Improved Global Urban Localization Based on Road Maps and 3D Detection of Road Intersections

Lucas de Paula Veronese, Jose Guivant and Alberto Ferreira De Souza

***Abstract*— This paper presents an improved version of the localizer presented in [1]. In the previous approach, partially known road maps and low quality dead reckoning information are processed by a Monte Carlo fusion algorithm for globally localizing a vehicle in urban context under temporary or permanent denial of GPS signal. The original approach, however, could temporarily present weak observability in cases where the vehicle does not perform turning actions (e.g. changing roads) for large travelled distances. The effect of this problem is evidenced as an increase of the uncertainty in the estimated position in the longitudinal direction of the platform.**

In order to improve the observability of the estimation process, additional observations need to be provided and these observations must be informative in the subspace corresponding to the longitudinal dimension of the position. Relevant observations for that purpose can be acquired through the detection of road intersections and other infrastructure usually present in urban or other road contexts. In this paper, an extended version of the localizer presented in [1] that uses such observations is introduced. Experimental results obtained with an autonomous car are presented and show the good performance of the approach.

I. Introduction

Accurate global localization is a necessary capability of autonomous vehicles intended to operate in large areas. In many cases, the context of operation does not allow the use of GPS, or degraded GPS estimates makes its direct use inappropriate (e.g. urban canyons, hostile territories in war scenarios, underground mines, etc.). Alternative localization mechanisms, such as those based on SLAM [2]–[5], require the vehicle to “close the loop” (revisit places) for maintaining the required accuracy, what, in certain cases, is not possible because the vehicle’s plans do not actually involve revisiting places. Approaches based on detailed a priori maps ([6]–[9]) are also possible, but require building its map in advance (usually by visiting and learning the context of operation).

One viable alternative can be implemented using information publically available in the form of road maps and on-board sensing capabilities that comes today as standard in most cars. These sensing capabilities allow easy implementation of dead-reckoning (at least low quality dead-reckoning) and of some level of environment modelling.

Based on those resources, a Bayesian estimation process can be performed in order to estimate the global pose (position and heading) of the platform. In the estimation process, the kinematic model of the vehicle can be used as a Process Model. The road map is used to define the main observation model (i.e. a likelihood function), based on the assumption that the vehicle is usually located on a known road. The additional sensing capabilities can offer extra observations for improving the convergence of the estimation process (what is the case presented in this work).

As the generated Probability Density Function (PDF) for describing the estimates (2D position or 3DoF Pose) is usually non Gaussian, the Bayesian estimation process is usually implemented through approaches such as Particle Filter (PF) or Sum of Gaussians (SoG). An approach for implementing such an estimation process was introduced in [1] and [10]. In that approach, an extended likelihood function was defined to improve the performance of the PF estimation process. The approach was adequate for treating cases where the road map was only partially known (e.g. incomplete). However, it was noted that the quality of the estimates could be degraded in certain operation conditions. Those conditions occur when the vehicle travels for large distances throughout almost rectilinear roads, i.e. when it does not turn at any intersection for long distances. That behavior could affect the observability of the estimation process, increasing the uncertainty on the longitudinal component of the position estimates.

Those situations suggest the need of using additional observations that could provide information related to the longitudinal component of the vehicle position. Particular cases where such observations are possible occur in the vicinity of known road intersections, even when the vehicle does not turn and keep going straight. From the detection of such cases, an associated likelihood function could be generated and used for updating the estimation process. In this paper, a new class of observation based on the detection of intersections in road maps is defined, and the verified improvements in our previous algorithm [1] are shown via experiments with an autonomous car.

II. Constrained Localization in Segment-Based Maps

In this approach, the 2D global localization of the platform is treated as a Bayesian estimation problem; where the estimated variable is the pose of the vehicle. The platform’s pose at time k is expressed through the 3 DoF vector

$$X_k = [x_k, y_k, \phi_k]^T \qquad (1)$$

then the localization problem implies the recursive synthesis of a belief, represented through a PDF,

$$p_{X_k|z^{(k)}} = \left(X_k|z^{(k)}\right) \qquad (2)$$

where $z^{(k)}$ represents the sequence of all the available observations until time k.

If the posterior $p\left(X_{k-1}|z^{(k-1)}\right)$ at time $k-1$ is available, then the prior at time k (due to a prediction step) is:

Lucas de Paula Veronese and Alberto Ferreira De Souza are with the Departamento de Informática, Universidade Federal do Espírito Santo, Vitória, ES, Brazil, 29075-915, Brazil (phone: +55-27-40092138; e-mail: lucas.veronese@lcad.inf.ufes.br).

