Lane Marking Based Vehicle Localization Using Particle Filter and Multi-Kernel Estimation

Wenjie Lu, Emmanuel Seignez, Sergio Alberto Rodriguez Florez, Roger Reynaud

Lane Marking Based Vehicle Localization Using Particle Filter and Multi-Kernel Estimation

Wenjie Lu, Emmanuel Seignez, Sergio A. Rodriguez F. and Roger Reynaud
Université Paris-Sud, Institut d'Electronique Fondamentale, UMR8622, Orsay, F-91405
CNRS, Orsay, F-91405
Email: {wenjie.lu, emmanuel.seignez, sergio.rodriguez, roger.reynaud}@u-psud.fr

***Abstract*—Vehicle localization is the primary information needed for advanced tasks like navigation. This information is usually provided by the use of Global Positioning System (GPS) receivers. However, the low accuracy of GPS in urban environments makes it unreliable for further treatments. The combination of GPS data and additional sensors can improve the localization precision. In this article, a marking feature based vehicle localization method is proposed, able to enhance the localization performance. To this end, markings are detected using a multi-kernel estimation method from an on-vehicle camera. A particle filter is implemented to estimate the vehicle position with respect to the detected markings. Then, map-based markings are constructed according to an open source map database. Finally, vision-based markings and map-based markings are fused to obtain the improved vehicle fix. The results on road traffic scenarios using a public database show that our method leads to a clear improvement in localization accuracy.**

I. Introduction

Vehicle localization plays a fundamental and critical role in Intelligent Transportation System (ITS) because it is the prior task for higher level operations. Various methods have been introduced to improve the accuracy of the results using different additional sensors [1]. Before digital map is considered, most of the localization methods rely on Global Navigation Satellite System (GNSS) and Inertial Navigation System (INS). In [2], only GNSS position is depended, the positioning problem is formulated as a Bayesian inference problem in a factor graph. In [3], GPS and INS signals are integrated because INS continues to estimate vehicle positions when GPS signals are lost in certain conditions.

With the development of cartography, digital maps are integrated into vehicle navigation. Map-based methods mainly contain two parts, map-matching algorithm and vehicle localization method. Map-matching [4] aims to locate the vehicle position on the map, by fusing both map features (i.e. geometrical and topological information) and data collected from vehicle (i.e. GPS, direction, road shapes, velocities). Particle filter is commonly used in map-matching [5]. Vehicle localization part uses map information to adjust vehicle location. Common methods in vehicle localization include Kalman Filter [6][7], Extended Kalman Filter (EKF) [8], optimization algorithm [9], and methods relying on interval analysis [10]. In [11], multi-object localization is performed to improve vehicle localization.

Lane marking based localization is regarded as one branch of map-based localization. On one hand, marking based methods introduce vision into localization, fusing image cues to help localization. On the other hand, most map-based localization methods adopt only the topological and geometrical information of the map, while in lane marking based localization, map information are adequately utilized, i.e., number of lanes, lane width and road type. In [12], traffic signs (i.e. arrows, pedestrian crossings and markings) from both vision source and map source are compared to provide a translation vector from vision space to map space. Vehicle location is then adjusted according to this vector. In [7], all markings from both offline maps and visions at a stop intersection are compared using particle filter to estimate vehicle position. In [6], belief theory combines several criteria to implement map-matching. A Kalman filter, combining the vehicle poses and selected map segments together, are designed to estimate vehicle coordinate. In [8], a full-state EKF, including located markings and GPS signals with error model, is adopted to locate vehicle fix.

Most existing papers on marking based localization require high-accuracy digital maps, including the precise locations of all the lane markings. Even all the traffic signs on the road surface are needed in some publications. High-accuracy digital maps help to improve localization results. However, these maps are specially customized, which is time-consuming when constructing and range-limited when adopting. Meanwhile, only simple road situations are considered in remaining methods, for instance, one direction way without multi-lane.

In this paper, a marking feature based vehicle localization method is proposed, aiming at a wide applied range and accurate localization. The main idea is to extract and fit vision-based and map-based markings to optimize vehicle position. In Section II, lane markings are detected through image processing and parameter estimation. Section III designs a particle filter to adjust rough GNSS positions using lane marking model. This particle filter can be considered as a first adjusting on rough vehicle positions. In Section IV, the road segment is estimated using multi-criterion estimation, the map-based lane markings of current state are then determined from an open source map database. In Section V, vehicle location is estimated according to fusing both vision-based lane markings and map-based lane markings, using a multi-kernel estimation method. Finally, Section VI presents experimental results.

II. Lane Marking Detection

In this section, images from on-vehicle camera are processed to extract road marking features. Marking model parameters are then estimated using a multi-kernel method. This lane marking detection method has been demonstrated in [13].

