. Illustration of GNSS-based update step

Fig. 4. projection to the nearest segment

1) Update from GNSS signals: our method is similar to [9] where the euclidean distance between each particle and the GNSS position is compared to a confidence value proper to the sensor. If the distance is inside the confidence interval, the particle is kept, otherwise, a zero weight will be attributed. Of course, such a method acts on both lateral and longitudinal directions. Differently, in our approach the GNSS signals are used to bound only the longitudinal error without affecting the lateral direction by changing the coordinate system from the absolute frame related to the map to the Frenet frame related to map road markings. This is ensured by projecting all the positions (GNSS + particles) onto map road markings and to obtain, as a result, what we call in this paper map tracker points (MTP) (Figure 3). For each particle, the average euclidean distance, $\mathcal{D}_{tracker}^{[i]}$, between particle MTPs and GNSS MTPs is computed. Thus, the calculation of $w_{GNSS}^{[i]}$ is given by:

$$\begin{cases} w_{GNSS}^{[i]} = 1, & \text{if } \mathcal{D}_{tracker}^{[i]} \leq \rho_{max} \\ w_{GNSS}^{[i]} = 0, & \text{otherwise.} \end{cases}$$

where ρ_{max} is a coarse estimation of the GNSS receiver position error.

2) Update from road markings: since the detected road markings are expressed in polar coordinates [2], the same data type should be computed from the map polylines. To do that, the nearest segment of a map polyline to a particle position is searched (see Fig 4). Then, the polar coordinates of the nearest segment are computed. The likelihood function for a detected road marking $\mathcal{L}_d = (r_d, \theta_d)$ and a projected map polyline $\mathcal{L}_{map}^{\mathcal{V}_i} = (r^{\mathcal{V}_i}, \theta^{\mathcal{V}_i})$ is given by:

$$f(\mathcal{L}_{map}^{\mathcal{V}_i}, \mathcal{L}_d) = e^{\frac{-(r^{\mathcal{V}_i} - r_d)^2}{2\sigma_r^2}} + e^{\frac{-(\theta^{\mathcal{V}_i} - \theta_d)^2}{2\sigma_\theta^2}} \quad (11)$$

The calculation of $w_{lanes}^{[i]}$ is given by summing over all the map polylines and over all the detected road markings:

$$w_{lanes}^{[i]} = \alpha \sum_{\mathcal{L}_{map}^{\mathcal{V}_i}} \sum_{\mathcal{L}_d} f(\mathcal{L}_{map}^{\mathcal{V}_i}, \mathcal{L}_d) \quad (12)$$

where α is a normalization factor. σ_r and σ_θ are the lane measurement variances.
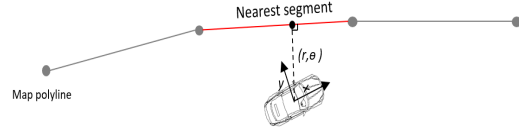
3) Update from road signs: the use of lane markings [2] has proved to be stable and relevant for the lateral localization. However, these features do not improve significantly the longitudinal error. Road curvatures can be utilized to reduce the longitudinal drift [26] but require high curvature values to be efficient. Since we are dealing with highway scenarios, road curvatures are low and are less suitable in this case. Our strategy is therefore to decouple the longitudinal and the lateral directions. For the lateral direction we solely rely on the road markings. The longitudinal error is bounded, on the one hand, by the GNSS position, on the other hand, by integrating road signs. Suppose that a detected road sign is given by: $\mathcal{S}_d = (s_x^d, s_y^d)$. The correspondent map road sign is extracted and projected into the particle reference frame: $\mathcal{S}_{map}^{\mathcal{V}_i} = (s_x^{\mathcal{V}_i}, s_y^{\mathcal{V}_i})$. The likelihood function is therefore given by:

$$f(\mathcal{S}_{map}^{\mathcal{V}_i}, \mathcal{S}_d) = e^{\frac{-(s_x^{\mathcal{V}_i} - s_x^d)^2}{2\sigma_{s_x}^2}} \quad (13)$$

The calculation of $w_{signs}^{[i]}$ is given by:

$$w_{signs}^{[i]} = \beta \sum_{\mathcal{S}_{map}^{\mathcal{V}_i}} \sum_{\mathcal{S}_d} f(\mathcal{S}_{map}^{\mathcal{V}_i}, \mathcal{S}_d) \quad (14)$$

Where β is a normalization factor and σ_{s_x} is a measurement variance.