Jose Guivant are with the School of Mechanical and Manufacturing Engineering, University of New South Wales, Sydney, NSW, Australia (phone: +61-2-93859820, e-mail: j.guivant@unsw.edu.au)

This work was supported in part by the CAPES Foundation, Ministry of Education of Brazil, Brazil (grant 5557/14-3), Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq, Brazil (grants 312786/2013-1 and 552630/2011-0) and Fundação de Amparo à Pesquisa do Espírito Santo – FAPES, Brazil (grant 48511579/2009).

$$p\left(X_k|z^{(k-1)}\right) = \int p(X_k|X_{k-1}) \cdot p\left(X_{k-1}|z^{(k-1)}\right) \cdot dX_{k-1} \qquad (3)$$

where the conditional PDF $p(X_k|X_{k-1})$ is provided by the process model of the system (in this case the kinematic model of the vehicle, based on dead reckoning resources).

At time k a set of measurements, z_k, become available, consequently allowing the synthesis of a posterior PDF that is obtained through a Bayesian update stage,

$$p\left(X_k|z^{(k)}\right) \propto p\left(X_k|z^{(k-1)}\right) \cdot p(z_k|X_k) \qquad (4)$$

where $p(z_k|X_k)$ is the likelihood function associated to observation $z = z_k$.

For nonlinear non-Gaussian estimation problems, there are usually no analytic expressions for the estimated PDFs. A common approach for treating these problems is the Particle Filter (PF), that is able to estimate non-Gaussian and potentially multimodal PDFs. The generated PDF is represented through a set of samples (particles) with associated weights, and the estimates are computed based on these samples and weights.

The system for localizing the platform is based on a process model and a set of likelihood functions that allow implementing the prediction and update steps of a PF. The state of the system represents the pose of the platform, i.e. its 2D position and its heading. The process model is the kinematic model of the vehicle employed in the experiments. And, in the context of this paper, there are two definitions of likelihood functions that are associated with the road segments: the *Base Likelihood* and the *Extended Likelihood*. The Base Likelihood is intended to model the likelihood of a position, i.e. the likelihood of a point being located on a valid road, while the Extended Likelihood (or path likelihood) is the likelihood of a path coinciding with a valid road. Both definitions of likelihood function are based on the segments defined in the road map.

A. Base Likelihood Associated with the Road Map

For a set of 3DoF particles, at time k, $\left\{X_k^i, w_k^i\right\}_{i=1}^N$, the Base Likelihood (BL), $L_B\left(X_k^i\right)$, associated with a given map (set of segments) is defined as follows:

$$L_B(X) = p(map \mid X) = \max_{j=1}^{N} \left\{f\left(X, S_j, C_j\right)\right\} \qquad (5)$$

where $\left\{S_j\right\}_{j=1}^N$represents the set of segments that compose the nominal road map and $C_j$denotes the properties of segment S_j (road's width, number of lanes, lane directions, etc.). The function $f(.)$ is highly dependent on the distance between the pose component (of the state X) and the segment S_j. The function $f(.)$ may be also dependent on certain segment's properties such as its width and nominal circulation direction.

A simplified version of (5) is

$$L_B(X) = p(map \mid X) = \begin{cases} 1; if\ X \in \Omega(map, \Omega_\mathrm{k}) \\ 0;\ if\ X \notin \Omega(map, \Omega_\mathrm{k}) \end{cases} \qquad (6)$$

where the region $\Omega_\mathrm{k}$is a convex hull (usually just a rectangle) that contains the current set of particles at time k, i.e. $\left\{X_k^i\right\}_{j=1}^N$.

The region $\Omega(map, \Omega_\mathrm{k})$ defines the roads as thick bands using the segment's locations and their associated properties provided in the road map definition. $L_B(X)$ is defined just on a small convex region of interest (ROI), Ω_k, which is big enough to contain all the current particles. Through the dynamic definition of a moving and resizable ROI, the likelihood function$L_B(X)$ can be evaluated for all of the particles in real time at low computational cost.

B. Extended Likelihood

For improving the performance of the estimation process in the presence of out-of-map situations, an extended likelihood function is defined. An out-of-map situation occurs when the vehicle is traveling through unknown sections of the map, i.e. sections of the environment that are not included in the nominal road map.

Particles will tend to cluster towards regions of high Base Likelihood, which are assumed to be consistent with the map (e.g. existing roads in the nominal map). This can be an adequate behavior in cases where the map is complete and the vehicle remains on it permanently. However, in certain cases, the vehicle might travel through roads that are not present in the nominal road map. For those cases, the convergence of the localization process can be improved by using the recent history to build an *estimated dead-reckoning path*.

An estimated dead-reckoning path for each particle at time k can be efficiently computed by combining dead-reckoning (obtained from an independent estimation process) with the current value of the particle X_k^i. This estimated path is only realistic for short time horizons, as dead-reckoning is accurate only for short time horizons. So, given a particle $X_k^i = \left[x_k^i, y_k^i, \phi_k^i\right]$ and recent dead-reckoning information, an estimated dead-reckoning path (it is actually a trajectory as well, since there is also time information), $\xi_i(t')$, $t' \in [k-\tau, k]$, is synthesized, where the value τ defines some short horizon of time. This path ends exactly at the current pose (i.e. matching position and heading) of the particle, i.e. $\xi_i(k) = X_k^i$.

