

High Integrity Lane-level Occupancy Estimation of Road Obstacles Through LiDAR and HD Map Data Fusion

Edoardo Bernardi, Stefano Masi, Philippe Xu, Philippe Bonnifait

Abstract—In the paper a fast and consistent method to manage uncertainties on detected traffic agents providing reliable results is presented. The information provided by a LiDAR-based object detector is combined with a high-definition map to identify the drivable space of the carriageway. Because the use of a HD map requires the use of a localization system, the uncertainty of the estimated pose shall be handled carefully. A novel approach taking into account the localization uncertainty in the perception task by direct propagation of it onto the LiDAR points is proposed. It is compared with a classical propagation that relies on linearized approximation. The good performances of this approach in terms of integrity are demonstrated by the use of real data acquired at the entrance of a roundabout being a particularly complex situation.

I. INTRODUCTION

Many actions performed by self driving vehicles are considered as safety critical, meaning that a system failure can lead to serious consequences to road users. These maneuvers, such as intersection crossing or roundabout insertion, require an accurate, reliable and consistent knowledge of the surrounding environment to be correctly accomplished. In particular, self-driving cars not only need to precisely detect and localize all road users (cars, bikes, pedestrian, etc.) in the sensor frame but should also be able to position them in the driving environment. The use of High-Definition (HD) maps can enhance the understanding of the driving scene by providing for instance a way to discriminate a vehicle driving on the road from one parked on its side. However, the use of maps requires the vehicle to have a localization system providing in real-time a good estimate of the pose (position and heading). Because localization is not always exact, its uncertainty shall be handled carefully in order to avoid wrong obstacle mapping that could lead to misdetections with possible catastrophic consequences.

In the paper, a method directly propagating the pose uncertainty onto the LiDAR points, guaranteeing the integrity computation of the space occupied by a perceived object is presented. As HD maps provide a lane-level description of the carriageways being very useful for autonomous driving, the obstacles curvilinear occupancy is also computed with reliable confidence bounds.

The paper is organized as follows: an overview of existing LiDAR-based detection algorithms and uncertainties handling methodologies in autonomous driving field is shown in Section II. The novel methodology is presented in Section III-B along with a more classical linearized approximation.

The authors are with Université de Technologie de Compiègne, CNRS, Heudiasyc, UMR 7253, 60203, Compiègne. name.surname@hds.utc.fr

In Section IV, HD maps are used to compute lane level drivable space of the carriageways. Finally, experimental results are given in Section V comparing the methods previously exposed in terms of integrity and availability on overall occupied space and lane level occupation. Such results has been obtained by using real data acquired at the entrance of a roundabout in the city of Compiègne, France (Fig. 2).

II. STATE OF THE ART

Obstacle detection is one of the topic arising most interest in computer vision and there exists a very large literature on image-based approaches. Recent methods using deep neural networks [9], [32], [28] achieve remarkable performance in terms of both precision and recall. Image-based methods usually represent perceived objects in terms of 2D bounding boxes in the image frame. Methods to perform a 3D computation of such boxes from a single camera [23], [19] or a stereo camera [17], [22] can also be found in literature.

More recently, LiDAR-based perception has gained more interest as modern multi-layer LiDARs, (such as the Velodyne VLP-32C used in this study) are capable of yielding a complete view of the environment via a dense point cloud (Fig. 2). The approaches can mainly be classified in two categories: machine learning based and geometrical.

Machine learning methods for vision processing can be adapted to LiDAR point cloud transforming them as a range image [25]. Alternatively the problem can be addressed as a point-wise classification one directly on the 3D data [30]. Nets such as VoxelNet [35], Fully Convolutional Neural Networks [21] or PointPillars [20] can be adopted to perform direct vehicles detection and bounding boxes estimation.

Geometrical approaches for obstacle detection are based on the use of a prior knowledge of the driving scene geometrical structure and are generally decomposed into the following sub-problems.

1) *Ground not-ground segmentation*: LiDAR points belonging to the ground surface are discerned from those not belonging to it. Techniques can mainly be divided in grid-based approaches [36], [18], [1], sector-based ones [13], [7] or plan fitting algorithms [33].

2) *Clustering*: Points not being on the ground and belonging to the same object are grouped together. Connected components in a 2D or 2.5D grid returned by a grid-based segmentation approach can be used [10]. Range image techniques [33], [3] or radially bounded nearest neighbor methodologies [16] can also be used to perform the operation.