In real highway roads, road signs are not repetitive features and can be absent for hundreds of meters. In this case, only lane markings are integrated in the update step and the longitudinal error is therefore subject to significant drift due to odometry integration errors. In this case, all particles are rapidly drifting away from the area where the true position is located (see Fig 5.a). This behaviour does not improve the longitudinal error as no good particles can be selected from the updated weights. One possible way to solve this problem is to increase the number of particles by populating more regions in the map and therefore increasing the density of particles around the true position, nevertheless, this may reduce the time performance as the number of particles grows significantly. In addition, randomly generating particles is not efficient since we know that some regions of the map are unlikely to be populated by particles. We propose a more elegant way to generate particles by relying on the lateral position of the filter estimate from the previous iteration and

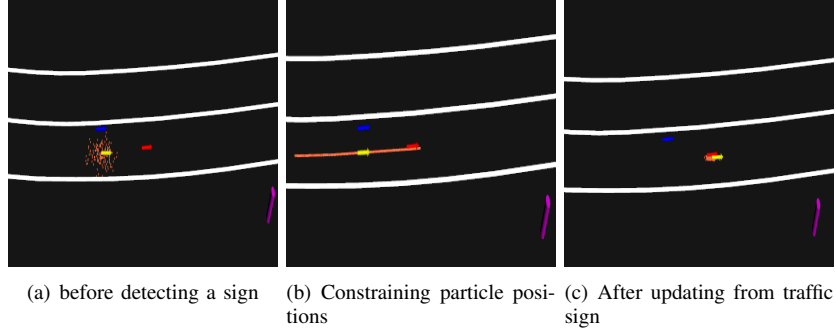(a) before detecting a sign   (b) Constraining particle positions   (c) After updating from traffic sign

Fig. 5. Map-based constrained update: the red marker is the RTK position, the blue marker is a low cost GNSS position and the yellow marker is the filter estimate. The drawn particles are in orange and the white lines are the map road markings. The purple pole illustrates a map road sign.

the geometry of the road inferred from the map, namely *constrained update*. The idea would be to redistribute the sample set more efficiently and to maintain the same number of particles. Thanks to the road markings, the filter estimate is often well localized in the lateral direction. A new set of particles is placed along a generator curve $\mathcal{C}$ that is parallel to the map road markings and that has the same lateral offset as the previous filter estimate (see Fig 5.b). Thus, the new set of samples covers more space in the longitudinal direction and the weighting step improves the accuracy (see Fig 5.c)

E. Re-sampling step

From the previous calculations, the total particle weight is given by:

$$ w^{[i]} = w^{[i]}_{GNSS} \times w^{[i]}_{lanes} \times w^{[i]}_{signs} \quad (15) $$

The re-sampling step is designed to alleviate one of the fundamental problems of particle filtering which is the degeneracy problem. After some iterations, particles tend to concentrate only on few particles and assign negligible weights to the others [27]. We implemented a systematic re-sampling strategy [28] that was applied to the normalized weights (16):

$$ w^{[n]} = \frac{w^{[n]}}{\sum_{i=1}^{N} w^{[i]}} \quad (16) $$

V. EXPERIMENTAL RESULTS

Experiments were conducted on a double lanes Highway test track of approximately 5 km long (Fig. 6). Our prototype vehicle is equipped with a GNSS/IMU (ixblue: ATLANS-C) localization unit with RTK correction signals of 5cm absolute accuracy, a Velodyne VLP-32C laser scanner (10-20 Hz, 32 laser beams) and an automotive GPS/IMU (u-blox B78-ADR). All the sensors are synchronized to GNSS time clock and data is collected at different ego vehicle speeds to evaluate the robustness of our approach. The map matching algorithm has been developed under the ROS framework.

