Backing up GPS in Urban Areas using a Scanning Laser

Maged Jabbour and Philippe Bonnifait
Heudiasyc UMR CNRS 6599, Université de Technologie de Compiègne, France
{maged.jabbour, philippe.bonnifait}@hds.utc.fr

***Abstract* – This paper studies the use of lidar for the ego-localization of car-like vehicles in conjunction with GPS. We consider a laser scanner installed at the front of a vehicle that detects both sides of the road. We present a method that fuses the lidar information to improve the localization process by providing additional exteroceptive information and by rejecting bad GPS fixes. The strategy is inspired by a SLAM paradigm and is efficient if the vehicle navigates often in the same area. We present the different main stages of such a strategy: lidar data processing, map data representation, and augmented Kalman filtering scheme. Finally, experimental results are reported to illustrate the performance of this approach.**

I. Introduction

Intelligent vehicles are vehicles that are equipped with exteroceptive sensors like cameras or lidars also called laser scanners to do obstacle avoidance, advanced driving assistance, autonomous navigation or platooning. For these functions, precise and reliable localization is often a key issue. For instance, localization can help pedestrian tracking by providing a good heading estimate. It is also a basic need for path planning and can help intelligent perception. On the contrary, exteroceptive sensors can help global localization. Indeed, in urban areas, high-rise buildings and monuments block often the GPS satellites and induce multipath [5].

In this work, we propose to study how a laser range finder that detects road edges can improve global localization in terms of:

- Availability, by backing up GPS in case of complete outage.
- Accuracy, by providing additional information.
- Integrity, by detecting bad measurements.

We consider a lidar installed on the hood of the vehicle that detects both sides of the road to improve the localization process, particularly efficient if the vehicle navigates often in the same area.

Here, the lidar is not used as an optic odometric sensor that provides information of the relative movement of the vehicle (like in [6]) or as an external sensor used to reject GPS outliers [8], but instead as mean to measure distances and angles on natural features of the environment and to build a model of it, while the vehicle navigates. This is a strategy inspired by the paradigm of SLAM (Simultaneous Localization and Map Building) very popular in robotics. In practice, a learning stage takes place, where the lidar is used to build an initial map of the environment using a hybridized localization that fuses GPS fixes, proprioceptive sensors using an Extended Kalman Filter. The built probabilistic map can be shared afterwards with other navigating vehicles. In later stages and when navigating in the same area, the vehicle uses this map for its current navigation needs, while simultaneously updating the features of the map. As the vehicle passes several times in the same areas, the accuracy of the map increases along with the accuracy of the vehicle pose.

Using a laser scanner to detect natural features in outdoor conditions is a difficult task because

- The reflectivity of the laser beam depends a lot on the surface of the detected object,
- The opening angle of the laser beam can induce several echos,
- Multipath can occur.

For these reasons, we study in this paper a method to detect efficiently only the road edges (i.e. the curbs in urban conditions) and that provides few incorrect detections (also called Missed Detections). Indeed, there are many situations that induce difficult perception conditions. For instance, the equipped vehicle can be overtaken or cars can be parked and hide the road edges.

The paper is organised in five technical sections. In the next section, we recall briefly the concepts of SLAM. Then, we present in section 3 a method that is able to detect simultaneously the road edges, if they exist in the scan. In section 4, we propose a map structure that handles segmented data for compression and high detection purposes. In section 5, the global localization method is described. In section 6, experimental results indicate that 1) the dead-reckoning lateral accuracy can be significantly enhanced by such an approach when a total GPS mask occurs 2) the method is able to detect and filter out GPS outliers resulting from multipath.

II. Simultaneous Localization And Mapping Concepts

SLAM (Simultaneous Localization And Mapping) [1] relies on estimating the pose (position and attitude) of a vehicle while building incrementally, at the same time, an

estimate of the coordinates of landmarks. SLAM techniques need necessarily exteroceptive sensors and often take benefice of dead-reckoning sensors to measure the relative displacements of the vehicle. SLAM looks like a chicken and egg problem but it can be shown [1] that the problem is convergent: when the vehicle navigates in a known area the location and mapping are improved. New works have shown that it is also possible to SLAM with natural outdoor features without needing to instrument the infrastructure with artificial landmarks [7].