3) *Objects bounding*: the space occupied by each clustered object is computed providing a more suited representation.

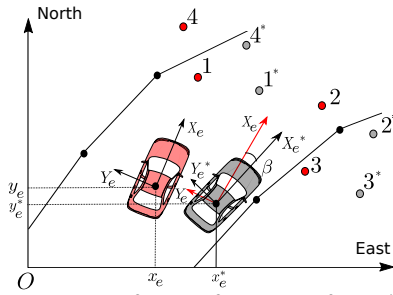


Fig. 1: Sensor to map reference frame transformation. A correct localization (grey) and an erroneous one (red) are considered highlighting four possible cases: true positive (1), false negative (3), true negative (2) and false positive (4).

The task can be performed by surrounding polygons computation algorithms or bounding boxes creation through L-shape fitting [26], [34] or minimum area rectangle ones [4].

In autonomous driving field, uncertainty handling is another widely studied topic. All the information provided by sensors have in fact a certain level of inaccuracy. Consequently, the intelligent vehicle localization and the other road agents position can never be considered as exactly known. Usually, localization uncertainty is reduced by enhancing vehicle position. To perform this task, LiDAR sensors are largely used by matching features to an a priori map [8], [15], or by simultaneous localization and mapping, e.g. SLAM-based approaches [6]. Alternatively the enhancement can be performed via multiple vehicles cooperation [5], [12].

Uncertainty in vehicle positioning is handled in [27] by defining protection levels on the pose in order to augment car size and redefine the occupied space but not considering the perception task. Moreover such lack of information is usually kept in account to perform motion planning [31] or maneuver prediction and decision making [14].

Unlike other methods, in the presented paper the localization uncertainty is directly handled in the perception task by adopting both the proposed methodology and a classical linearization one [29] and comparing the obtained results.

III. HIGH INTEGRITY OBSTACLES 2D OCCUPANCY

To have a complete representation of the surrounding context and correctly locate the perceived objects in the driving environment for drivable surface extraction, it is necessary to project them from the sensor reference frame to the map one. To perform this task the localization uncertainty has so to be carefully taken into account. In the paper the aforementioned positioning lack of information is directly transferred into the perception task to correctly map-matching the detected object and to not incur in misdetections leading to possible catastrophic consequences.

The aforementioned projection is depicted in Fig. 1 in which grey points represent objects perceived by performing sensor to map frame transformation from ground truth localization (grey car). Red ones are instead computed from a wrong localization (red car). It is possible to observe how different cases can be discerned performing frame transformation from a not correctly located sensor. By considering

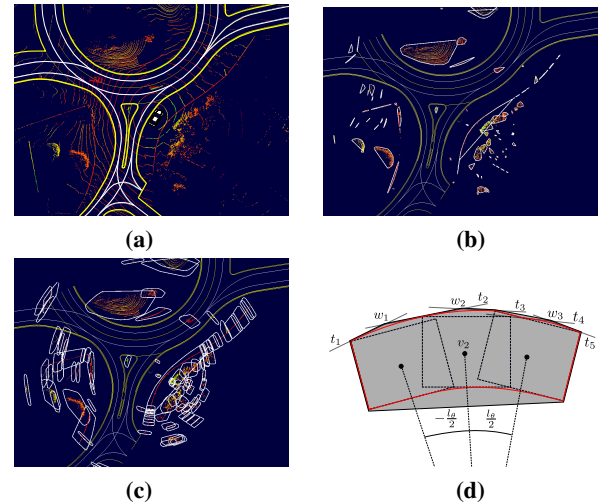


Fig. 2: LiDAR point cloud processes. (a) Full LiDAR point cloud and HD map (in yellow). (b) Clustering and bounding convex hulls computation on not-ground point cloud obtained by segmentation. (c) Convex hulls extension by localization uncertainty propagation. (d) The real occupied space obtained by the rectangle rotation is in red. The convex approximation adopted in the explained methodology is in grey.

as true the position on the road, cases 1 and 3 constitute respectively a true positive and a false negative in which the perceived objects are correctly positioned inside and outside the road surface. On the contrary case 2 represents a true negative where an obstacle is detected on the carriageway while being outside, leading to an unpredictable behavior. More dramatic is case 4, a false positive corresponding to a misdetection in which a road obstacle is not perceived.