The estimated dead-reckoning path is usually defined in a different coordinate system, as it is the result of an independent process (it could even be expressed in a completely uncorrelated coordinate frame). One important aspect of the estimated dead-reckoning path is its quality given a specific coordinate system, i.e. if its shape is, after proper rotation and translation, similar to the real path of the vehicle. If the estimated dead-reckoning path is expressed as the path $\mu^i(t') = \left(x_\mu(t'), y_\mu(t'), \phi_\mu(t')\right)$, then the process of associating it with an individual particle and of mapping it into the global coordinate frame is performed simply by applying the rotation and translation defined by the current particle position and heading (see details in [10]).

The Extended Likelihood (EL) of a particle is given by the line integral of the BL function along the estimated dead-reckoning path:

$$L_E(X_k^i) = \int_{k-\tau}^{k} L_B(\xi_i(t')) \cdot dt', \tag{7}$$

where $L_B(\xi)$ is the BL function of the point ξ, (as it was defined in (6)).

In order to avoid the influence of the vehicle speed on the path likelihood, we evaluate BL according to the arc length parameter s, integrating over the path, (i.e., in space and not in time):

$$L_E(X_k^i) = \int_{0}^{l_s} L_B(\xi_i[s]) \cdot ds, \tag{8}$$

where $\xi_i[s]$ is the path expressed as a function of its intrinsic parameter s, and l_s is the length of path. The continuous line integral of the path can be approximated by its discrete version,

$$L_E(X_k^i) = \sum_{j=1}^{N_J} L_B(\xi_i[s_j]), \tag{9}$$

where the samples of the intrinsic parameter, s_j, are densely distributed over the estimated dead-reckoning path.

Some additional refinements can be considered for the definition of BL (5). For instance, one can consider the direction of the road. In this case, BL would not be just a function of the position, but it would depend on the vehicle heading at each point of the path. A path's segment that crosses a road would add to the likelihood if it crosses the road in the proper direction.

Figure 1 shows a synthetic example of BL (shown in gray), particles that represent the pose of the vehicle, and their associated estimated dead-reckoning paths (in cyan). The particles' positions and headings are represented by blue arrows. The red path corresponds to the most likely hypothesis (according to the Extended Likelihood measure).

By using observations that consider the estimated dead-reckoning path, the out-of-map problem is mitigated. The transition between a situation where the vehicle is on the road map to another where it is completely out of it (i.e. current pose and estimated dead-reckoning path are outside the road map) can be performed safely by using an approach based on hysteresis, as discussed in [1], [10].

In the synthetic example shown in Figure 1, the region of interest (ROI) is a square of 200 by 200 meters. This ROI is big enough to contain the current population of particles and their associated estimated dead-reckoning paths. The most likely particles are those that have a path mostly inside the road. In the case of Figure 1 (d), although all particles are located on the road (high Base Likelihood), many of their associated paths have large segments outside the zones of high Base Likelihood. In Figure 1 (b) one can see the remarkable effect that a wrong heading direction may cause. This situation is also illustrated in the Figure 1(c).

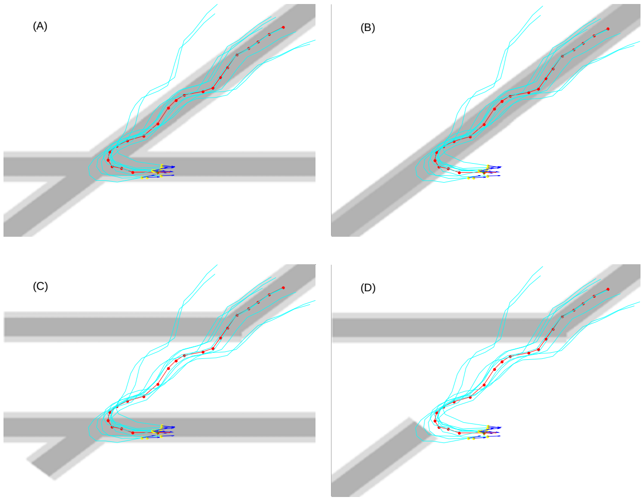


Figure 1: Synthetic examples for a ROI of 200m x 200m. The value of the Base Likelihood ($L_B(X)$) is shown in gray (the darker, the higher). A set of particles and their associated paths are included. The particle and path with maximum extended likelihood, $L_E(X_k^i)$ are shown in red. In three of the four cases, the vehicle temporarily abandons the road. In cases (b) and (c), all particles have equally low Base Likelihood; however, many particles still have high Extended Likelihood. Images taken from [10].

When the filter infers that the vehicle has been outside the map for sufficient time (i.e. no particle shows relevant part of their estimated dead-reckoning paths inside the road map), no updates are performed on the particles, i.e. the filter works in pure prediction mode. When the vehicle enters the road map again and there are enough particles with the required EL, i.e. higher than certain threshold, then the filter restart to apply updates on the particles. However, this transition typically is not immediate. There could be some delay until the required estimated dead-reckoning paths are consistent with the road map – the fact that a particle is well inside a road of the road map does not mean that its likelihood should be high. It needs a relevant fraction of its associated estimated dead-reckoning path inside roads of the road map in order to be considered "inside the road map".