A. Image Processing

The input images are processed through Inverse Perspective Mapping (IPM). This translation provides vertical and paralleled lane markings in Bird Eye's View (BEV) space, which greatly facilitates marking detection strategies. A second order derivative filter along the horizontal direction is applied to process BEV images where lane markings are nearly vertical. To eliminate the remaining outliers, a cell-based blob algorithm, improved from [14], is introduced. By the end of image processing part, a binary image is produced.

B. Parameter Estimation

A parabola: $x = c + d \cdot y + e \cdot y^2$ is chosen as the marking model in BEV space. The marking initialization step determines the zero order component c. The first and second order components d and e are estimated through an improved multi-kernel based method with hierarchical weights.

Marking initialization step aims to determine the zero order components c_l and c_r of left and right markings together using a parametric Gaussian model based method. To this end, the intersections of both lane markings at x-axis in BEV images are estimated using multi-criterion. These two intersections are exactly model parameters c_l and c_r. The distributions considered to estimate c_l and c_r are: previous detection distribution X_1, white pixels distribution X_2, Hough lines distribution X_3, prior data distribution X_4 and lane width distribution X_5. The probability distribution of c_l and c_r is then given as:

$$p_t(c_l, c_r) = \left[\sum_{j=l,r} \left(\sum_{i=1}^{3} k_i \cdot p_i\left(c_j\right) \right) \cdot p_4(c_j) \right] \cdot p_5(c_l - c_r), \quad (1)$$

where $p_i(x)$ is probability density function of normalized $\overline{X_i}$, k_i is an importance coefficient of $p_i(x)$. Finally the intersections of left and right lane markings c_l^* and c_r^* are decided by:

$$c_l^*, c_r^* = \arg\max_{c_l, c_r} p_t(c_l, c_r). \quad (2)$$

A multi-kernel density based method with hierarchical weights, improved from [15], is introduced for estimating the remaining model parameters. The basic unit of this algorithm is the similarity between an image pixel (x_i, y_i) and a marking model (c, d, e) in a binary image:

$$G_{pi}(c, d, e, x_i, y_i) = \int_{-\infty}^{+\infty} K_x' K_y dy, \quad (3)$$

where,

$$K_y = \frac{1}{\sqrt{2\pi\sigma_{yi}^2}} exp(-\frac{(y - y_i)^2}{2\sigma_{yi}^2}), \quad (4)$$

$$K_x' = \frac{1}{\sqrt{2\pi\sigma_{xi}^2}} exp(-\frac{(c + dy + ey^2 - x_i)^2}{2\sigma_{xi}^2}). \quad (5)$$

GaussHermite quadrature method [16] is employed to compute the numerical solution of G_{pi}. When G_{pi} is derived, the probability of a specified model $p_{Gpi}(d, e)$ is defined as:

$$p_{Gpi}(d, e) = \frac{1}{n_t} \sum_{i=1}^{n_{total}} w(x_i, y_i) \cdot G_{pi}(c_m, d, e, x_i, y_i), \quad (6)$$

where n_{total} is the total number of white pixels in the binariezd image. $w(x_i, y_i)$ is a hierarchical weight, which offers corresponding coefficient according to different areas of the BEV image.

The left and right marking models are computed together as Eq. (7) to obtain the optimized parameters d_l^*, e_l^*, d_r^*, e_r^*, with a geometrical constraint presented in Eq. (8) and (9).

$$d_l^*, e_l^*, d_r^*, e_r^* = \arg\max_{d_{j_l}, d_{j_r}, e_{j_l}, e_{j_r}} [p_{Gpi}(d_{i_l}, e_{j_l}) + p_{Gpi}(d_{i_r}, e_{j_r})]. \quad (7)$$

$$d_{j_r} \in (d_{j_l} - \triangle d, d_{j_l} + \triangle d), \quad (8)$$

$$e_{j_r} \in (e_{j_l} - \triangle e, e_{j_l} + \triangle e). \quad (9)$$

Therefore, c_l^* and c_r^* are estimated in the initialization step, and d_l^*, e_l^*, d_r^*, e_r^* are optimized using multi-kernel method. At each sampling time, a pair of lane markings are detected. The confidence accorded to this procedure is modeled by the means of a self-assessment indicator detailed in our previous work [13]. Unreliable detections can then be efficiently identified and discarded based on the estimated confidence. In such cases, the last reliable estimation should be considered and predicted.