To avoid dangerous circumstances, the localization uncertainty is transferred in the perception by enlarging the space occupied by each perceived object. The methodologies presented hereafter can be identically applied with 2D surrounding polygons or 3D bounding boxes (see Fig. 2).

A. Linearized propagation

Let $wq = [w_x, w_y, w_\theta]^T$ be the 2D pose of the vehicle in the world frame and $vz = [v_{x_p}, v_{y_p}]^T$ the coordinates of a point in the vehicle frame. The coordinate of such point in the world frame is given by:

$$wz = f(vz, wq) = \begin{bmatrix} \cos(w\theta) & -\sin(w\theta) \\ \sin(w\theta) & \cos(w\theta) \end{bmatrix} vz + \begin{bmatrix} w_x \\ w_y \end{bmatrix}. \quad (1)$$

Suppose the pose of the vehicle being estimated by $w\hat{q} = wq + \epsilon_q$ and the error following a normal distribution $\mathcal{N}(0, \Sigma_q)$. The uncertainty over the pose coded via the covariance matrix Σ_q can be propagated upon the estimated point position $w\hat{z} = f(vz, w\hat{q})$ using a first order approximation $\Sigma_z \approx J\Sigma_q J^T$ where J is the Jacobian matrix of f defined as

$$\frac{\partial f(vz, wq)}{\partial wq} = \begin{bmatrix} 1 & 0 & -v_{x_p} \sin(w\theta) - v_{y_p} \cos(w\theta) \\ 0 & 1 & v_{x_p} \cos(w\theta) - v_{y_p} \sin(w\theta) \end{bmatrix} \quad (2)$$

Given a risk $\alpha \in [0, 1]$, it is possible to compute a confidence domain $C(\alpha)$ such that it is possible to guarantee

$$Pr(wz \in C(\alpha)) = 1 - \alpha. \quad (3)$$

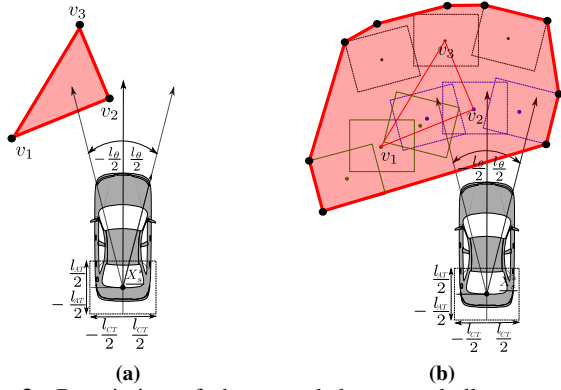


Fig. 3: Description of the extended convex hulls computation representing the confidence domain of a perceived object. (a) Perceived object in terms of bounding polygon including it. The confidence domains on the three dimension around the vehicle position are also depicted. (b) Extended convex hull built from the confidence domain of each vertex of the original hull representing the confidence domain of the perceived object.

There exists an infinite number of such confidence domains. In the case of a Gaussian distribution with covariance matrix Σ , this domain is commonly represented in the form of an ellipse with its major and minor axis aligned along the eigen vectors of Σ . This domain is chosen because it results being the smallest, in terms of area, verifying Eq. (3). In the presented case, an ellipse confidence domain is difficult to be geometrically manipulated. In particular, it is complex to compute the overall occupied space by a perceived object. Consequently it is proposed the use of a rectangle having sides aligned with the eigen vectors of Σ and with length l_i computed as:

$$l_i = 2\Phi^{-1} \left(\frac{1 + (1 - \alpha)^{1/2}}{2} \right) \lambda_i, \quad (4)$$

where λ_i is the i -th eigen value of Σ and Φ is the cumulative probability density of the normal distribution $\mathcal{N}(0, 1)$.

Applying this formalism, each point of a perceived object, in terms of LiDAR point or bounding polygon vertex, is represented as a rectangle containing the position uncertainty. The overall occupied space, with such uncertainty included, is then obtained by computing the polygon bounding all the rectangles vertices previously calculated.

This technique can be conceptually compared with a Minkowski sum between the general polygon bounding an object and the rectangle approximating the uncertainty ellipsoid of the centroid of such polygon. In fact the ellipsoid dimension depends on the hull vertex distance from the sensor perceiving it.