Fig. 6. Renault test track (google earth map)

A. Evaluation metric

To evaluate accuracy results of our localization system, we adopted a curvilinear coordinate system relative to the road as explained in [3]. Errors are given in terms of cross-track (CT) error (lateral error) and along track (AT) error (longitudinal error). The evaluation of the CT error has been explained in [2]. In this section we will explain the calculation of the along track error. At first, map tracker points related to map lane markings L_i are computed for our estimated position and for the ground truth position: they are noted by p^i_{est} and p^i_{gt}, respectively. The AT error is therefore computed as the line integral along road marking curves and is given as follows:

$$ AT = \frac{1}{N} \sum_{i=1}^{N} \int_{p^i_{est}}^{p^i_{gt}} L_i(r) ds $$

This formula is simplified when the curve L can be approximated by a line and becomes: $\int_{p^i_{est}}^{p^i_{gt}} L_i(r) ds = \| p^i_{est} - p^i_{gt} \|$. This is convenient for highway use cases as road curvatures are low and can be modeled by polylines. The new AT error value is given by:

$$ AT = \frac{1}{N} \sum_{i=1}^{N} \| p^i_{est} - p^i_{gt} \| $$

B. Localization accuracy

The AT and CT errors have been evaluated for a high speed scenario (90 Kph) and a low speed scenario (30 Kph). Three filter configurations are proposed for the evaluation:

- *The integration of road signs without the implementation of the constrained update strategy.*
- *The integration of road signs with the implementation of the constrained update strategy.*
- *The only use of GNSS data without the consideration of road signs.*

TABLE I
ERROR EVALUATION AT 30 KPH

	Along track error (m)	Cross track error (m)
Un-constrained update	mean: **1.49** std: **2.4**	mean: **0.11** std: **0.35**
Constrained update	mean: **0.93** std: **1.39**	mean: **0.09** std: **0.22**
GNSS Only	mean: **2.81** std: **3.63**	mean: **0.15** std: **0.43**

TABLE II
ERROR EVALUATION AT 90 KPH

	Along track error (m)	Cross track error (m)
Un-constrained update	mean: **1.53** std: **1.89**	mean: **0.02** std: **0.49**
Constrained update	mean: **0.78** std: **1.11**	mean: **-0.01** std: **0.43**
GNSS Only	mean: **1.92** std: **2.41**	mean: **0.03** std: **0.49**

All the configurations share the same conditions: the same particle initialization, the same number of particles (= 200) and the same odometry errors. The difference lies only within the weighting update. Figure 7 illustrates the time variations of the along track and the cross track errors for two different vehicle speeds. Detection occurrences of road signs are indicated by red arrows. For the first few seconds, the along track error is similar for the three configurations. However after the first detection of a road sign the error decreases significantly. The proposed constrained update strategy performs better than the classical particle filtering implementation which, in turn, performs better than the use of the GNSS only. The mean AT error at low speed is $0.93m$ for the constrained update strategy versus $1.49m$ for the un-constrained update. At high speed, the AT error is $0.78m$ for the proposed approach versus $1.53m$. Tables I and II summarize the AT and CT errors. The proposed approach gives comparable results for high and low speeds.

However, when no road sign is detected, the AT error is subject to drift. This is clearly visible in figures 7.a and 7.c between the third and the last detected road signs (third and fourth red arrow). In addition to the AT error, the proposed implementation has a faster convergence rate than the un-constrained implementation. This is crucial for high speed scenarios as the road sign is seen for a shorter time. Although the integration of road signs is supposed to act only on the longitudinal direction as depicted in the equation (13). The proposed approach performs better in terms of the CT error.

Indeed the achieved accuracy is $0.31m$ for the constrained update versus $0.46m$ for the un-constrained update at low speed and $0.42m$ versus $0.51m$ at high speed.

VI. CONCLUSION

In this paper, we presented a LIDAR-based localization system within a High Definition (HD) map on highway scenarios. Detected features in this paper are: lane markings and road signs. Detection of road signs is obtained by constructing a front grid/image from the LIDAR and by detecting Region Of Interests in a binary image resulted from LIDAR reflectivities. RANSAC plane fitting is then applied to refine the selected ROI candidates. Localization is implemented by virtue of particle filtering which consists in matching detected features with a third party map.

The first contribution of this paper is to prove the usefulness of third party sparse maps in order to achieve high localization accuracy. This is challenging since third party maps contain geometric features that are sometimes not adapted to the type of the sensor being used and to the setup of the vehicle. Put differently, geometric maps have the advantage to be reusable with different sensors granted that a good perception system can be developed. Moreover, sparse maps make real time application more realistic and achievable. Finally, a solution to cope with longitudinal drift and its impact on particles weighting is proposed by implementing a constrained update strategy based on map road shape.

Future work will include a camera sensor to integrate a visual-based localization. The core idea is to reduce the drift between two consecutive road signs since vision-based localization systems are more accurate then inertial measurement units.