III. Lane Marking Based Particle Filter

Particle filter is a general Monte Carlo method to estimate the posterior density of the state variables given the observation variables. A Sampling Importance Resampling (SIR) filter performs three operations sequentially: prediction step, update step and resampling. Particle filter framework is used in the following to observe the vehicle position fusing the information provided by the detected lane markings and rough GPS and speed measurements (i.e. lineal and angular). The vehicle state at time k is represented by $s_k^{(i)} = \left[x_{enu,k}^{(i)}, y_{enu,k}^{(i)}, v_{veh,k}^{(i)}, \gamma_{veh,k}^{(i)}\right]^T$, where $\left[x_{enu,k}^{(i)}, y_{enu,k}^{(i)}\right]^T$ is the vehicle coordinate in meter in East-North-Up (ENU) space, $v_{veh,k}^{(i)}$ is the forward speed of host vehicle in meter per second and $\gamma_{veh,k}^{(i)}$ is the vehicle heading in degree. Particles are sampled in prediction step and their weights and filtered state are computed in update step. SIR algorithm is introduced to deal with degeneracy problem.

A. Initialization Step

At time 0, particles are initialized according to a first vehicle position reference denoted $(x_{enu,0}, y_{enu,0})$ in ENU space. N_{pf} particles $(x_{enu,0}^{(i)}, y_{enu,0}^{(i)})$, $i = 1, ..., N_{pf}$ are created following a uniform distribution along both x and y axis centering at $(x_{enu,0}, y_{enu,0})$, with the maximum peak of ± 10 meters.

B. Prediction Step

At time $k > 0$, velocity $v_{veh,k}^{(i)}$ and yaw angle $\gamma_{veh,k}^{(i)}$ of each particle are determined according to uniform distribution based on vehicle velocity $v_{veh,k}$ and vehicle yaw angle $\gamma_{veh,k}$ measurements. The maximum velocity noise is ± 10 m/s, and the maximum heading noise is $\pm 5°$. So the motion model of a particle in ENU space is represented as $L_{enu,k}^{(i)} = v_{veh,k}^{(i)} \cdot$

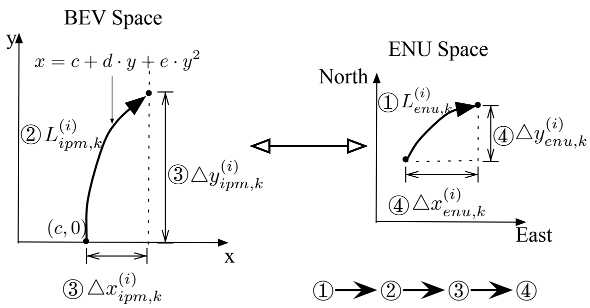


Fig. 1. Predicting step of particle filter.

T_{pf}, where T_{pf} is the time interval of a filter cycle. $L^{(i)}_{enu,k}$ is then translated to BEV space as $L^{(i)}_{ipm,k}$, shown in Fig. 1. Meanwhile, in BEV space, the trajectory length along marking model $x = c + d \cdot y + e \cdot y^2$ from the starting point $(c, 0)$ to a certain point (x, y) on the curve can be expressed as:

$$\begin{aligned} L(y) &= \int_0^y \sqrt{1 + (\partial x/\partial y)^2} dy \\ &= \frac{1}{2e}\{\frac{d+2ey}{2}\sqrt{(d+2ey)^2+1} + \frac{1}{2}ln[(d+2ey) \\ &+ \sqrt{(d+2ey)^2+1}] - \frac{d}{2}\sqrt{d^2+1} \\ &- \frac{1}{2}ln[d + \sqrt{d^2+1}]\}. \end{aligned} \quad (10)$$

Eq. (10) are derived from the integration expression $\int \sqrt{x^2+1}dx = \frac{x}{2}\sqrt{x^2+1} + \frac{1}{2}\ln\left(x + \sqrt{x^2+1}\right) + C$.

Therefore, the motion of a particle is given by:

$$\triangle y^{(i)}_{ipm,k} = \underset{y \in (0,+\infty)}{\arg\min} \left| L^{(i)}_{ipm,k} - L(y) \right|, \quad (11)$$

$$\triangle x^{(i)}_{ipm,k} = c + d \cdot \triangle y^{(i)}_{ipm,k} + e \cdot (\triangle y^{(i)}_{ipm,k})^2. \quad (12)$$