B. Direct propagation

A drawback in the previously introduced approach is the first order approximation induced by Eq. (2) leading to underestimated covariance matrices. To overcome this issue, an approach directly propagating vehicle pose confidence domain onto the one of the perceived object is proposed.

First, the covariance matrix Σ_q of the vehicle pose in the cross-track (CT) and along-track (AT) directions w.r.t. the heading ${}^w\theta$ of the vehicle is projected. Let σ_{CT}^2 , σ_{AT}^2 and σ_θ^2 be the resulting variances in these three dimensions. Similarly to the previous approach, a confidence domain is built over the vehicle pose in the form of a “cube” in the CT , AT , and heading three-dimensional space. The length of each side is given by:

$$l_i = 2\Phi^{-1} \left(\frac{1 + (1 - \alpha)^{1/3}}{2} \right) \sigma_i, \quad (5)$$

where $i \in \{CT, AL, \theta\}$.

To propagate this confidence domain onto a point v_z , the rectangle confidence domain resulting from the uncertain position is firstly computed:

$$[{}^v x_p \pm l_{CT}/2] \times [{}^v y_p \pm l_{AL}/2]. \quad (6)$$

Then, in order to take into account the uncertainty over ${}^w\theta$, two other rectangles are computed by using a rotation of angles $\pm l_\theta/2$ (in the vehicle frame the heading is always null, ${}^v\theta = 0$). In order to fully represent the space occupied by the rectangle rotation, the circles generated by the farthest vertices are computed. Then the tangent lines to such circles, passing through the aforementioned vertices and the corresponding ones in rotated rectangles, are extracted. Finally, the tangent lines intersection points along with the corners of the three rectangles are used to compute a convex hull corresponding to the confidence domain of the point.

This procedure is depicted in Fig. 2 (d): the rectangles represent the confidence domain resulting from the uncertain position and its rotations. The lines t_1 , t_2 and t_3 are obtained as tangent lines to the circle generated by the top left corner rotation, passing through such vertex in the three rectangles. The same reasoning is applied for the computation of t_4 and t_5 by the use of the top right corner (in a symmetric case also the tangent through the third rectangle vertex has to be computed). Points w_i are computed from the intersections of the previously calculated tangent lines. Finally, the convex hull representing the point confidence domain is computed using rectangle vertices and tangent lines intersection points.

To obtain the confidence domain of the whole perceived object, the convex hull of each of its points (in terms of LiDAR points or bounding polygon vertices) is computed and the resulting vertices are used for the computation of an extended convex hull representing such domain. This computation is well depicted in Fig. 3.

This procedure can be applied to the linearization method by replacing the direct propagation confidence domain polygons with the rectangles mentioned in Section III-A.

As all these computations are performed in the vehicle reference frame, the transformation f is directly applied on the vertices of the previously computed confidence domain to get the final occupancy domain in the world frame. Unlike the previously presented approach, no approximation is necessary to propagate such confidence domain.

IV. LANE-LEVEL CURVILINEAR OCCUPANCY

Once the 2D space occupied by a perceived object has been calculated, it is possible to utilize the HD map in order to compute whether it results being outside the road surface or on it. For the obstacles on the carriageway or partially on it, the lane level occupation area can also be computed.

A. Map-based filtering

The HD maps adopted in the presented study contain the borders of the drivable surface. The world space is therefore divided into two categories “road” and “not road” with the aim of determining whether a detected object is on the drivable surface or not. To exploit this task a general intersection algorithm between the polygon defined by the street borders and the confidence domain of an object can be adopted. The resulting classification is therefore constituted by three categories: “road”, “not road” and “uncertain” (partially inside and partially outside the road).

B. Lane-level curvilinear occupancy intervals

For autonomous navigation, interaction between ego-vehicle and surrounding road users can be performed at lane level rather than at Euclidean space level. Therefore, the curvilinear occupancy intervals generated by each obstacle along the driving lanes result being more informative than the 2D occupancy space. In the available HD map, additionally to the road borders, the middle of each lane of the road is encoded as a polyline, *i.e.*, a sequence of line segments. In general, this task can be performed with every map presenting lanes information.