C. *Likelihood Based on Real Sensing Capabilities*

The previously defined likelihood functions are not based on real measurements (provided by sensors), i.e. the likelihood functions $L_B(x)$ and $L_E(x)$ are just based on the hypothesis that the vehicle is usually travelling on the roads of a known road map.

If the vehicle is equipped with proper sensors, it might be able to infer when it is effectively on a road or not (even though the road may not be part of the road map). This inference process could be based on 3D perception capabilities, such as the ones used in this work. Additionally, by processing the 3D sensor data, the vehicle may be able to infer if it is located nearby a road intersection or not. The detection of a road intersection is highly informative because it provides rich information for reducing the uncertainty of the estimates of the longitudinal position of the vehicle.

D. *Likelihood Associated with Road Intersections*

Different to the previously defined likelihood functions, the likelihood associated with the detection of road intersections is computed using observations of the real world. These observations just say that the vehicle is currently located in the vicinity of a road intersection, usually not specifying which one is it (e.g., no Data Association is available). However, this is not a problem because, although Data Association is desirable, it is not strictly necessary for identifying which intersection the vehicle is close to, since PF can deal with multi modal likelihoods.

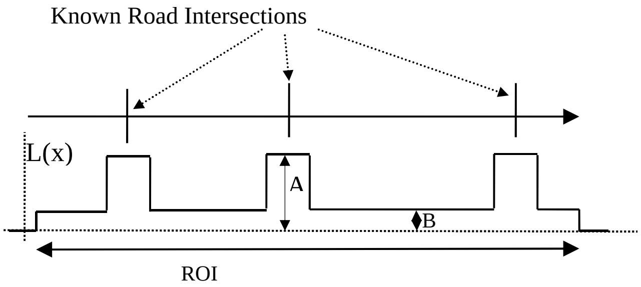


Figure 2: Likelihood $L(X)$ associated with the detection of a road intersection. $L(X)$ is equal to zero in any point outside the current ROI. Inside the ROI, it takes high values $(= A)$ at points that are close to known road intersections and low values $(= B, A \gg B > 0)$ at points far from those known intersections.

The proposed likelihood function takes in consideration the probability false positives, i.e. the detection of road intersections that are not included in the road map. A simplistic 1D example of our likelihood function is shown in Figure 2. The fact that the value of the parameter B is higher than zero implies that this likelihood considers the possibility that a detected road intersection is not on the road map.

III. Detection of Road Intersections Based on 3D Point Clouds

Usually, there are diverse features that allow detecting a road intersection, such as traffic lights, pedestrian cross lanes, and also the pattern of the sites nearby. However, some of these characteristics may vary from cases of busy avenues to quiet neighborhood streets. Additionally, those are usually different in many countries as they have different rules and roads architecture. Therefore, techniques for inferring the proximity of road intersections should be based on some "universally common" feature, such as the boundaries of the roads themselves.

A. Obstacle classification

In the current implementation, that is appropriate for the autonomous vehicle being used, the feature extractor was designed for the 3D sensing capabilities offered by the best on-board sensor, i.e. a Velodyne HDL-32E 3D laser scanner. This sensor scans the environment in 360^o of azimuth with 32 lasers that operate simultaneously, covering an angular interval of elevation of about 40^o. It produces a dense 3D point cloud per 360^o revolution, at rates of up to 20 revolutions per second.

It is possible to classify with reasonable accuracy which 3D points correspond to obstacles and, in particular, to the lateral boundaries of the road, such as curbs. For that, vertical scan lines (the sequence 32 points having constant sensor's azimuth, extracted from the 3D scans) are processed individually. Each vertical scan line is modeled as PieceWise Linear (PWL) function (as defined for 2D lasers scans in [11]). The relative angles between the segments of the PWL function are used to infer discontinuities in the slope of the terrain, as proposed in [11] and [12], and shown in Figure 3.

B. Mapping and Road Extraction

Each 3D point cloud (360^o Velodyne revolution) provides a substantial amount of data. However, one point cloud does not offer sufficient coverage in elevation. The 3D images are also affected by occlusions (e.g. due to the presence of other vehicles, etc). For this reason, the inference process needs to be based on the fusion of multiple 3D point clouds. Occupancy Grid (OG) maps provide an efficient way of fusing the data present in a sequence of 3D point clouds.

To use OG maps, the relative positions and orientations (poses) of the point clouds must be sufficiently well known. The relative pose between 3D point clouds can be obtained from short term 6DoF estimates of the vehicle pose, e.g. based on scan matching of the point clouds themselves, such as in the approaches presented in [13] and [14]. Consequently, for a short sequence of 3D point clouds, a common coordinate frame can be defined and the 3D frames can be accurately expressed within it.