We recall in this section what is called recursive Bayesian SLAM that is often implemented using Extended Kalman Filtering. SLAM can be formalized like estimating an a posteriori probability density function:

$$p(s_k, m_k \mid z_{1..k}) \tag{1}$$

Where s_k is the pose of the vehicle at time k, m_k is an estimate of the map representing the environment and $z_{1..k}$ represents sensors observations from the beginning of the process.

In SLAM, the system state is augmented: the vehicle pose is augmented by the landmarks coordinates (the map). Let x be this augmented state. At a time k, this can be written like the following:

$$x_k = \begin{bmatrix} s_k \\ m_k \end{bmatrix} \tag{2}$$

The evolution model of the augmented state has typically the following form:

$$\begin{bmatrix} s_k \\ m_k \end{bmatrix} = \begin{bmatrix} f(s_{k-1}, u_k) \\ m_{k-1} \end{bmatrix} \tag{3}$$

Where u_k represents the system input that often corresponds to dead-reckoning sensors. Writing $m_k = m_{k-1}$ indicates that the map is static (it doesn't move in the evolution model). To simplify, we will omit the index k in the following.

By convention in what follows, when considering the augmented state containing the state s and the map m, we will replace each model by its associated augmented model noted with a « ~ ». So $\tilde{f}$ and $\tilde{h}$ represent respectively the evolution and the observation models of the augmented state. We can then write:

$$x_k = \tilde{f}(x_{k-1}, u_k) \tag{4}$$

Exteroceptive observations always involve pose and landmarks coordinates, so:

$$z_k = h(s_k, m) = \tilde{h}(x_k) \tag{5}$$

The standard state filtering SLAM algorithm is described by Table 1. Please let us briefly discuss the key stages of SLAM. At step E.5, the problem is to associate the detected features with the known features. By using those that have been associated, the full state is updated, i.e. the accuracy of pose and the map decreases. At step E.12, new landmarks are added in the map. Their covariance (i.e. accuracy) can be computed by the presented formula.

One can notice that, as the vehicle moves, the size of the map increases. In practice, the system needs always to manage local maps to handle the complexity issues [4].

Function $[s_k, m] = SLAM(s_{k-1}, z_k, m)$

E.1 $\hat{x}_{k|k-1} = \tilde{f}(\hat{x}_{k-1|k-1}, u_k)$ // *Prediction*

E.2 $F_x = \left[\frac{\partial \tilde{f}}{\partial x}(\hat{x}_{k-1|k-1}, u_k)\right]$ // *Jacobian*

E.3 $F_u = \left[\frac{\partial \tilde{f}}{\partial u}(\hat{x}_{k-1|k-1}, u_k)\right]$ // *Jacobian*

E.4 $P_{k|k-1} = F_x.P_{k-1|k-1}.F_x^T + F_u.Q_u.F_u^T + Q$ // *Prediction*

E.5 $(a,b) \leftarrow Association(s_k, z_k, m)$ // *Data Association*

E.6 $\mu_k = z_{k,a} - \hat{z}_{k,a}$ // *Innovation*

E.7 $R_a \leftarrow z_{k,a}$ // *Observation Covariance*

E.8 $H_x = \left[\frac{\partial \tilde{h}}{\partial x}(\hat{x}_{k|k-1})\right]$ // *Jacobian*

E.9 $K_k = H_x.P_{k|k-1}.(H_x.P_{k|k-1}.H_x^T + R_a)^{-1}$ // *Gain*

E.10 $\hat{x}_{k|k} = x_{k|k-1} + K_k.\mu_k$ // *State update*

E.11 $P_{k|k} = (I - K_k.H_k).P_{k|k-1}$ // *Covariance update*

E.12 $\hat{x}_{k|k} = \begin{bmatrix} \hat{x}_{k|k} \\ \tilde{g}(\hat{x}_{k|k}, z_b) \end{bmatrix}$ // *Mapping of landmarks*