To compute the curvilinear occupation of a 2D occupancy polygon, first the intersection between such polygon and each one representing a lane is computed. The result for each lane is a sub-polygon representing the 2D space effectively occupied on such lane. Then each vertex of this sub-polygon is map-matched onto the HD map lanes. The *lanelet* formalism has been used to compute the map-matching as it has been shown to be more robust to discontinuities [11], [2]. To make computation faster, the HD map has been organized with a graph-based structure and a breadth-first search algorithm is used during the map-matching step.

This operation is performed on all the obstacles being classified as “road” or “uncertain” in order to keep the integrity constraint. In fact, not considering an object which confidence domain intersects a portion of the drivable surface, even if small, could lead to collisions.

Regarding the polygon intersection two different situations may occur:

1) *Only one lane is intersected*: The resulting-map matching operation is performed directly on the original occupation polygon vertices. A single occupancy interval is defined by the minimum and the maximum curvilinear abscissa of all the points on the polyline.

2) *Several lanes are intersected*: This may occur when a vehicle is changing lane or when the occupancy space is too large. This situation implies that there is no certainty regarding which lane is effectively occupied. From an integrity

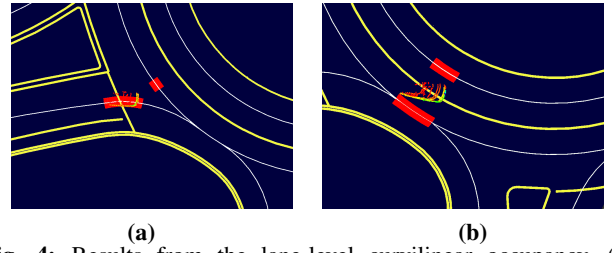


Fig. 4: Results from the lane-level curvilinear occupancy. (a) Occupation intervals generated by a vehicle entering a roundabout. (b) Occupation intervals from a vehicle driving between two lanes.

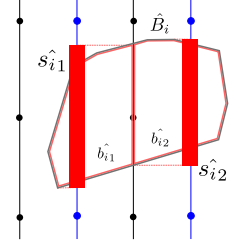


Fig. 5: The red polygon \hat{B}_i intersects two different lanes generating two sub-polygons b_{i1} and b_{i2} . All the vertices of each sub-polygons are map-matched on the polyeline in the middle of the occupied lane. The result is a set of curvilinear intervals representing lane level occupied space $\hat{S}_i = \{\hat{s}_{i1}, \hat{s}_{i2}\}$.

point of view, every candidate lane has to be considered as occupied by a curvilinear interval (Fig. 4).

The whole procedure is represented in Fig. 5.

V. EXPERIMENTAL RESULTS

In the paper the adopted method, for both segmentation and clustering, is the recently developed one by Zermas *et al.* [33]. The reader may refer to this paper for detailed information. The ground not-ground segmentation step is based on the assumption that the ground points belong to planes, which means that those having low height values are more likely to be on the soil. The point cloud is first divided in multiple sub-point clouds and, for each of them, a plan is computed. In this way is possible to have a better evaluation of the road slopes. Once the planes have been estimated, a thresholding is applied on the LiDAR points to discriminate whether a point belongs to the ground or not. The output of this algorithm is the not-ground point cloud.

In the clustering phase, the goal is to group all the not-ground points belonging to the same object. To perform this task, the 3D LiDAR points are treated as pixels in a 2D cylindrical image and labeled by using a two-run connected components algorithm [10].

Once all points of the point cloud have been assigned to a cluster, they are first projected in a 2D space by discarding the upward dimension, assumed to be orthogonal to the ground surface, in order to have a representation corresponding to the map one. Because the clusters may represent any kind of obstacle, it has not been used a predefined bounding shape, such as rectangles, usually chosen to bound vehicles. On the contrary, each detected obstacle has been represented by its minimal convex hull resulting in convex polygons surrounding the clusters as shown in Fig. 2.

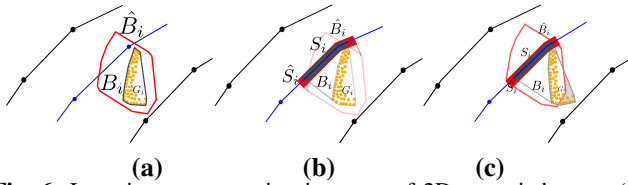


Fig. 6: Integrity representation in terms of 2D occupied space (a) and curvilinear occupied space (b). Red hulls and segments are acquired from wrongly localized sensor (red car) while grey ones are obtained from ground truth localization (grey car). (c) As the red polygon \hat{B}_i does not contain the gray cluster B_i entirely, it does not satisfy the 2D integrity property. Its projection \hat{S}_i at lane level along the blue polyline, however, includes the true curvilinear occupancy S_i .