$\triangle x^{(i)}_{ipm,k}$ and $\triangle y^{(i)}_{ipm,k}$ are then translated to ENU space through:

$$[\triangle x^{(i)}_{enu,k}, \triangle y^{(i)}_{enu,k}, 1] = [\triangle x^{(i)}_{ipm,k}, \triangle y^{(i)}_{ipm,k}, 1] R(\gamma^{(i)}_{veh,k}), \quad (13)$$

where $R(\gamma^{(i)}_{veh,k})$ is translation matrix from BEV space to ENU space. The particle state after movement is then given by:

$$x^{(i)}_{enu,k} = x^{(i)}_{enu,k-1} + \triangle x^{(i)}_{enu,k}, \quad (14)$$

$$y^{(i)}_{enu,k} = y^{(i)}_{enu,k-1} + \triangle y^{(i)}_{enu,k}. \quad (15)$$

C. Update Step

In this step, the weight of each particle $w^{(i)}_k$ is refreshed with respect to the observation at time k. The importance weight of a particle is calculated according to a two-dimension Gaussian distribution centering at $(x_{enu,k}, y_{enu,k})$:

$$w^{(i)}_k = \frac{exp(-\frac{(x^{(i)}_{enu,k} - x_{enu,k})^2 + (y^{(i)}_{enu,k} - y_{enu,k})^2}{2\sigma_w^2})}{\sqrt{2\pi\sigma_w^2}}. \quad (16)$$

When $w^{(i)}_k$ is derived, normalized weight is computed as $\bar{w}^{(i)}_k = w^{(i)}_k / \sum_{i=1}^{N_{pf}} w^{(i)}_k$.

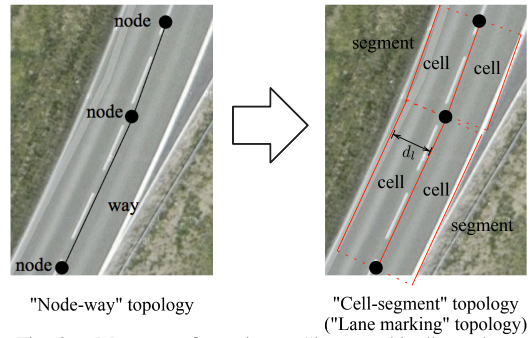


Fig. 2. Map reconfiguration to “lane marking” topology.

D. Resampling

The particles are resampled according to SIR algorithm. SIR algorithm can be seen as a variant of Sequential Importance Sampling (SIS). If the weight of a particle $w^{(i)}_k$ is below weight threshold, this particle is resampled to a certain existing particle. The probability to become a certain existing particle is proportional to the weight of this existing particle. SIR algorithm is implemented every 5 filtering cycles.

Finally, the approximated vehicle fix is given by

$$\hat{x}_{enu,k} = \sum_{i=1}^{N_{pf}} \bar{w}^{(i)}_k \cdot x^{(i)}_{enu,k}, \quad (17)$$

$$\hat{y}_{enu,k} = \sum_{i=1}^{N_{pf}} \bar{w}^{(i)}_k \cdot y^{(i)}_{enu,k}. \quad (18)$$

IV. Road Matching

In this section, the map database is at first reconfigured from a “node-way” topology to a “lane marking” topology. Then the current lane is selected according to multi-criterion. Thus the map-based lane markings are derived.

A. Map Reconfiguration

Map-based lane markings are obtained from OpenStreetMap (OSM), a collaborative project to provide open source map database. The basic components of OSM are a “node” and a “way”, as shown in Fig. 2. A node represents a specific point in map space. A way is an ordered list of nodes which is able to denote an object in the map.

In order to derive a more detailed subdivision of a way, the number of lanes and lane width d_l are taken into account. According to the geometrical relationship, nodes and ways are able to be translated to “segments” and “cells”, as shown in Fig. 2. A “segment” is a line segment of a poly-line “way”, and a “cell” is a single lane in a “segment”. The left and right cell boundaries of a cell (red lines in Fig. 2) are exactly lane markings.

B. Lane Selection

The aim of lane selection is to estimate the current “cell” of the host vehicle in a multi-lane “segment”. Lane selection stage is implemented to determine the current cell the vehicle is in two steps. In the first step, the map information and filtered vehicle positions are fused to decide the current “segment”. In ENU space, define d^j_{vs} as the distance between host vehicle V

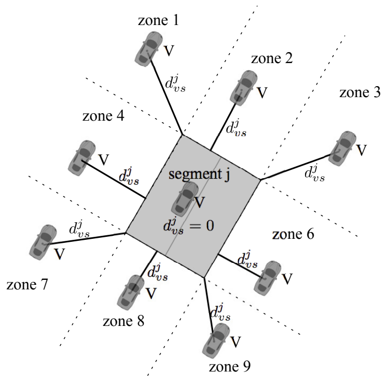


Fig. 3. Segment selection.