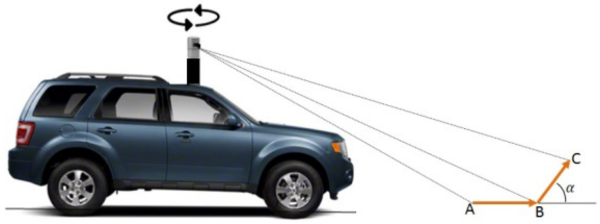


Figure 3: Detection of obstacles. A section of the PWL function of one vertical scan line that presents a discontinuity, i.e. two adjacent linear segments having substantial difference in their inclinations. The feature is shown in the longitudinal plane for the sake of simplicity. Vertical scans are taken throughout the 360^o degrees of azimuth.

Each discontinuity in the environment detected using individual scan lines is used for defining the likelihood of an obstacle. These likelihoods are mapped to a 2D plane (the floor) using a common coordinate frame, e.g. (x_i, y_i).

The process of fusing all the information could be mathematically expressed via the following function, in the 2D domain (x, y) of the common coordinate frame (the floor),

$$L_o(x_i, y_i) = \mu\big(\Delta\alpha(x_i, y_i)\big) \tag{10}$$

where the function $\mu(.)$ is a kernel (e.g. a Gaussian one). The argument $\Delta\alpha(x_i, y_i)$ is an approximation of the variation of slope, in the 2D direction given by the intersection between the horizontal plane (the floor, $z = 0$) of the local common coordinate frame and the plane of scanning for that vertical scan line.

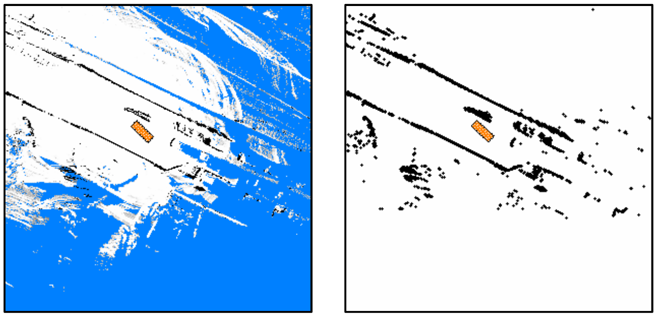
The points that belong to the interior of the PWL function segments are assumed to have constant inclination, consequently having zero variation and minimum likelihood, $L(x, y) = \mu(0)$. The advantage of using a PWL function is its high immunity to sensor noise and to terrain variations that have low amplitude and high spatial frequency, that are features of no interest for the purpose of detecting obstacles.

In the implementation used in this work, a Gaussian kernel was used, having the following expression,

$$\mu\big(\Delta\alpha(x_i, y_i)\big) = e^{-\left(\frac{\cos(\Delta\alpha(x_i,y_i))}{\sigma^2}\right)} \qquad (11)$$

where σ is a parameter of the mapping process. Then, the 1D directional likelihood about the existence of obstacles is projected to the common OG using log-odds [15]. It is important to mention that other $\mu(.)$ functions could be used, provided that small slope variations of $|\Delta\alpha(x_i, y_i)|$ result in low likelihoods and that high slope variations imply in high likelihoods. The discrete directional derivative $\Delta\alpha(x_i, y_i)$ generates values in the range $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$.

Figure 4 presents two typical grid-maps of the environment generated by the mapping process. This figure show maps of 60m x 60m, with grid resolution of 0.2m in each dimension. Figure 4(a) is an OG map built via the processing a short sequence of 3D point clouds. The black cells represent the estimated obstacles, while the white ones correspond to smooth regions. The light blue cells in Figure 4(a) represent regions not touched by the Velodyne rays (its occupancy state is unknown). The orange rectangle illustrates the position of the platform in the map at the time when it (the map) had the appearance shown in the figure. Figure 4(b) shows grid-map derived from the one shown in Figure 4(a). This grid-map is produced by removing the unknown cells and applying the erosion morphological operation with an elliptical kernel. This grip map is used for the detection of road intersections and, from now on, we will call it simply *OG map*.



(a) (b)

Figure 4: 2D occupancy grip (OG) maps produced using a sequence of 3D point clouds captured by the on-board Velodyne 3D scanner. The OG maps cover a region of 60 x 60 meters, having a resolution of 0.2 meters in each dimension. The black cells characterize the obstacles, the white ones non-obstacles, and the blue ones unknown state (cells not touched by the Velodyne rays). (a) OG map created adding the log-odds of the obstacles likelihoods. (b) version of (a) created by removing the unknown regions and by applying the erosion morphological operation.

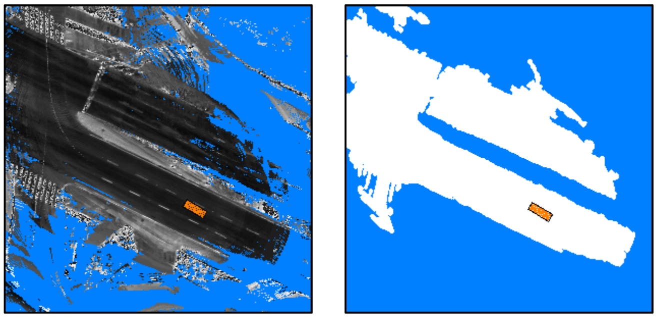
In addition to the OG maps, an extra grid map is computed using laser remission information. This map is built by averaging on each map cell the reflectance of the obstacles hit by the laser rays that projected onto these cells. Figure 5(a) shows the remission grid-map associated with the OG map of Figure 4(b).