TABLE I: Average computational time in *ms* for the whole processing (segmentation, clustering, uncertainty propagation, map filtering and map matching).

Step	Segm.	Clust.	Propagation	Map Filt.	Map Match.
Direct	7.753	9.317	0.183	11.384	5.075

Then uncertainty propagation has been performed through the methodology presented in Section III-B in order to extract the overall 2D space occupation, also at lane level, and avoid the dangerous situations depicted in Fig. 1.

To validate the proposed method, several experiments have been carried out. A dataset has been recorded at the entrance of a roundabout in the cities of Compiègne and Rambouillet, France. A Renault ZOE experimental vehicle was equipped with a Velodyne VLP-32 LiDAR and a GNSS/IMU NovAtel SPAN-CPT with RTK correction to provide a ground truth localization. The Velodyne VLP-32 LiDAR provided scans at a frequency of 10 Hz. The dataset is composed of ten sequences of about 10 minutes of road traffic, with a moderately dense traffic flow. To simulate errors on the localization, a Gaussian noise was injected into the pose provided by the ground truth. The standard deviation values for such noise are $\sigma_x = 0.1$ m, $\sigma_y = 0.16$ m and $\sigma_\theta = 0.01$ rad.

First, the 2D occupancy of the linearized and direct method from an integrity point of view has been computed. In the presented context the integrity property refers to the capability of the estimated occupation space, computed from a wrongly localized sensor, to contain the whole cluster of point cloud or bounding polygon representing the object perceived at exact localization. Given a raw-data cluster G_i , composed of points g_i , a bounding polygon B_i that bounds points of G_i , a transformation function f and a risk α , a bounding polygon defined as \hat{B}_i , obtained by applying the transformation f to each vertex of B_i , keeps the integrity property if and only if: $Pr(B_i \subseteq \hat{B}_i) \geq 1 - \alpha$.

The same reasoning can be applied at lane level. Given a set of segments S_i representing the lane level drivable space occupied by a bounding polygon B_i and a risk α , a set of segments \hat{S}_i , computed by map matching the transformed bounding polygon \hat{B}_i , satisfies the integrity property if and only if: $Pr(S_i \subseteq \hat{S}_i) \geq 1 - \alpha$.

These integrity definitions are depicted in Fig. 6.

Considering a risk α and the consequent confidence level $1 - \alpha \in \{0.90, 0.95, 0.99, 0.999, 0.9999\}$, the occupancy con-

TABLE II: Integrity comparison of the linearized and direct methods for 2D occupancy confidence domain.

$1 - \alpha$	90%	95%	99%	99.9%	99.99%
Linearized	44.54%	49.50%	56.45%	62.59%	66.33%
Direct	97.69%	98.87%	99.21%	99.69%	99.88%

TABLE III: Integrity comparison of the linearized and direct methods for lane-level occupancy confidence domain.

$1 - \alpha$	90%	95%	99%	99.9%	99.99%
Linearized	91.04%	91.50%	91.96%	92.82%	93.05%
Direct	99.23%	99.30%	99.35%	99.73%	99.90%

fidence domain has been computed by using both methods from a not correctly localized sensor. Then, for each obstacle cluster, it has been computed whether all the points within the cluster, acquired from the exact localization, were contained in the occupancy confidence domain or not. A method has been considered to satisfy the integrity constraint if the ratio of clusters entirely contained in their confidence domain is greater than the confidence level $1 - \alpha$. Table II summarizes the results obtained in the experiments. It can be seen that the linearized method does not provide reliable results. This is due to the non linearity induced by an error on the heading of the vehicle that leads to a *banana* shaped distribution of the LiDAR points which is badly approximated by a Gaussian one. On the other hand, the direct approach keeps the integrity up to a confidence of 99%. For higher degrees of confidence, the integrity objective is not exactly reached but the statistics result being very close.

The same analysis has also been performed in terms of occupied curvilinear space. Table III shows that the direct propagation method also outperforms the linearization one at lane level. This is directly related to the better performances identified in 2D space occupation. Nevertheless, the curvilinear integrity values are significantly higher than the 2D occupation ones. This behavior is due to the fact that a 2D bound that does not fully include a given obstacle may still include its curvilinear occupancy at the lane level. Fig. 6 (c) illustrates such a situation.