REFERENCES

[1] J.-M. Zogg, *GPS: Essentials of Satellite Navigation: Compendium: Theorie and Principles of Satellite Navigation, Overview of GPS/GNSS Systems and Applications.*, 2009
[2] F. Ghallabi, F. Nashashibi, G. El Haj Shhade, and M.-A. Mittet, "LIDAR-Based Lane Marking Detection for Vehicle Positioning in an HD Map," *21st IEEE International Conference on Intelligent Transportation Systems*, nov 2018
[3] E. Héry, S. Masi, P. Xu, and P. Bonnifait, "Map-based Curvilinear Coordinates for Autonomous Vehicles," pp. 1699–1705, 2017.
[4] N. Mattern, R. Schubert, and G. Wanielik, "High-accurate vehicle localization using digital maps and coherency images," in *IEEE Intelligent Vehicles Symposium, Proceedings*. IEEE, jun 2010, pp. 462–469
[5] K. Jo, Y. Jo, J. K. Suhr, H. G. Jung, and M. Sunwoo, "Precise Localization of an Autonomous Car Based on Probabilistic Noise Models of Road Surface Marker Features Using Multiple Cameras," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3377–3392, dec 2015
[6] J. Levinson and S. Thrun, "Robust vehicle localization in urban environments using probabilistic maps," *2010 IEEE International Conference on Robotics and Automation*, pp. 4372–4378, 2010
[7] M. Buczko and V. Willert, "Efficient Global Localization Using Vision and Digital Offline Map," pp. 1689–1694, 2017.
[8] Z. Zhu, T. Oskiper, S. Samarasekera, R. Kumar, and H. S. Sawhney, "Real-time global localization with a pre-built visual landmark database," *26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2008

(a) AT error at 30 kph

(b) CT error at 30 kph

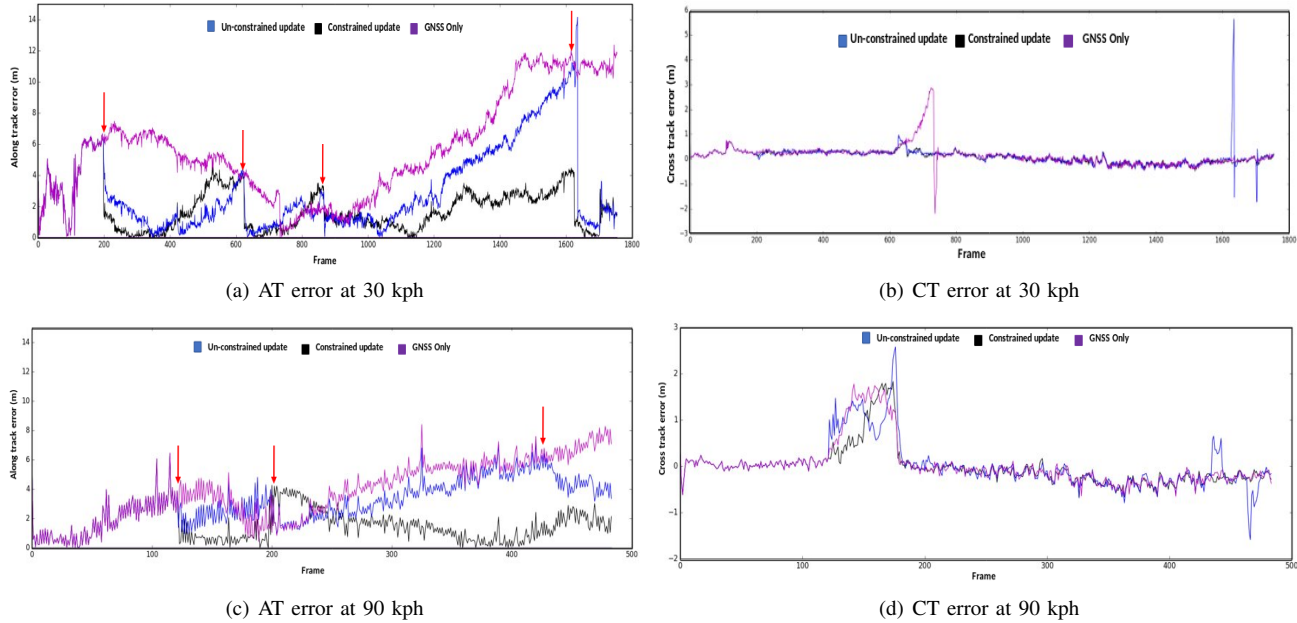
(c) AT error at 90 kph

(d) CT error at 90 kph

Fig. 7. Illustration of the cross track (CT) and the along track (AT) errors at different vehicle speeds and for the three use cases: constrained update, un-constrained update and GNSS only. Occurrences of road signs update are indicated by red arrows.