A segmentation operation is performed in the remission grid map to highlight roads. For that, a simple threshold is used. To remove narrow regions of high remission (such as lane lines) the dilation morphological operation is employed. Figure 5(b) shows a typical result of our segmentation operation. This grip map is also used for the detection of road intersections and, from now on, we will call it simply *filtered remission*.

C. *Detection of Road Intersections*

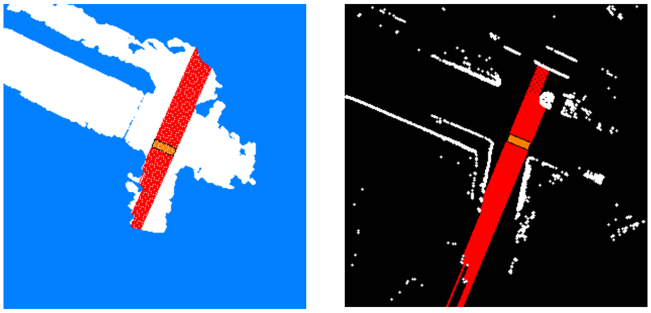
The OG map and the filtered remission map are used for detecting road intersections. The detection process is based on the road width that we estimate using these grid maps. Given the grid maps (that represent the current surroundings of the vehicle) and the estimated position and orientation of the vehicle on these grid maps, exploration lines are generated from the lateral boundaries of the vehicle in its transversal direction (i.e. perpendicular to the longitudinal direction of movements of the vehicle). Each exploration line is used to find the first obstacle in the set of map cells crossed by it. This allows the estimation of the distance between a point at the border of the vehicle and the first obstacle (see Figure 6 for an illustration of the concept).

This operation is performed on both sides of the vehicle. The pair of distances estimated for both sides allows estimating the width of the road. Each exploration line provides an instantaneous estimate of the road width, which is stored in a regressor. Exploration lines are not sampled by time but by space; consequently, the read width estimates are not affected by the speed of the vehicle.



(a) (b)

Figure 5: Filtered Remission map. (a) Remission grid map built by averaging the infrared reflectance of laser targets associated with each cell. The blue cells in (a) represent the parts of the environment that have not been scanned by the sensor. (b) Road-map extracted from (a) by thresholding and dilation. The white areas are classified as road surfaces, while the blue areas are classified as non-road surfaces.



(a) (b)

Figure 6: Transversal exploration lines. In red, the virtual scan performed by the transversal lines until they hit obstacles. The size of the lines is used to estimate the road width. The evolution of the estimated road width is used for inferring road intersections.

Exploration lines are computed in both maps (OG map and filtered remission map), in order to mitigate the impact of the presence of other cars and imperfections in the filtered remission. The inferred width of the road is actually estimated as an average of the contents of the regressor (i.e., it is in fact a low pass filter). The use of these two maps is justified by the fact that they capture complementary information. In certain circumstances, the obstacle detection mechanism employed for building the OG map cannot find any obstacle to infer the limits of the road, e.g. in cases of roads without curbs. However, in such cases, the remission information might still provide useful information to infer the boundaries of the road surface. In other situations, the remission information might be affected by traffic marks painted on the road surface, such as dense zebra crossing lines, but curbs might be present.

Even using both sources of information, there are cases when both are affected, e.g. when other cars (obstacles) are stopped nearby on the zebra lines during sufficient amount of time. Even in those cases, the inference process can avoid negative effects. Even though, in this context of application, the presence of false negatives (failure to infer a road intersection) is not crucial, while false positives (fictitious detection) would have a more negative impact. However false positives would still be tolerable by the localization process.

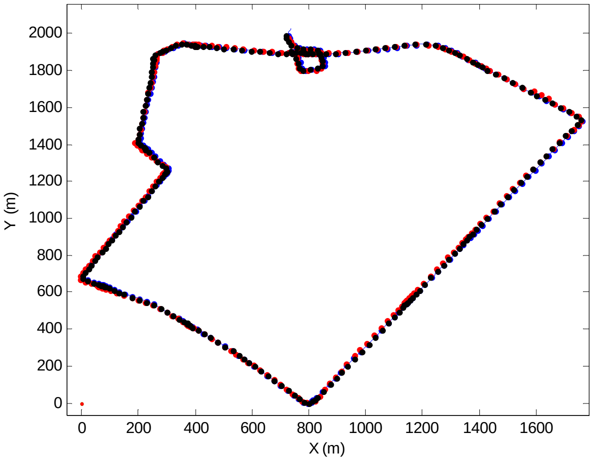


Figure 7: An experiment performed in Vitória-ES, Brazil, where long straight roads are travelled. The PF estimated results (red) are compared with GPS measurements (black) and estimates provided by an off-line graph SLAM (blue).