All the algorithms have been coded in C++ using the ROS middleware. The computation for the whole processing including segmentation, clustering, uncertainty propagation, map filtering and map matching reached real-time computation with an average computational time of about 33 ms on a 2.2 GHz CPU. From Table I it is possible to see how the uncertainty handling and the map-matching phases do not degrade the real time performances being perfectly suited for autonomous vehicles navigation.

Finally, to demonstrate the usefulness of the method, it has been qualitatively validated in an autonomous driving scenario. The roundabout crossing algorithm presented in [24] has been implemented replacing vehicles positions by their occupancy intervals extracted by LiDAR detection and uncertainty injection in curvilinear domain. Autonomous driving tests were conducted on the Séville experimental test track of the University of technology of Compiègne. A vehicle and a bicycle were looping inside a roundabout with the aim of having the autonomous vehicle inserting in

the roundabout. Tests with a 95% confidence level led to satisfactory behavior for autonomous navigation.

VI. CONCLUSION

In the paper, a fast and efficient method to propagate the uncertainty of the ego-vehicle localization in the perception task with good integrity has been proposed. It has been proved that the presented methodology of direct uncertainty propagation preserves the integrity property and outperforms a state-of-the-art method. This result has been shown in both domains, the 2D space occupation and the lane-level curvilinear space occupation. By its application it has been possible to extract the occupied drivable surface avoiding dangerous situations, making the obtained information suited to perform safety critical maneuvers. In the end, the methodology has been implemented and tested in a real autonomous vehicle to perform such maneuvers.

Acknowledgment: This work was carried out in the framework of Equipex ROBOTEX (ANR-10-EQPX-44-01) and Labex MS2T (ANR-11-IDEX-0004-02). It was also carried out within SIVALab, a shared laboratory between Renault and Heudiasyc CNRS/UTC, through the TORNADO project.