E.13 $G_x = \left[\frac{\partial \tilde{g}}{\partial x}(\hat{x}_{k|k}, z_b)\right]$ $G_b = \left[\frac{\partial \tilde{g}}{\partial z_b}(\hat{x}_{k|k}, z_b)\right]$ // *Jacobian*

E.15 $P_{m_b} = G_x.P_{k|k}.G_x^T + G_b.P_b.G_b^T$ // *Covariance*

E.16 $P_{m_b x} = P_{k|k}.G_x^T$ // *Inter-covariance*

E.17 $P_{k|k} = \begin{bmatrix} P_{k|k} & P_{m_b x}^T \\ P_{m_b x} & P_{m_b} \end{bmatrix}$ // *Augmented Variance*

End **Algorithm**

Table 1 - *Standard SLAM algorithm using state filtering*

In our experiments presented in section VI, we will present the performance of an adaptation of SLAM using road edges detected by a lidar. The landmarks being segments and not points like in usual SLAM makes the SLAM problem more difficult to solve: the landmarks are not directly observed. A solution to a similar problem called "delayed" SLAM has been recently presented in [9]. Nevertheless, since this approach would be memory consuming in our case, we have split the SLAM problem in two stages: a learning stage (rough mapping only) and a navigation stage (localization and map updating).

III. DETECTING ROAD EDGES WITH A LIDAR

Laser range finders use generally a laser beam that scans the environment in a plan. In order to be sure to

detect both sidewalk edges, a SICK LMS 291 laser has been installed on the hood of the vehicle (see Fig. 1). One can notice that the measured data can also be exploited for obstacle detection (e.g. cars or pedestrians).

Fig. *1. LMS SICK 291 on the hood of the vehicle*

The lidar processing consists in two stages (see Fig. 2):

- A geometrical transformation that projects the detection in a 2D horizontal frame.
- A processing of the laser signal in order to extract the sidewalk edges, if they exist, in the scan view.

With a frequency of 75 Hz, the lidar sends sentences of 181 values, each one corresponding to a bearing angle respectively from 0 to 180 degrees. In our test, we chose a resolution of 1° and a range of 81m.

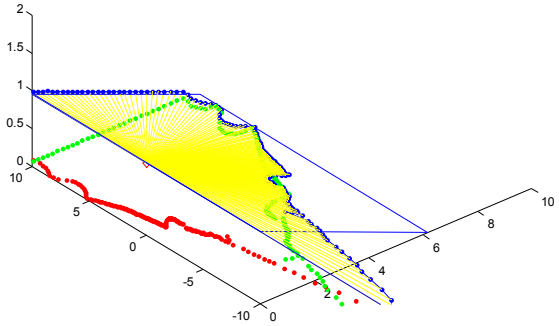

Fig. *2. 3D visualisation of the laser signal (real data)*

Let consider the projection of the scan on the horizontal plan, since we do the process in a 2D local frame tangent to the Earth.

Let $(x_i, y_i, 0)$ be the coordinates of a point P detected by the lidar in its own scanning plan at a distance R with a bearing angle β:

$$\begin{cases} x_i &= R.\sin(\beta) \\ y_i &= R.\cos(\beta) \\ z_i &= 0 \end{cases} \quad (6)$$

We are looking at transforming the coordinates of P in frame R_F attached to the front of the car. Let h and d two constants:

-h is the laser height from the ground.

-d is the wheel radius.

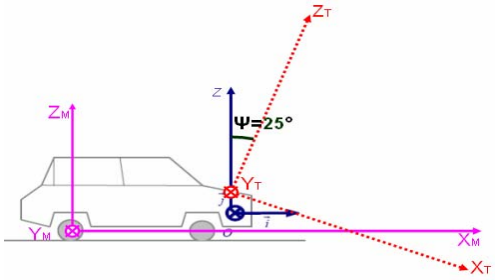

Fig. *3. Frames: subscribe M for Mobile, T for the lidar. The intermediate frame R_F (in blue) corresponds to a pure translation of R_M.*

The transformation of the coordinates of point P in R_F is given by:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_F = {}^F R_T * \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}_T + Trans(z) \quad (7)$$