TABLE I
LANE CHANGING CRITERION AND THIRD MARKING CRITERION

lane changing criterion					third marking criterion				
lane number:		1	2	3	lane number:		1	2	3
left to right	$p_{l,lc}$	0	0	0	only left	$p_{l,mk}$	0	0	0
	$p_{m,lc}$	0	0	0.5		$p_{m,mk}$	0	0	0.5
	$p_{r,lc}$	0	1	0.5		$p_{r,mk}$	0	1	0.5
right to left	$p_{l,lc}$	0	1	0.5	only right	$p_{l,mk}$	0	1	0.5
	$p_{m,lc}$	0	0	0.5		$p_{m,mk}$	0	0	0.5
	$p_{r,lc}$	0	0	0		$p_{r,mk}$	0	0	0
lane keeping	$p_{l,lc}$	0			both	$p_{l,mk}$	0	0	0
	$p_{m,lc}$	0				$p_{m,mk}$	0	0	1
	$p_{r,lc}$	0				$p_{r,mk}$	0	0	0
					none	$p_{l,mk}$	0		
						$p_{m,mk}$	0		
						$p_{r,mk}$	0		

and current segment j , shown in Fig. 3. d_{vs}^j is determined according to which zone the host vehicle is in. The host segment is zone 5 in gray area. In zone $1, 3, 7, 9$, d_{vs}^j is the distance of vehicle center and a segment vertex, in zone $2, 4, 6, 8$, d_{vs}^j is the distance between vehicle center and a segment boundary line, in zone 5, the vehicle is in the segment, $d_{vs}^j = 0$. The current segment j_{cur} is selected as $j_{cur} = \arg\min_j d_{vs}^j$.

The second step is to determine which cell the vehicle is in, according to multi-criterion. Considered criteria include lane changing criterion, third lane marking criterion, and history vehicle state criterion. Assuming that the maximum lane number is 3, which are left lane, middle lane, and right lane. The corresponding probabilities of each lane are p_l, p_m, and p_r. If it is one-lane situation, only the middle lane p_m is considered. If dual-lane, introduce both left and right lanes p_l, p_r. If tri-lane, consider all p_l, p_m, and p_r. The first criterion is lane changing, which is determined according to the zero order component c in lane marking model $x = c + d \cdot y + e \cdot y^2$. The strategy is shown as in the left part of Table I, $p_{l,lc}$, $p_{m,lc}$ and $p_{r,lc}$ are the probabilities of left, middle, and right lane respectively. Take one situation as example, if there is a lane changing from a left lane to a right lane in a 3 lane segment, the host vehicle can be considered from the left lane to the middle lane, or from the middle lane to the right lane. So the current lane can be the middle lane or the right lane, which means $p_{l,lc} = 0$, $p_{m,lc} = 0.5$, $p_{r,lc} = 0.5$ respectively.

The second criterion is the third lane marking. At first, third lane marking parameter range is estimated through the detected two current lane markings in Sect II. $p_{l,mk}$, $p_{m,mk}$ and $p_{r,mk}$ are the probabilities of third lane marking criteria respectively. According to the third lane detection result, a similar strategy as lane changing criterion is demonstrated as the right part of Table I. For instance, if only the left lane marking is detected in a 3 lane segment, the vehicle can be in the middle lane or in the right lane. But noises such as road barriers and shadows may affect the third marking detection, which leads to some paradoxical conditions. For instance, a third marking on the left is detected in a one lane segment. In such paradoxical conditions, the third markings criterion doesn't work, $p_{l,mk}$, $p_{m,mk}$ and $p_{r,mk}$ are valued to 0.

The last criterion is the historical vehicle state, which means the left lane, the middle lane or the right lane the vehicle was in previous states. The probabilities of historical state are represented as $p_{l,hs}$, $p_{m,hs}$, and $p_{r,hs}$. Therefore, the integrated probabilities of lanes are:

$$p_l = k_{lc} p_{l,lc} + k_{mk} p_{l,mk} + k_{hs} p_{l,hs}, \tag{19}$$

$$p_m = k_{lc} p_{m,lc} + k_{mk} p_{m,mk} + k_{hs} p_{m,hs}, \tag{20}$$

$$p_r = k_{lc} p_{r,lc} + k_{mk} p_{r,mk} + k_{hs} p_{r,hs}, \tag{21}$$

where k_{lc}, k_{mk} and k_{hs} are coefficients of the three cues, used to tune the importance of different cues. The current lane is selected according to the maximum value of p_l, p_m and p_r.

V. Marking Based Vehicle Localization

When both vision-based and map-based lane markings of current lane is obtained, vehicle positions are optimized using a multi-kernel based estimation method. Fig. 4 shows the vehicle localization procedure. On one hand, when the current "cell" is selected in the map in Section IV, the markings of current cell and the cell in front are projected to BEV space, as shown the black lines in Fig. 4. Markings of a cell are a pair of paralleled lines in map topology, so the map-based markings are combination of several straight lines. The sets of left and right marking pixels are denoted as $S_{l,ipm}$ and $S_{r,ipm}$ respectively. On the other hand, vision-based lane markings are represented as the form of quadratic model $(c_i^*, d_i^*, e_i^*), i = l, r$ from (2) and (7), as shown the gray curves in Fig. 4. The translation from $(c_i^*, d_i^*, e_i^*), i = l, r$ to $S_{l,ipm}$ and $S_{r,ipm}$ can be regarded as the relative rigid transformation from rough GPS positioning to positions according to map information.