IV. Experimental Results

A number of experiments and realistic simulations were used to validate the approach. Experimental data obtained using a Brazilian Autonomous Car [16] and in Sydney [1], [13] were adequate for verifying the performance of the estimation process. The experiments performed in Brazil had the advantage of using a better 3D sensor (Velodyne) that, in addition to offer dense 3D point clouds, also offered remission data. Remission data was not available in the Sydney experiment, whose 3D sensing capability was implemented through a nodding LMS200 scanner (as presented in [13], [17]). In this document, only one of the cases is presented.

The experiments were performed in an urban area of Vitória-ES, Brazil, whose satellite image is shown in Figure 9. The road map employed for computing the Base Likelihoods was created automatically by processing and segmenting the road-map images (Figure 9 (left)) from Google-Maps of the area of operation. The generated road maps are low quality representations of the roads. The dead-reckoning process was based on the fusion of vehicle speed and front wheel angle (odometry), and heading direction measurements. The heading direction measurements were provided by a low cost 3D magnetometer. The vehicle used in the experiments is shown in Figure 10.

In order to verify the accuracy of the estimates, two ground truth resources were considered. One of them was provided by the estimates of an expensive (computationally) off-line graph SLAM [5] based on the 3D point clouds. The second reference for positions was given by a GPS working in differential mode. The estimates computed by the PF estimation process we designed were generated in two different conditions. The first is the original approach that does not exploit the observations of the road intersections [1], [10]. The second includes the observations of road intersections described in this paper.

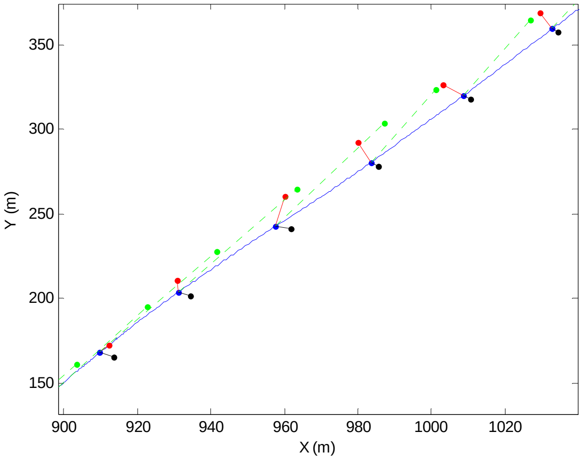


Figure 8: Performance of the localizers. The points indicate the estimated position provided the localizers and the ground truths. Off line graph SLAM, assumed to be the main ground truth, is shown in blue (points and continuous line). GPS estimates are shown in black. PF estimates, using and not using road intersection likelihood functions, are shown in red and green, respectively. Small segments connecting the SLAM estimates to the remaining estimates are used to show the different position estimates associated with the same timestamp. The estimates shown were subsampled in time for simplifying the figure.

The results can be appreciated in Figures 7 and 8. Figure 7 presents the ground truths and the pose estimated with both versions of our PF. As this figure shows, in large scale, the differences appear small, but it is important to note that they are visible. The region in the rectangle in Figure 7 is show in detail in Figure 8. As Figure 8 shows, our new approach performs significantly better than the previous one, dramatically increasing the accuracy in the longitudinal direction.

In Figure 8, the position estimates are shown subsampled in time for helping the interpretation of the results. In order to correlate them in time (not just in space), small line segments are used to indicate the estimates of the localizers that correspond to the same timestamp. Regions where the GPS estimates presented jumps are not presented, as the aim of the figure is to compare the PF estimates with good GPS and

SLAM localization, assumed to be highly accurate. It can be seen that, when no observations of road intersections are used, the expected value of the PF estimates present relevant discrepancies in the longitudinal component of the position.

V. Conclusion

This paper introduces an extension to the global localization method originally presented in [1]. Here, the use of 3D point clouds allowed the detection of road intersections that provided highly informative observations for reducing the uncertainty in the longitudinal component of the estimates of the vehicle's position.

The performance of the new estimation process was tested in two different platforms, in two countries. The obtained performance corroborated the expectation that the extended localization approach is capable of solving the limitation of the original approach in cases were the robotic platform travels long straight line roads. The overall process runs in real-time requiring low processing capabilities.

Future work will include the detection of additional features of the context of operation, such as the proximity of traffic lights and other objects of interest that are usually indicated in public road maps.

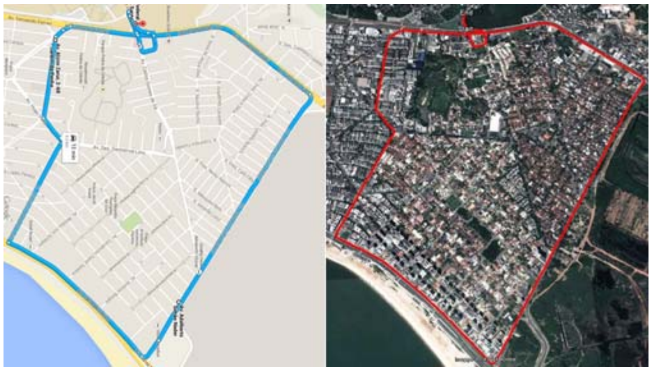


Figure 9: Google maps of the area where the experiments were performed. The map on the left was used for computing the road map, needed for the synthesis of all Base Likelihoods used in the PF estimation process proposed.