Where R_T =Rot(y, Ψ) is a tilt angle rotation around the y-axis (Ψ is known after calibration):

$$R_T = \begin{pmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\psi \end{pmatrix} \text{ and } trans(z) = \begin{pmatrix} 0 \\ 0 \\ h-d \end{pmatrix} \quad (8)$$

Afterwards, the coordinated of point P in R_F are the following:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_F = \begin{pmatrix} x_i.\cos\psi \\ y_i \\ -x_i\sin\psi + h - d \end{pmatrix} = \begin{pmatrix} R.\sin\beta.\cos\psi \\ R\cos\beta \\ -R\sin\beta.\sin\psi + h - d \end{pmatrix} \quad (9)$$

The edge detection is done by derivating the evolution of z as a function y in the intermediate frame. Figure 4 shows the detection result on both sides: the upper curve shows the projection of telemetric signal in the plan (yz), and the lower curve shows the derivative of this signal. The two maximum values (right) and minimum (left) show the two detected roadsides. This simple processing strategy gives experimentally good results by detecting often candidates. An outlier rejection is performed by comparing the measurement height with the one of a usual sidewalk and comparing locally the detection with the signature of a road edge. This is useful for rejecting data belonging to objects like vehicles, pedestrians and roadside equipment since their height is bigger than that of standard sidewalk and since they are rarely similar to an edge signature.

More advanced treatments can be used for more noisy perception conditions. For instance, in [2] the authors use a Kalman filter applied on the 1-D lidar scan data to detect a road boundary when the innovation is higher than a threshold. In [8], a classification approach is proposed to differentiate between vehicle and non-vehicle lidar targets.

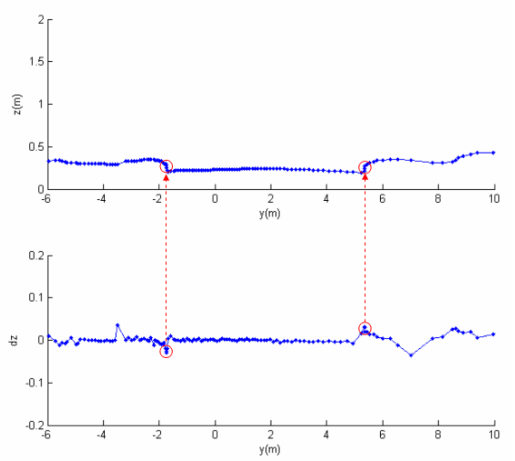

Fig. *4. Sidewalk edges detected on both sides of the road*

There are 4 detection cases that can be isolated:

- There is a roadside edge and it has been detected (ideal case)
- There is a sidewalk edge, and it has not been detected (a Missed Detection MD).
- There is no sidewalk, and the detection algorithm detected nothing (ideal case)
- There is no sidewalk, and the detection algorithm gave positive result (False Detection FD).

An ideal robust algorithm is the one who gives results corresponding to case 1 and case 3, and avoids being in case 2 and case 4.

IV. DATA SEGMENTATION AND MAP BUILDING

The map that we consider contains the road boundaries. It is built based on the current estimation of the vehicle pose and on the information provided by the algorithms detecting the road edges from the lidar data. A first idea could be to store all sidewalk impacts. Nevertheless, this would induce a large amount of data and, during the next passages, the car would not detect exactly the same landmarks. For these two reasons, the map information is segmented.

We used a split-and-merge segmentation algorithm [3]. When navigating in a new area, the vehicle buffers the lidar data. When a discontinuity in the lidar data on one side is detected, the corresponding buffered data is segmented. Figures 5 and 6 illustrate this mechanism. A crucial issue relies is the fact that a road boundary does not always exist because of junctions, for instance. For this reason and in order to manage properly the environment map, a hybrid (metric and topological) representation is used. The metric nature comes from the segmented coordinates in the state vector and the topological nature is due to the management of the connectedness of the different nodes. An interesting characteristic of a road boundaries map is that each node can be linked to two others, at the maximum.

Because of noises, outages and missed detections, the connectedness between nodes has to be handled in a probabilistic way. We propose to use confidence scores associated to every link. They can be interpreted as likelihoods of the topological connections. The method we have developed to estimate these scores relies on several heuristic criteria which depend on the distance between the two extreme points of the buffered block, the number of raw points and the length of each obtained segment. Please also notice that these scores are updated when navigating in a learn area.