[9] F. Chausse, J. Laneurit, and R. Chapuis, "x," *IEEE Intelligent Vehicles Symposium, Proceedings*, vol. 2005, pp. 243–248, 2005.

[10] J. Košecká, F. Li, and X. Yang, "Global localization and relative positioning based on scale-invariant keypoints," *Robotics and Autonomous Systems*, vol. 52, no. 1, pp. 27–38, 2005

[11] T. Wu and A. Ranganathan, "Vehicle localization using road markings," in *IEEE Intelligent Vehicles Symposium, Proceedings*. IEEE, 2013, pp. 1185–1190

[12] H. Li, F. Nashashibi, and G. Toulminet, "Localization for intelligent vehicle by fusing mono-camera, low-cost GPS and map data," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*. IEEE, sep 2010, pp. 1657–1662

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

[14] H. Fu, L. Ye, R. Yu, and T. Wu, "An efficient scan-to-map matching approach for autonomous driving," in *2016 IEEE International Conference on Mechatronics and Automation, IEEE ICMA 2016*. IEEE, aug 2016, pp. 1649–1654

[15] E. Pollard, J. Perez, and F. Nashashibi, "Step and curb detection for autonomous vehicles with an algebraic derivative-based approach applied on laser rangefinder data," *IEEE Intelligent Vehicles Symposium, Proceedings*, pp. 684–689, 2013

[16] Z. J. Chong, B. Qin, T. Bandyopadhyay, M. H. Ang, E. Frazzoli, and D. Rus, "Mapping with synthetic 2D LIDAR in 3D urban environment," *IEEE International Conference on Intelligent Robots and Systems*, pp. 4715–4720, 2013

[17] R. W. Wolcott and R. M. Eustice, "Visual Localization within LIDAR Maps for Automated Urban Driving," 2016.

[18] H. Lategahn, M. Schreiber, J. Ziegler, and C. Stiller, "Urban localization with camera and inertial measurement unit," in *IEEE Intelligent Vehicles Symposium, Proceedings*. IEEE, jun 2013, pp. 719–724

[19] M. Schreiber, C. Knoppel, and U. Franke, "LaneLoc: Lane marking

based localization using highly accurate maps," in *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013, pp. 449–454

[20] L. Zhou and Z. Deng, "LIDAR and Vision-Based Real-Time Traffic Sign Detection and Recognition Algorithm for Intelligent Vehicle," in *17th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, oct 2014, pp. 578–583

[21] B. Riveiro, L. Diaz-Vilarino, B. Conde-Carnero, M. Soilan, and P. Arias, "Automatic Segmentation and Shape-Based Classification of Retro-Reflective Traffic Signs from Mobile LiDAR Data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 1, pp. 295–303, 2016.

[22] A. Vu, Q. Yang, J. A. Farrell, and M. Barth, "Traffic sign detection, state estimation, and identification using onboard sensors," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*. IEEE, oct 2013, pp. 875–880

[23] M. A. Fischler and R. C. 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

[24] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," *Robotics and Automation,*, 1999

[25] R. Schubert, C. Adam, M. Obst, N. Mattern, V. Leonhardt, and G. Wanielik, "Empirical evaluation of vehicular models for ego motion estimation," *IEEE Intelligent Vehicles Symposium, Proceedings*, no. Iv, pp. 534–539, 2011.

[26] N. Suganuma and T. Uozumi, "Precise position estimation of autonomous vehicle based on map-matching," *IEEE Intelligent Vehicles Symposium, Proceedings*, no. Iv, pp. 296–301, 2011.

[27] T. Li, S. Sun, T. P. Sattar, and J. M. Corchado, "Fight sample degeneracy and impoverishment in particle filters: A review of intelligent approaches," *Expert Systems with Applications*, vol. 41, no. 8, pp. 3944–3954, aug 2014

[28] O. C. R. Douc, "Comparison of resampling schemes for particle filtering," *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis*, pp. 64–69, 2005.