However, it is difficult to compute a translation matrix from a model (vision-based markings) to a set of pixels (map-based markings). The idea is to at first estimate the translation from the set of pixels to the marking model. The required matrix is then derived using inverse matrix transform.

A translation matrix from map-based markings to vision-based lane markings are estimated according to a multi-kernel based estimation method, the same method to determine marking parameters in Section II. The translation matrix is defined as $T_{loc}(\triangle x_{loc}, \triangle\theta_{loc})$, where $\triangle x_{loc}$ is lateral displacement, and $\triangle\theta_{loc}$ is vehicle rotation. A multi-kernel based descriptor $G_p(c_i^*, d_i^*, e_i^*, x, y)$, defined in (3), is introduced to describe the "distance" between a single pixel on a map-based marking and a vision-based marking model. The optimized

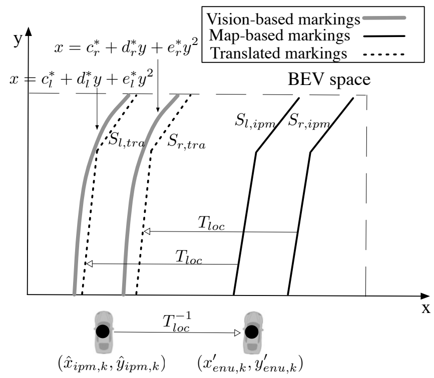


Fig. 4. Vehicle localization procedure.

lateral displacement $\triangle x_{loc}^*$ and rotation $\triangle\theta_{loc}^*$ are optimized according to $G_p(c_i^*, d_i^*, e_i^*, x, y)$ as

$$(\triangle x_{loc}^*,\triangle\theta_{loc}^*)=\arg\max_{\triangle x_{loc},\triangle\theta_{loc}}[\sum_{i=l,r}\sum_{(x,y)\in S_{i,tra}} G_p(c_i^*, d_i^*, e_i^*, x, y)], \quad (22)$$

where

$$S_{i,tra}=\{(x,y)|(x,y,1)=(x',y',1)\cdot T_{loc},(x',y')\in S_{i,ipm}\}. \quad (23)$$

Translated markings through $T_{loc}(\triangle x_{loc}^*, \triangle\theta_{loc}^*)$ are marked as dashed lines in Fig. 4. The inverse matrix $T_{loc}^{-1}(\triangle x_{loc}^*, \triangle\theta_{loc}^*)$ is the translation of lane marking based vehicle localization, from vehicle position filtered by particle filter $(\hat{x}_{ipm,k}, \hat{y}_{ipm,k})$ to the adjusted vehicle fix $(x'_{ipm,k}, y'_{ipm,k})$. So $(x'_{ipm,k}, y'_{ipm,k})$ is computed as

$$(x'_{ipm,k}, y'_{ipm,k}, 1)=(\hat{x}_{ipm,k}, \hat{y}_{ipm,k}, 1)\cdot T_{loc}^{-1}(\triangle x_{loc}^*,\triangle\theta_{loc}^*). \quad (24)$$

Therefore, the marking based localized position is $(x'_{enu,k}, y'_{enu,k})$, translating $\left(x'_{ipm,k}, y'_{ipm,k}\right)$ from BEV space to ENU space.

VI. Results

The proposed method is experimentally validated using the data from an intelligent vehicle platform provided by KITTI [17]. Since, the considered dataset was acquired using a high-end positioning system, on-road vehicle environment perturbations were modeled by adding uniform distribution noises to the corresponding vehicle fix, speed and yaw angle measurements. A uniform distribution based noise of 10 meters on both horizontal and vertical dimensions [18] are added on the ground truth vehicle positions, performing as input rough GPS data of our method. The vehicle velocity and yaw angle measurements are noised with uniform distributions of 10 m/s and 5°respectively.