Suppose that at a time *k*, only the part 1 to 7 is built (detected and segmented) (see figure 5). Table 2 gives an example of the hybrid representation that we use. Figure 6 shows the map building process later on.

Node ID	Linked to	Weight or connectedness indicator ($0<w_{ij}<1$)
1	0	N/A
	2	W_{12}
2	1	W_{12}
	3	W_{23}
3	2	W_{23}
	4	W_{34}
4	3	W_{34}
	5	w_{45} (= 0)
5	4	w_{45} (= 0)
	0	N/A
6	0	N/A
	7	W_{67}
7	6	W_{67}
	0	N/A

Table 2 - *Example of the topological representation of the map at time k*

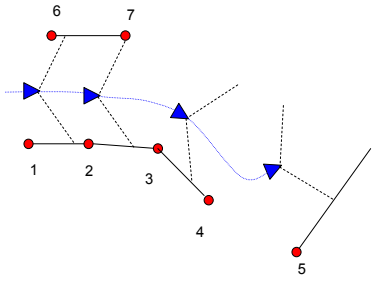

Fig. *5. The map at time k. The triangle represents the vehicle, the doted lines, the laser beams and the trajectory.*

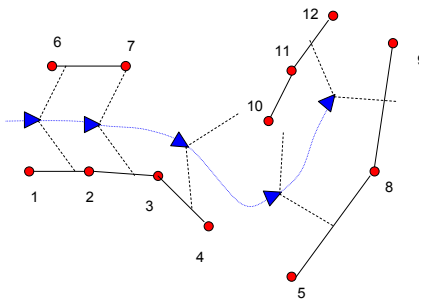

Fig. *6. The map later on.*

V. Enhanced Localization Using SLAM

Each time the vehicle navigates in a known area, the localization and the map are improved. This SLAM approach is done by an Extended Kalman Filter like in Table 1.

In the evolution step (steps [E.1..E4] in Table 1), the proprioceptive sensors of the vehicle (odometers on the rear wheels and gyro) are used to estimate the vehicle pose (see section II):

$$s_{k|k-1} = f(s_{k-1|k-1}, u_k, \gamma_k) + \alpha_k \tag{10}$$

Where $s_{k|k-1}$ is the vehicle's state vector at instant k, composed of (x_k, y_k, θ_k), u_k the vector of the measured inputs consisting of (Δ_k, w_k); Δ_k and w_k being respectively the elementary distance covered by the rear wheels and the elementary rotation of the mobile frame. α_k is the process noise and γ_k represents the measurement error of the inputs. α_k and γ_k are assumed to be uncorrelated and zero mean noise.

If the road is perfectly planar and horizontal, and if the motion is locally circular, the evolution model can be expressed by:

$$\begin{cases} x_{k+1} = x_k + \Delta_k . cos\left(\theta_k + \frac{w_k}{2}\right) \\ y_{k+1} = y_k + \Delta_k . sin\left(\theta_k + \frac{w_k}{2}\right) \\ \theta_{k+1} = \theta_k + w_k \end{cases} \tag{11}$$

The values of Δ_k and w_k are computed using the ABS measurements of the rear wheels and a fibre optic yaw rate gyroscope.

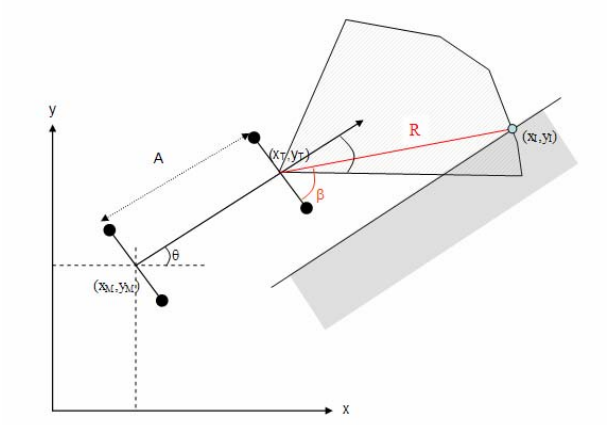

Fig. *7. Coordinates and lidar measurements*

Using the exteroceptive sensors (the GPS and the lidar), it is now possible to estimate both the map and the vehicle pose. The observations are the 2D coordinates of the GPS fix and the distance and heading to both detected sidewalk edges (see Fig. 7). This stage corresponds to the steps [E.5..E.11] of Table 1.