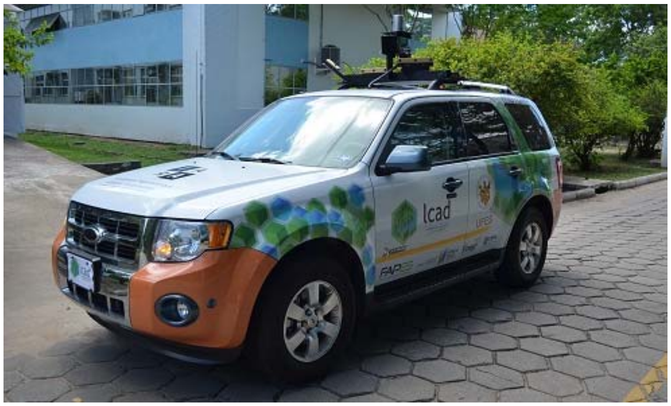


Figure 10: Autonomous platform of the High Performance Computing Lab at Universidade Federal do Espírito Santo, Brazil, named IARA (Intelligent Autonomous Robotic Automobile). This vehicle was used for the experiments in Brazil. The Velodyne sensor is located on the roof of the car. The magnetometer, odometry and GPS sensors are no visible. The data provided by cameras were not used in the localization process.

Acknowledgment

We would like to thank CAPES Foundation, Ministry of Education of Brazil, Brazil (grant 5557/14-3), Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq, Brazil (grants 312786/2013-1 and 552630/2011-0) and Fundação de Amparo à Pesquisa do Espírito Santo – FAPES, Brazil (grant 48511579/2009).

References

[1] J. Guivant and R. Katz, "Global urban localization based on road maps," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007. IROS 2007*, 2007, pp. 1079–1084.

[2] J. Weingarten and R. Siegwart, "EKF-based 3D SLAM for structured environment reconstruction," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, 2005, pp. 3834–3839.

[3] M. Montemerlo and S. Thrun, "FastSLAM 2.0," *FastSLAM Scalable Method Simultaneous Localization Mapp. Probl. Robot.*, pp. 63–90, 2007.

[4] F. Auat Cheein, G. Steiner, G. Perez Paina, and R. Carelli, "Optimized EIF-SLAM algorithm for precision agriculture mapping based on stems detection," *Comput. Electron. Agric.*, vol. 78, no. 2, pp. 195–207, 2011.

[5] S. Thrun and M. Montemerlo, "The GraphSLAM algorithm with applications to large-scale mapping of urban structures," *Int. J. Robot. Res.*, vol. 25, no. 5, pp. 403–430, 2006.

[6] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust Monte Carlo localization for mobile robots," *Artif. Intell.*, vol. 128, no. 1–2, pp. 99–141, May 2001.

[7] J. Levinson, M. Montemerlo, and S. Thrun, "Map-Based Precision Vehicle Localization in Urban Environments," presented at the Robotics: Science and Systems, 2007.

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

[9] A. Ranganathan, D. Ilstrup, and T. Wu, "Light-weight localization for vehicles using road markings," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 921–927.

[10] J. Guivant, M. Whitty, and A. Robledo, "Robust Global Urban Localization Based on Road Maps," in *Robot Localization and Map Building*, H. Yussof, Ed. InTech, 2010.

[11] J. Guivant, S. Cossell, M. Whitty, and J. Katupitiya, "Internet-based operation of autonomous robots: The role of data replication, compression, bandwidth allocation and visualization," *J. Field Robot.*, vol. 29, no. 5, pp. 793–818, 2012.

[12] A. Robledo, S. Cossell, and J. Guivant, "Outdoor ride: Data fusion of a 3d kinect camera installed in a bicycle," in *Australasian Conference on Robotics and Automation. Melbourne, Australia*, 2011.

[13] A. Lohr and J. Guivant, "A Semantic-Based Method for Point Cloud Angular Consistency in Urban Environments," *Australas. Conf. Robot. Autom. Aust.*, 2014.

[14] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP.," in *Robotics: Science and Systems*, 2009, vol. 2.

[15] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, Mass.: MIT Press, 2005.

[16] L. J. Lyrio Júnior, T. Oliveira-Santos, A. Forechi, L. Veronese, C. Badue, and A. De Souza, "Image-based global localization using VG-RAM Weightless Neural Networks," in *2014 International Joint Conference on Neural Networks (IJCNN)*, 2014, pp. 3363–3370.

[17] J. E. Guivant, "Real Time Synthesis of 3D Images Based on Low Cost Laser Scanner on a Moving Vehicle," In Proceedings of the *V Jornadas Argentinas de Robotica*, Bahia Blanca, Argentina, 2008.