Fig. 5 illustrates the result of lane selection. In lane selection part, the coefficients in Eq. (19) to (21) are set as $k_{lc} = 0.60$, $k_{mk} = 0.24$ and $k_{hs} = 0.16$, because lane changing behaviors effect lane selection decision most deeply, while the history states play the slightest role to choose the current lane. The ground truth states are labeled manually according to vision images. Original state is the state of the nearest cell to the vehicle, and filtered state is the state estimated from multi-criterion lane selection. The success ratio of original states is 52.42%, while the success ratio of filtered states using

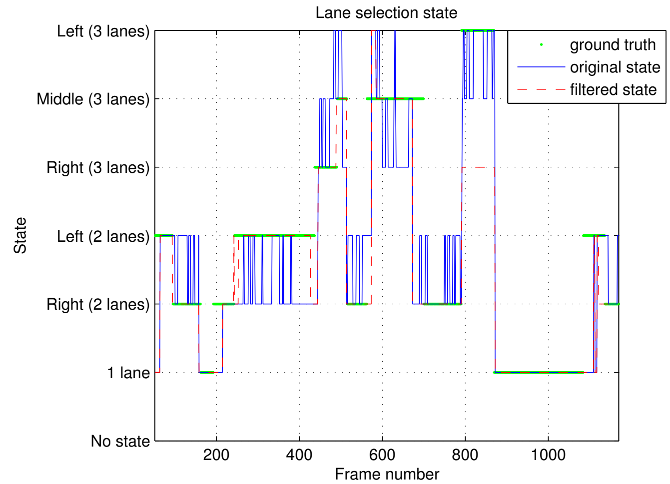
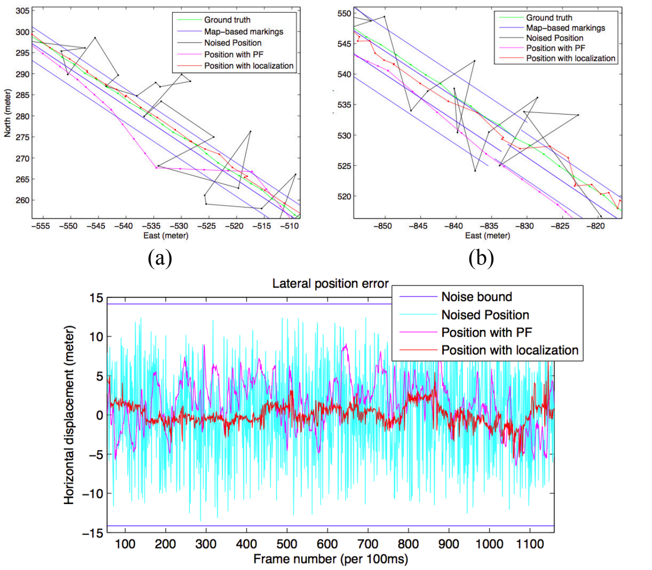


Fig. 5. Lane selection result.

lane selection increases to 78.23%. One critical reason of unsuccessful lane selection is that the vehicle position after particle filter is away from the ground truth position along the vehicle forward direction, which leads to a “delay”. For instance, at around frame 200, the benchmark position already drives from a one lane segment to a two lane segment, but the filtered vehicle position is still in one lane segment. This “delay” lasts until the filtered vehicle position drives to two lane segment. Another potential reason is noise disturb. At frames around number 800 to 850, road barriers are detected as a third left lane marking by mistake, resulting in a wrong lane selection judgment.

The results of particle filter and marking based localization are depicted in Fig. 6. Fig. 6(a)-(b) are two zoomed map areas. In Fig. 6(a), the pink curve is the vehicle positions filtered from rough GNSS signals (black), using particle filter. But this pink curve is not in the road cell area which the vehicle is in. The vehicle positions on pink curve is used to select road cell according to multi-criterion. When the road cell is determined, marking based localization is implemented, the vehicle position is adjusted to the central area of road, as red curve in Fig. 6(a). Fig. 6(b) depicts an exceptional example, a mismatch occurred between two sources, benchmark positions from KITTI and map information from OSM. In this example, the ground truth GPS data is in the middle of two lanes, but in the vision, the vehicle is in the middle lane, obviously, at least one source is not accurate. This mismatch leads to an error even the marking based method is utilized. Fig. 6(c) is a numerical comparison on lateral displacement errors, among noised GPS measurements (cyan), vehicle position after particle filter (pink) and marking-based position (red). In Fig. 6(c), both the pink and red curves are included in noise error bound. And the red curve is closer to ground truth than the pink curve, which implies that marking based localization improves particle filter based localization.

Table II provides the performance metrics of localization results. In this table, the mean value of both errors are far below the noise error bound (14.2m). The maximum error of



(a) (b) (c)

Fig. 6. Localization result. (a)-(b) are zoomed areas in map space. (c) is lateral position error.

TABLE II
ERROR STATISTICS.

	Lateral position error after particle filter	Lateral position error after marking based localization
Mean value	1.884m	0.089m
MAE	2.867m	1.006m
Standard deviation	2.942m	1.284m
Max	9.083m	5.429m
95th percentile	6.345m	2.589m

particle filter (9.083m) does not exceed the noise error bound neither. Comparing position errors of the two methods, all the statistics of marking based method are less than those of particle filter, which numerically proves that marking based localization helps to improve the performance of vehicle fix.