$$Y = \left[x_{GPS} \quad y_{GPS} \quad R_{laserL} \quad \beta_{laserL} \quad R_{laserR} \quad \beta_{laserR}\right]^T \tag{12}$$

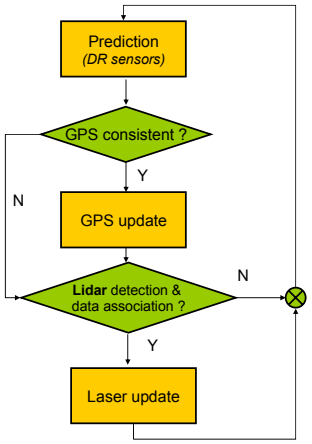

Fig. 8. *Diagram of the SLAM process using GPS and lidar*

As there are two sources of observation, we chose to serialize the updating stages under the assumption that the noises are uncorrelated (Fig. *8*). If the GPS data is present and consistent a first update stage of the global state vector is done. Then, if a valid laser data corresponding to a sidewalk edge is present, a second updating stage occurs. The laser update can be repeated twice as 2 roadside edges can be detected at each scan. Please note that the data association between the detected sidewalks and the existent map is done using a nearest neighbour algorithm.

The consistency tests are performed using innovation gates: if the norm of the squared innovation vector is higher than a threshold (depending on a false alarm probability), the measurement is considered as doubtful and not used in the filter. This strategy improves the integrity of the localizer since it can reject aberrant GPS fixes (due to multipath for instance) and aberrant road boundary detections.

VI. Experimental Results

We present, in this section, the behaviour of the system during the SLAM stage.

Experiments have been performed in Compiègne using a KVH fibre optic gyro, an odometer input, a Trimble AgGPS 132 (L1-only receiver) and a laser range finder LMS SICK 291.

In order to compute a localization error, a Post Processed Kinematic (PPK) GPS receiver (a Trimble 5700 dual-frequencies L1/L2) has been used with an offline computing software (Trimble Total Control - TTC). In order to simplify the computation, the Ag132 and the 5700 receivers shared the same antenna. TTC software uses the 5700 raw data recorded at 1 Hz during the trials. In order to get a better accuracy, the precise GPS ephemeris data (SP3) have been downloaded from IGS (International GNSS service). We have also collected the ionosphere (IONEX).

For the errors computation, only the PPK positions which had centimetre-level accuracy (known thanks to the observation of the residuals) were used as reference positions.

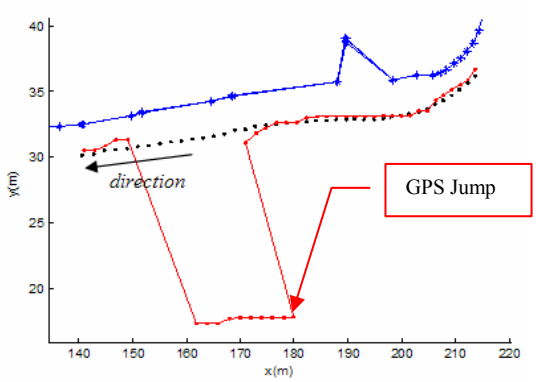

Fig. *9. System robustness to a GPS jump due to a change of satellite visibility*

Figure 9 shows that the SLAM process (in dotted black) is robust to a GPS jump (in thin). In such a situation, due to the change of the satellite visibility, the receiver computes the navigation solution from a 3-D mode to a 2-D mode, in which the altitude is supposed to be constant. The GPS jump is eliminated by a consistency test.