REFERENCES

- [1] A. Asvadi, P. Peixoto, and U. Nunes. Detection and tracking of moving objects using 2.5D motion grids. In *IEEE International Conference on Intelligent Transportation Systems*, pages 788–793.
- [2] P. Bender, J. Ziegler, and C. Stiller. Lanelets: Efficient map representation for autonomous driving. In *IEEE Intelligent Vehicles Symposium Proceedings*, pages 420–425.
- [3] I. Bogoslavskiy and C. Stachniss. Fast range image-based segmentation of sparse 3d laser scans for online operation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 163–169.
- [4] A. Börcs, B. Nagy, and C. Benedek. *Dynamic 3D Environment Perception and Reconstruction Using a Mobile Rotating Multi-beam Lidar Scanner*, pages 153–180. Springer International Publishing, 2015.
- [5] F. Bounini, D. Gingras, H. Pollart, and D. Gruyer. Real time cooperative localization for autonomous vehicles. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems*, pages 1186–1191.
- [6] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser. Simultaneous localization and mapping: A survey of current trends in autonomous driving. *IEEE Transactions on Intelligent Vehicles*, 2(3):194–220.
- [7] P. Chu, S. Cho, S. Sim, K. Kwak, and K. Cho. A fast ground segmentation method for 3D point cloud. *Journal of Information Processing Systems*, pages 491–499.
- [8] F. Ghallabi, G. El-Haj-Shhade, M. Mittet, and F. Nashashibi. LIDAR-Based road signs detection For Vehicle Localization in an HD Map. In *IEEE Intelligent Vehicles Symposium*.
- [9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778.
- [10] L. He, Y. Chao, and K. Suzuki. A run-based two-scan labeling algorithm. *IEEE Transactions on Image Processing*, pages 749–756.
- [11] E. Héry, S. Masi, P. Xu, and P. Bonnifait. Map-based curvilinear coordinates for autonomous vehicles. In *IEEE International Conference on Intelligent Transportation Systems*, pages 1–7.
- [12] E. Héry, P. Xu, and P. Bonnifait. Pose and covariance matrix propagation issues in cooperative localization with lidar perception. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 1219–1224.
- [13] M. Himmelsbach, F. v. Hundelshausen, and H. . Wuensche. Fast segmentation of 3D point clouds for ground vehicles. In *IEEE Intelligent Vehicles Symposium*, pages 560–565.
- [14] C. Hubmann, J. Schulz, M. Becker, D. Althoff, and C. Stiller. Automated driving in uncertain environments: Planning with interaction and uncertain maneuver prediction. pages 1–1.
- [15] E. Javanmardi, Y. Gu, M. Javanmardi, and S. Kamijo. Autonomous vehicle self-localization based on abstract map and multi-channel lidar in urban area. volume 43, pages 1 – 13.
- [16] K. Klasing, D. Wollherr, and M. Buss. A clustering method for efficient segmentation of 3D laser data. In *IEEE International Conference on Robotics and Automation*, pages 4043–4048.
- [17] H. Königshof, N. O. Salscheider, and C. Stiller. Realtime 3D Object Detection for Automated Driving Using Stereo Vision and Semantic Information. In *IEEE International Conference Intelligent Transportation Systems*.
- [18] D. Korchev, S. Cheng, Y. Owechko, and K. Kim. On real-time LiDAR data segmentation and classification. 2013.
- [19] J. Ku, A. Pon, and S.L. Waslander. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 11867–11876.
- [20] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom. Pointpillars: Fast encoders for object detection from point clouds. 2018.
- [21] B. Li. 3D fully convolutional network for vehicle detection in point cloud. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1513–1518.
- [22] P. Li, X. Chen, and S. Shen. Stereo r-cnn based 3d object detection for autonomous driving. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7644–7652.
- [23] F. Manhardt, W. Kehl, and A. Gaidon. ROI-10D: monocular lifting of 2D detection to 6D pose and metric shape. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2069–2078.
- [24] S. Masi, P. Xu, and P. Bonnifait. Adapting the virtual platooning concept to roundabout crossing. In *IEEE Intelligent Vehicles Symposium*, pages 1366–1372.
- [25] J. Mei, B. Gao, D. Xu, W. Yao, X. Zhao, and H. Zhao. Semantic segmentation of 3D LiDAR data in dynamic scene using semi-supervised learning. In *IEEE International Conference Intelligent Transportation Systems*.
- [26] S. Qu, G. Chen, C. Ye, F. Lu, F. Wang, Z. Xu, and Y. Gel. An efficient L-shape fitting method for vehicle pose detection with 2D lidar. In *IEEE International Conference on Robotics and Biomimetics*, pages 1159–1164.
- [27] T. G.R. Reid, S.E. Houts, R. Cammarata, G. Mills, S. Agarwal, A. Vora, and G. Pandey. Localization requirements for autonomous vehicles. *SAE International Journal of Connected and Automated Vehicles*, 2(3).
- [28] J. Ren, X. Chen, J. Liu, W. Sun, J. Pang, Q. Yan, Y.W.Tai, and L. Xu. Accurate single stage detector using recurrent rolling convolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5420–5428.
- [29] R. Smith, M. Self, and P. C. Cheeseman. Estimating uncertain spatial relationships in robotics. *CoRR*.
- [30] B. Wu, A. Wan, X. Yue, and K. Keutzer. SqueezeSeg: Convolutional neural nets with recurrent CRF for real-time road-object segmentation from 3D LiDAR point cloud. In *IEEE International Conference on Robotics and Automation*, pages 1887–1893.
- [31] W. Xu, J. Pan, J. Wei, and J. M. Dolan. Motion planning under uncertainty for on-road autonomous driving. In *2014 IEEE International Conference on Robotics and Automation*, pages 2507–2512.
- [32] F. Yang, W. Choi, and Y. Lin. Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2129–2137.
- [33] D. Zermas, I. Izzat, and N. Papanikolopoulos. Fast segmentation of 3d point clouds: A paradigm on lidar data for autonomous vehicle applications. In *IEEE International Conference on Robotics and Automation*, pages 5067–5073.
- [34] X. Zhang, W. Xu, C. Dong, and J. M. Dolan. Efficient L-shape fitting for vehicle detection using laser scanners. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 54–59.
- [35] Y. Zhou and O. Tuzel. VoxelNet: End-to-end learning for point cloud based 3D object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4490–4499.
- [36] Q. Zhu, L. Chen, Q. Li, M. Li, A. Nüchter, and J. Wang. 3D LiDAR point cloud based intersection recognition for autonomous driving. In *IEEE Intelligent Vehicles Symposium*, pages 456–461.