The method is run on a laptop using C++. The average processing time per frame is 42.9ms, including marking detection and localization. In vision-based marking detection part, the average run time per frame is 22.7ms, accompanied by the maximum time 62ms. In vehicle localization part, the mean time is 20.2ms, and the maximum processing time of a single frame is 25.3ms. The sum of maximum run time in both parts is 87.3ms, which is less than the time cycle of KITTI database frames (100ms). Therefore, the proposed method is adequate to run in real-time.

VII. CONCLUSIONS

A lane marking based vehicle localization technique, exploiting rough GPS, speed and yaw angle measurements and an open source map, has been demonstrated and experimentally validated. The results verifies a real-time and precise vehicle localization. In vision-based lane marking detection part, a parametric Gaussian model based initialization and multi-kernel based estimation provide lane markings of current lane where the host vehicle is. In map-based marking selection part, a "lane marking" map topology is created according to OSM database at first. Then the lane markings of the current lane are selected through a multi-criterion method. The vehicle position is adjusted in two steps. Firstly, a particle filter is designed to adjust the rough vehicle position. Secondly, vision-based markings and map-based markings are fused to enhance vehicle position, using a multi-kernel estimation method.

Future work focuses on creating the initial identification of vehicle position, perfecting map reconfiguration at road connections and road branches, developing the confidence indicators for estimated vehicle position, as well as studying on data mismatch between KITTI and OSM.

REFERENCES

[1] I. Skog and P. Handel, "In-car positioning and navigation technologies—a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 1, pp. 4–21, March 2009.

[2] N. Sunderhauf, M. Obst, S. Lange, G. Wanielik, and P. Protzel, "Switchable constraints and incremental smoothing for online mitigation of non-line-of-sight and multipath effects," in *IEEE Intelligent Vehicles Symposium*, June 2013, pp. 262–268.

[3] K. Redmill, T. Kitajima, and U. Ozguner, "Dgps/ins integrated positioning for control of automated vehicle," in *IEEE Conference on Intelligent Transportation Systems*, 2001, pp. 172–178.

[4] M. A. Quddus, R. B. Nolanda, and W. Y. Ochienga, "A high accuracy fuzzy logic based map matching algorithm for road transport," *J Intell Transport Syst Tech Plann Oper*, vol. 10, no. 3, pp. 103–115, 2006.

[5] I. Miller, M. Campbell, and D. Huttenlocher, "Map-aided localization in sparse global positioning system environments using vision and particle filtering," *Journal of Field Robotics*, vol. 28, no. 5, pp. 619–643, September 2011.

[6] M. E. E. Najjar and P. Bonnifait, "A road-matching method for precise vehicle localization using belief theory and kalman filtering," *Autonomous Robots*, vol. 19, no. 2, pp. 173–191, 2005.

[7] A. Barth, J. Siegemund, and J. Schwehr, "Fast and precise localization at stop intersections," in *IEEE Intelligent Vehicles Symposium Workshops*, June 2013, pp. 75–80.

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

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

[10] E. Seignez, M. Kieffer, A. Lambert, E. Walter, and T. Maurin, "Real-time bounded-error state estimation for vehicle tracking," *The International Journal of Robotics Research*, vol. 28, no. 1, pp. 34–48, January 2009.

[11] S. A. Rodríguez F, V. Fremont, P. Bonnifait, and V. Cherfaoui, "An embedded multi-modal system for object localization and tracking," *IEEE Intelligent Transportation Systems Magazine*, vol. 4, no. 4, pp. 42–53, November 2012.

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

[13] W. Lu, S. Rodriguez, E. Seignez, and R. Reynaud, "Monocular multi-kernel based lane marking detection," in *IEEE 4th Annual International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, June 2014.

[14] R. Jiang, R. Klette, T. Vaudrey, and S. Wang, "Lane detection and tracking using a new lane model and distance transform," *Machine Vision and Applications*, vol. 22, no. 4, pp. 721–737, June 2011.

[15] G. Liu, F. Worgotter, and I. Markelic, "Lane shape estimation using a partitioned particle filter for autonomous driving," in *Proc. IEEE International Conf. on Robotics and Automation*, May 2011, pp. 1627–1633.

[16] X. Liu, Q. Song, and P. Li, "A parabolic detection algorithm based on kernel density estimation," in *Emerging Intelligent Computing Technology and Applications*, 2009, pp. 405–412.

[17] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, September 2013.

[18] S. Miura and S. Kamijo, "Gps error correction by multipath adaptation," *International Journal of Intelligent Transportation Systems Research*, January 2014.