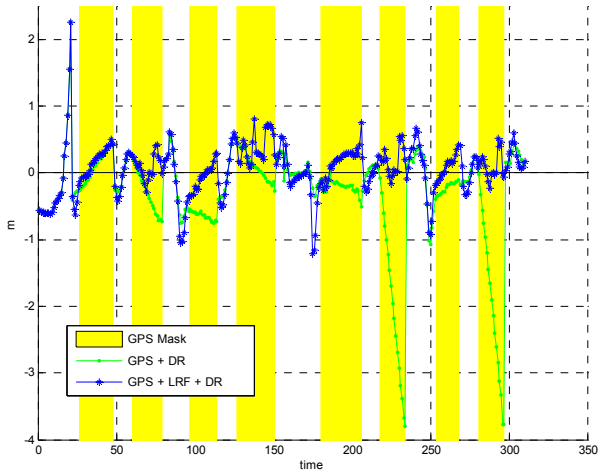

Fig. *10. Lateral localization errors*

Figure 10 shows the lateral errors of the localization algorithms. These lateral errors are computed in the vehicle frame. The curve in green represents the errors in meters of the localization algorithm using GPS coupled with DR sensors only. The one in blue represents the lateral errors of the complete system which fuses GPS, lidar, and DR sensors. The yellow bars show the GPS outages, made manually on data. It can be seen that the complete system performs better in the lateral direction. These results indicate clearly that, when there is a GPS outage, the algorithm which uses the SLAM concept is very robust to lateral drift thanks to the fusion of the lidar with the map and the DR sensors. The lateral error rarely exceeds 1 meter error, while the GPS+DR algorithm can drift significantly during GPS outages. When GPS is available, both localization processes give similar results.

VII. Conclusion

This paper has presented a method that can back up a GPS receiver in challenging environments like in urban areas. It can also improve the integrity of the navigation solution by helping to detect bad GPS fixes. In this work, the features detected by the lidar are road boundaries. As shown, they are easy to detect in a robust way and well adapted to a 2D localization in a local frame tangent to the Earth. The method we have studied needs a first learning when the vehicle navigates in an unknown area. Then, each time the vehicle evolves again in this area, the SLAM approach allows improving the accuracy of the localization while rejecting bad GPS fixes.

References

[1] M. W. M. G. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, M. Csorba, “A solution to the simultaneous localization and map building (SLAM) problem”, IEEE Trans. on Robotics and Automation, 17, 3 (June 2001), pp. 229-241.

[2] W. S. Wijesoma, K. R. S. Kodagoda, and A. Balasuriya, "Road-Boundary Detection and Tracking Using Ladar Sensing", IEEE Trans. on Robotics and Automation, VOL. 20, NO. 3, pp.456-464 June 2004.

[3] G. Borges, M.-J. Aldon, “A Split-and-Merge Segmentation Algorithm for Line Extraction in 2-D Range Images”, Int. Conf. on Pattern Recognition (ICPR'00) September 03 - 08, 2000 Barcelona, Spain.

[4] M. Jabbour, Ph. Bonnifait, V. Cherfaoui. “Enhanced Local Maps in a GIS for a Precise Localisation in Urban Areas”, 9th International IEEE Conference on Intelligent Transportation Systems (ITSC 06), Toronto, Canada, September 17-20, 2006.

[5] Y. Cui and S.S. Ge, “Autonomous Vehicle Positioning With GPS in Urban Canyon Environments”, in the IEEE Transactions on Robotics and Automation, Vol. 19, N. 1 February 2003.

[6] A. Soloviev. “What are the benefits of combining laser radar (ladar) and inertial data for navigation”. InsideGNSS, July/August 2007. pp 26-29.

[7] E. Royer, J. Bom, M. Dhome, B. Thuillot, M. Lhuillier, F. Marmoiton, “Outdoor autonomous navigation using monocular vision”. IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 3395-3400, Edmonton, Canada, Aug. 2005.

[8] B. Gao and B. Coifman. “Vehicle Identification and GPS Error Detection from a LIDAR Equipped Probe Vehicle”. IEEE Intelligent Transportation Systems Conference, Toronto, Canada, September 17-20, 2006. pp 1537- 1541.

[9] D. Ribas, P. Ridao, J. Neira, J.D. Tardós, “SLAM using an Imaging Sonar for Partially Structured Underwater Environments”, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 06), October 2006, Beijing China.